Development of Gas-Liquid Flow Regimes Identification Using a Non-Invasive Ultrasonic Sensor, Belt-Shape Features, and Convolutional Neural Network in an S-shaped Riser

Somtochukwu Godfrey Nnabuife[a¶], Boyu Kuang[b*¶], James Whidborne[c&], Zeeshan A. Rana[b&]

*Abstract --- * **The problem of classifying gas-liquid two-phase flow regimes from ultrasonic signals is considered. A new method, belt-shaped features (BSF), is proposed for performing feature extraction on the pre-processed data. A convolutional neural network (CNN/ConvNet)-based classifier is then applied to categorize into one of the four flow regimes: annular, churn, slug, or bubbly. The proposed ConvNet classifier includes multiple stages of convolution and pooling layers, which both decrease the dimension and learns the classification features. Using experimental data collected from an industrial-scale multiphase flow facility, the proposed ConvNet classifier achieved 97.40%, 94.57, and 94.94% accuracy, respectively, for the training set, testing set, and validation set. These results demonstrate the applicability of the BSF features and the ConvNet classifier for flow regime classification in industrial applications.**

Index Terms— **ultrasonic sensor, belt-shaped features (BSF), convolutional neural networks (CNNs), S-shaped riser**

I. INTRODUCTION

TWO-phase gas-liquid flows occur in pipelines and risers in a wide range of industrial and engineering processes, such as in the transport of hydrocarbon in subsea, nuclear power plants, petrochemical plants, food processing plants, chemical reactors, steam boilers, and their associated process piping and condensers. [1]

Two-phase gas-liquid flows are classified into flow regimes according to the phase distribution in pipes. The most frequently experienced flow regimes in vertical or S-shaped pipeline risers are annular, bubbly, slug, and churn flows.

Different two-phase flow regimes occur depending on the volume fractions, the velocities, the properties of fluids, and the pipe geometry [2]. The identification of flow regimes is vital for the design of pipelines and equipment, as well as safety operations [3]. For example, the detection of slug flow can help control the system to mitigate the risk of overflowing the separator on the offshore production platform [4]. The flow regime can be predicted using an empirical flow regime map [5]; however, flow regime maps are prone to errors, as only a few factors are considered in creating them [6]. Moreover, the input parameters, such as gas and liquid superficial velocities, are often not measured in operations.

Measurements to identify the flow regime may be made using various methods, including nuclear radiation methods, Coriolis flowmeters, laser Doppler anemometry (LDA), computer tomography (optical, resistive, capacitive, and X-ray), impedance sensors, magnetic resonance imaging (MRI), and particle image velocimetry (PIV). Non-intrusive and non-invasive methods are preferred for practical reasons. Radiation techniques, especially the gamma-ray absorption-based methods, are non-intrusive and reliable [7]–[10] and are effective and accurate. However, because ionizing radiation requires strict safety compliance, they are not widely accepted by oil and gas operators.

A Doppler ultrasonic sensor that deploys continuous-wave ultrasound signals can also achieve non-invasive flow velocity measurement [11]. This method deploys a shift in frequency constituting the flow velocities to develop a means of making flow regime predictions [12]. The applicability of continuous-wave Doppler ultrasound (CWDU) in two-phase flow velocity measurements was investigated by [13]. They suggested the deployment of frequency resolution methods to solve the issues in velocity profile measurement through colored noise presence, but this caused a severe problem for the classical frequency estimators.

Impedance sensors have a raw output signal that is a function of the void fraction [14]. This feature, due to its close relation to the flow regime, requires less computational effort in mapping the signal features to the flow regime. Pressure transducers are less expensive, readily available for a wide range of operating conditions, well developed, and fulfill most

¶ These authors contributed equally to this work.
& These authors also contributed equally to this work.

Nnabuife Godfrey Somtochukwu is with the Geo-Energy Engineering Centre, School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, MK43 0AL, UK (Email: g.nnabuife@cranfield.ac.uk).

Boyu Kuang and Zeeshan A. Rana are with the Centre for Computational Engineering Sciences (CES), School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, MK43 0AL, UK (Email: neil.kuang@cranfield.ac.uk; zeeshan.rana@cranfield.ac.uk).

James Whidborne is with the Dynamics Simulation and Control Group, School of Aerospace, Transport and Manufacturing, Cranfield University, Cranfield, MK43 0AL, UK (Email address: j.f.whidborne@cranfield.ac.uk).
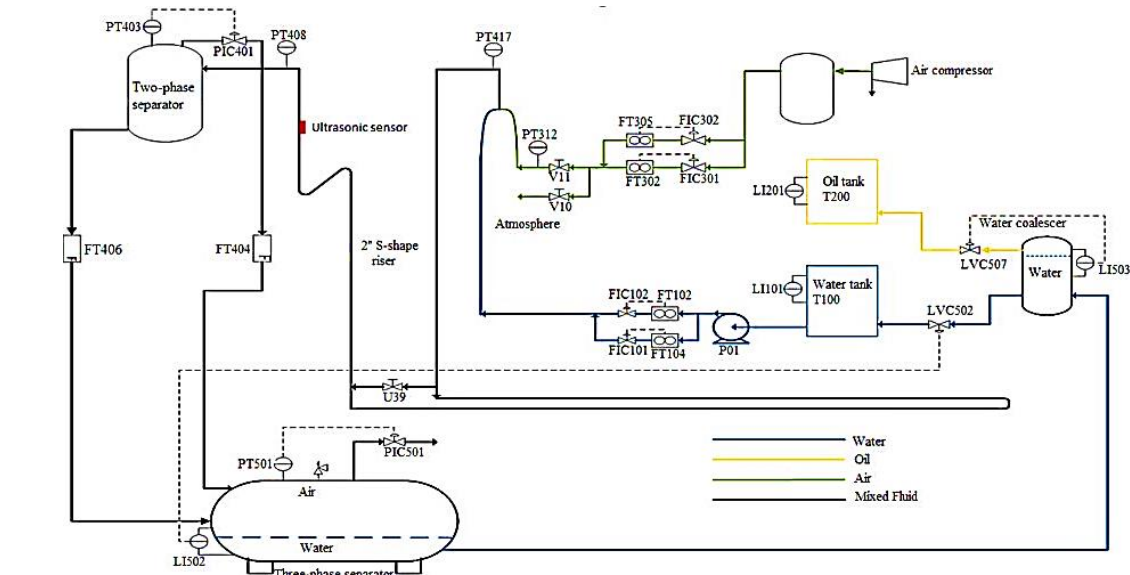
Fig. 1 Schematic diagram of the multiphase flow test facility [22], [24].

of the operational safety regulations [14]. Electrical impedance and pressure transducers are invasive techniques that require either direct contact with fluids, making them prone to corrosion and blockage, or the insertion of a special dielectric pipe for electrical capacitance sensors. Non-invasive and non-radioactive methods are highly attractive, as they remove the need for the immersion of instrumentation in the flow, and are less expensive to design, as they do not use radioactive elements. Moreover, the operating expenditure (OPEX) is lower compared to radioactive instruments due to the excessive costs associated with safety, environmental, and health issues [15]. The ultrasonic technique is a promising alternative that is less complex, hazardous, and costly.

A significant advance in the objective identification of the flow regime was established by the introduction of an artificial neural network (ANN) [16]–[19]. A more successful objective process was established by classifying the flow regime indicators acquired through non-intrusive impedance probes and a Kohonen self-organizing neural network (SONN) [20]. The classification using SONN was initially carried out using the PDF of the void fraction signals as an indicator. This was later enhanced when the cumulative probability density function (CPDF) of the impedance void meter signals was introduced [19]. Flow regime identification using an ANN was reviewed with regard to its applications with electrical impedance sensors and pressure differential transducers as measuring devices. This choice was made due to increasing interest in using an ANN with these measurement methods, in contrast to the use of a support vector machine (SVM) [14] or an image analysis of dynamic neutron radiograph videos [21]. In recent research, a flow regime classification method based on principal component analysis (PCA) and SVM achieved 85% accuracy [22]. Eyo et al. [23] developed an online gas-liquid objective flow regime identifier using conductance signals and kernel methods and achieved a performance accuracy of 90% against the subjective visual method. Furthermore, another method based on the twin-window feature extraction (TFE) algorithm and deep neural networks (DNN)

achieved an outstanding accuracy of 96.28%; the twin-window strategy accomplished significant performance in the one-dimensional signal [24]. Kuang et al. further propose a flow regime identification benchmark, which mainly covers fully convolutional networks [25]. Since the early 1980s, the artificial neural network has been employed extensively for applications such as parameter estimation, fault detection, model-based control, dynamic modeling, process monitoring, and adaptive control [11], [14], [26]–[28].

Convolutional neural networks (CNNs/ConvNets) are a class of deep learning methods that is a popular and powerful approach for applications in computer vision [29], natural language processing [30], and medical image processing [31]. A convolutional neural network is designed to adaptively and automatically learn spatial hierarchies of features through the backpropagation algorithm using multiple layers such as pooling layers, convolution layers, and fully connected layers. The pooling and convolution layers extract features, while the fully connected layer maps the extracted features into the output layer, which is a typical approach for a classification task [32]. Convolutional neural networks' identification superiority and accuracy have been proven in a wide range of applications such as face recognition [33], handwritten digits [34], and traffic sign recognition [35].

In this paper, we propose a new approach to the problem using a non-invasive ultrasonic sensor for the measurements, a new belt-shape features (BSF) extraction algorithm, and a ConvNet classifier. The approach is applied to identify two-phase gas-liquid flow regimes in an S-shaped riser system. The method is based on the assumption that the gas-liquid flow patterns will have unique signatures on the output signals of the CWUD device when subject to the two-phase flows. The main contributions of this research are as follows: (i) this research appears to be the first implementation of CWDU signals and a CNN-based classifier to identify the flow regime in an S-shaped riser; (ii) this research proposes a novel BSF feature extraction algorithm from the one-dimensional ultrasonic signals, which codes the information to a belt-shaped feature – compared to
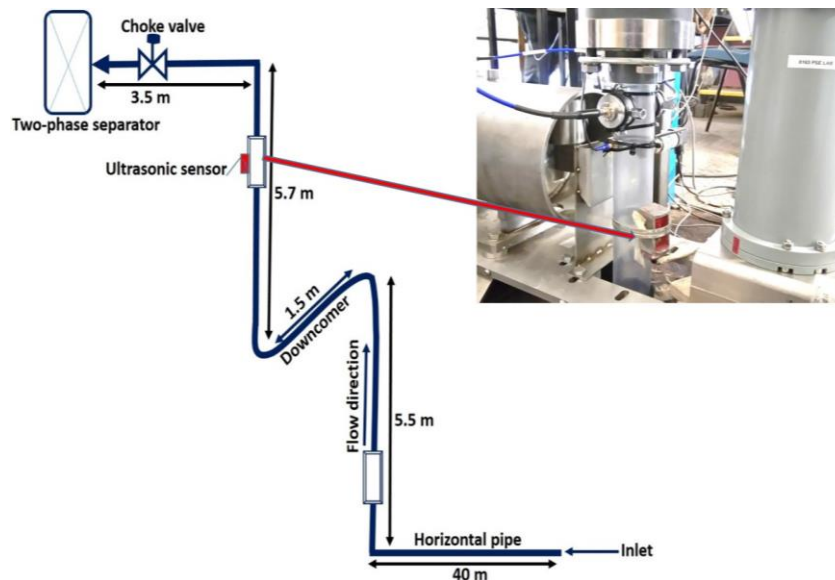
Fig. 2 An S-shaped pipeline-riser system with clamped-on ultrasonic sensor [22], [24].

the state-of-art feature extraction algorithm, the BSF algorithm is more flexible, robust, and generalized; and (iii) the proposed BSF-based pre-processing could expand the number of samples for ConvNet classifier training, which would address the common challenge of lack of data in experimental engineering.

The paper is organized as follows. In the next section, the experiment details and data acquisition are described. In Section III, the data processing required for the flow regime identifier is described including the new BSF algorithm and the ConvNet classifier training. In Section IV, the results are presented, analyzed, and compared with other methods. Section V provides some conclusions. Section VI depicts the future works.

II. EXPERIMENT AND DATA COLLECTION

A. The multiphase flow test facility

The Cranfield University Oil and Gas Centre has a nearly industrial-scale multiphase test rig and is one of the best in the UK. It is fully computerized with the standard modern industrial control system. The multiphase flow test rig was designed to carry out a systematic investigation of multiphase flow measurement, control, and transportation of gas and liquid. Off-shore systems are normally made up of oil, water, and gas; however, in this experiment, only air and water were passed through the test facility. The multiphase flow rig is operated and controlled using DeltaV, supervisory control, and data acquisition (SCADA) software from Emerson Automation Solutions. The test rig schematic diagram is displayed in Fig. 1.

The air-water flow rate is automatically regulated using the DeltaV SCADA to obtain different operating flow conditions [22]. The air-water flow is mixed in the mixing section of the pipeline before being passed through the horizontal section and then through the vertical riser, with an S-shaped riser installed as illustrated in Fig. 1. After the topside riser system, the air-water flow is separated using a two-phase separator. The air is released into the atmosphere while the water is returned to the water repository tank after cleaning.

B. Process variables

The process variables were acquired using the DeltaV SCADA system. All the sensors connected to the pipeline-riser system were sampled at 1 Hz. Select system process variables are presented in TABLE I with their associated units and tags. The experimental data used in this paper can be found in Cranfield Online Research Data (CORD) [36].

C. Ultrasonic sensor data collection

The ultrasonic sensor and algorithm used in this work were those used in [22]. The Doppler ultrasonic sensor consisted of two piezoelectric elements. One of the elements continuously released an acoustic beam into the flow while the other element received the scattered acoustic beam reflected by the moving scatters. The velocity of the fluid was estimated with the frequency shift based on the Doppler effect [37].

A clamp-on non-intrusive CWUD transducer with an excitation voltage of ± 10V, operating at a 500 kHz frequency, was attached at the topside of the S-shaped riser as shown in Fig. 2. The incident angle of the ultrasound beam was 45° with respect to the flow direction on the S-shaped riser. The ultrasonic sensor was placed on the flow pipe at least 10 times the pipe diameter away from tees, valves, and bends to avoid

TABLE I
EXPERIMENTAL PROCESS AND INSTRUMENTATION LIST

| Sensor tags	DESCRIPTION	Unit
PT312	Air delivery pressure	bar
PT403	Top separator pressure	bar
PT408	Riser top pressure	bar
PT417	Riser base pressure	bar
PT501	Three-phase separator pressure	bar
PIC501	Three-phase separator outlet air valve	%
FT102	Inlet water temperature	$^{\circ}$C
FT102	Inlet water density	Kg/m^3
FT102/104	Inlet water flow rate	Kg/s
FT305	Inlet air temperature	$^{\circ}$C
FT305/302	Inlet air flow rate	Sm^3/h
FT404	Top separator gas outlet	M^3/h
FT406	Top separator liquid outlet	Kg/s
LI101	Liquid tank level	m
LI502	Three-phase separator water-oil level	%
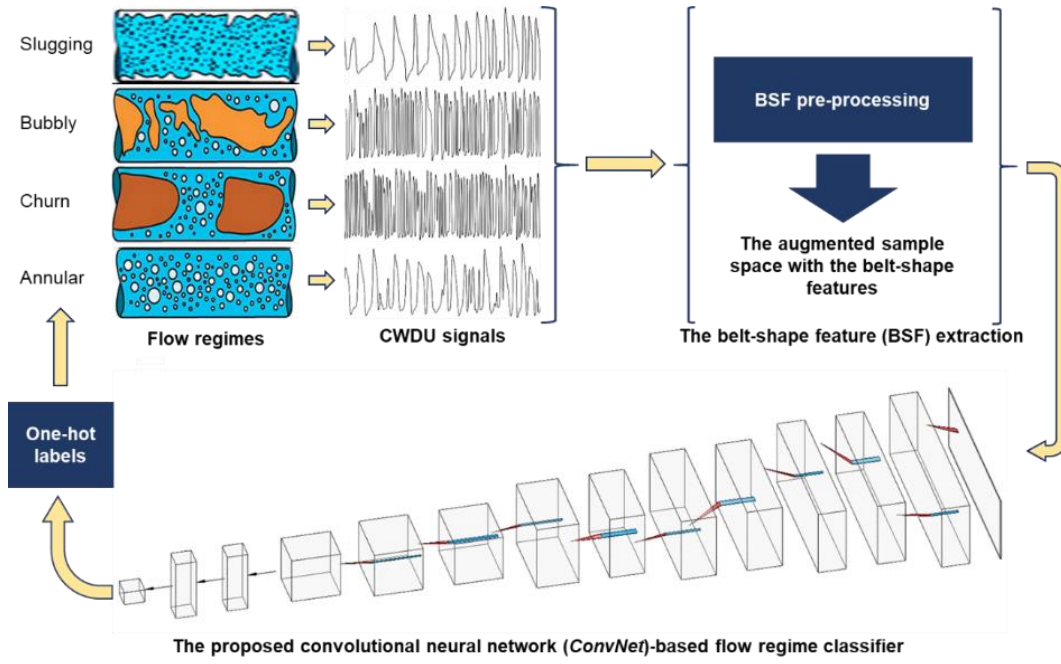
Fig. 3 The overall process of the proposed flow regime classification method.

measurement errors due to cavitation, swirls, and turbulent eddies. Ultrasonic coupling gel was applied between the pipe wall and the Doppler transducer to minimize the loss of ultrasound energy. The electronics of the CWUD flow meter were adapted to record the voltage signals of the Doppler frequency shift.

A LabVIEW data acquisition system was used to acquire the voltage signals of the Doppler frequency shift. The frequency shift was estimated to be less than 2 kHz at the highest flow velocity. A sampling frequency was set at 10 kHz, which satisfied the Nyquist-Shannon sampling theorem [38].

D. Data processing platform

The data processing hardware for this project was a Lenovo workstation. The operating system was Linux Ubuntu 18.04, the CPU was Inter Core i7-7700, the memory was 32GB, and the graphics card was NVIDIA GTX1080. The volume integral classifier was built with Tensorflow 2.2 [39], the one-hot encoding of the data used the Sklearn library [40], and the visualization used Matplotlib [41]. Some experiments and visualizations also used MATLAB.

III. DATA PROCESSING AND THE FLOW REGIME IDENTIFIER

A. Data explanation and overall process

The overall process of the proposed flow regime identification method is illustrated in Fig. 3. The CWDU signals consisted of 125 gas-liquid experiments [36]. Each experiment provided 1.3 million data points during the sampling period, and all experiments were labeled with the ground-truth flow regime. Therefore, the CWDU signals (see Fig. 3) can be arranged as a matrix with 125 rows and 1.3 million columns. Then, the pre-processing block (in Fig. 3) brought in all CWDU signals and conducted the belt-shaped feature (BSF) extraction algorithm to produce the belt-shaped features (BSFs). Notably, the BSF extraction algorithm correspondingly assigned the ground-truth labels to the BSFs. The BSFs were then divided into training, testing, and validation set. The ConvNet circle denotes the proposed ConvNet-based flow regime classifier, which conducted the flow regime classification task using the training and testing sets and was verified using the validation set.

B. Pre-processing based on the belt-shape feature (BSF) extraction

1) The BSF pre-processing algorithm

In this subsection, the new pre-processing, feature-extraction algorithm is described. The algorithm is called "belt-shaped feature" (BSF) for reasons that are explained later in the subsection. The algorithm was inspired by the twin-window strategy adopted in [24], [25], which has verified the twin-window strategy's reliability in augmenting the sample space. In particular, the averaging operation among all Window Bs in the TFE algorithm brings smoothness to the TFE features. Although the smoothness can suppress noise, it can also eliminate the distinctions which limit the performance achievement [24]. The beauty of the deep learning method, however, is its considerable capability for capturing such distinctions while overcoming the noise. Moreover, the TFE algorithm [23] experienced amplitude loss. The new algorithm can eliminate the averaging smoothness and suppress amplitude loss.

Fig. 4 depicts the data pre-processing for the proposed BSF algorithm. The upper-left block depicts the original CWDU data. The expanded waveform Doppler ultrasonic partial signal (Fig. 4) is used to symbolize the CWDU signal. Furthermore, the CWDU data refers to the data points in the sampling period from Section II-A, which includes significant noise.

The first step in the BSF algorithm (the orange block) picks up one of the experimental records corresponding to a single row with 1.3 million recorded data points. The second step (the
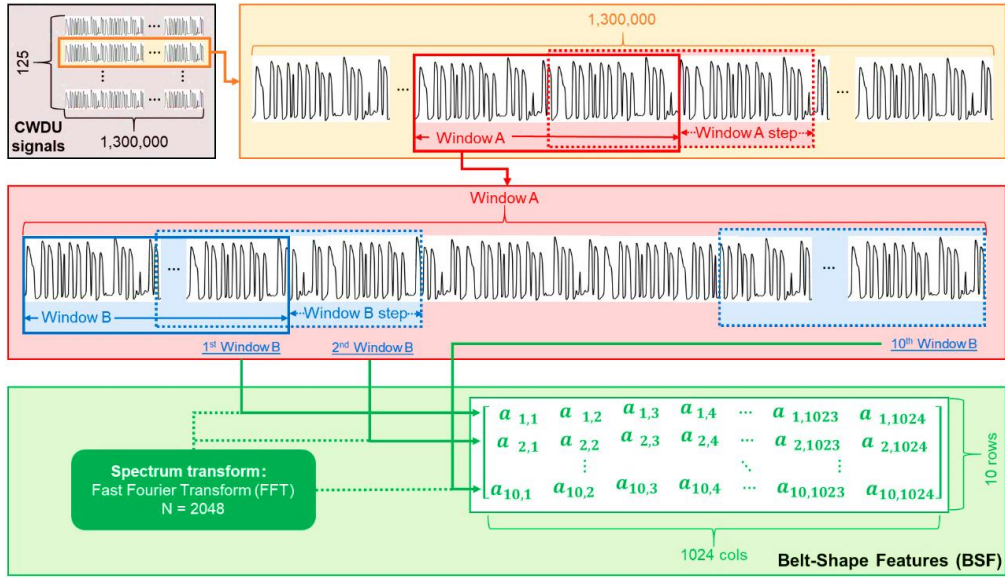
Fig. 4 The pre-processing step based on the belt-shaped feature (BSF) extraction algorithm. The solid arrows refer to the steps of the BSF algorithm. The upper-left square refers to the 125 by 1.3 million data matrix mentioned in Section III-A. The solid and dashed frames refer to the adjoint Window A or B. "Window A", "Window A step", "Window B", and "Window B step" respectively correspond to the length of Window A (l_{wa}), the step of Window A (s_{wa}), the size of Window B (l_{wb}) and the step of Window B (s_{wb}). The "N = 2048" in the bottom square (the green region) refers to the N value for the fast-Fourier-transform (FFT) operation. Only half of the symmetrical FFT result has remained, such that each BSF is a vector with a single row and 1,024 columns.

red block) copies a part of the experimental record by "Window-A," which directly relates to the valid feed-in data for a single BSF. Notably, the BSF is a projection function, which inevitably causes information loss. (1) depicts the numerical relationship between the input data (ϕ_{input}) and the output data (ϕ_{output}), where $\Delta\phi$ refers to the lost information during the projection process. The "Window-A step" is the stride length of Window A shifting, which affects the similarity between the adjacent BSFs (stride refers to the single moving step-length of the convolution kernel). (2) illustrates the numerical relationship between the number of BSFs ($Number_{BSF}$) and the length of Window-A (l_{wa}) and the step length of Window-A shifting (s_{wa}). The third step (the blue block) further copies the Window B region from Window-A. The length of Window B ("Window-B" in Fig. 4) affects the sampling resolution size. This twin-window strategy is used to avoid amplitude loss. The step length of Window B (the "Window-B step") directly relates to the number of rows in the BSF features. (3) characterizes the numerical relationship between the number of rows in a single BSF and l_{wa} (length of Window-A), l_{wb} (length of Window-B), and s_{wb} (step length of Window-B shifting). The step length of Window-B (s_{wb}) must less than l_{wb} so that the overlaps can suppress the amplitude loss of the window boundary.

The fourth and final step (the green block) uses the fast-Fourier-transform (FFT) to transform the signal from the time domain to the frequency domain, and the results of different Window Bs are listed row by row in a two-dimensional (2D) array. This research uses the term "belt-shaped feature" (BSF) because, like a belt, the number of columns in this 2D array is much greater than the number of rows. The BSF algorithm does not adopt the average among Window B to maintain the detailed distinction.

$$\phi_{output} = \phi_{input} - \Delta\phi \qquad (1)$$

$$Number_{BSF} = \frac{l_{CWDU} - l_{wa}}{s_{wa}} \qquad (2)$$

$$Number_{rows} = \frac{l_{wa} - l_{wb}}{s_{wb}} \qquad (3)$$

Note that each CWDU experimental record has been labeled with a corresponding ground-truth flow regime. The BSF algorithm directly assigns the ground-truth flow regimes to the output BSFs. For example, each experimental record can generate n_{wa} BSFs; thus, all of these n_{wa} BSFs used the same flow regime label. This research uses integer values 1, 2, 3, and 4 to label the slug flow, bubbly flow, churn flow, and annular flow, respectively.

2) The experimental design for the BSF pre-processing algorithm

Two modules enact the BSF extraction algorithm as shown in the upper right bracket of Fig. 3. The belt shape features are thus the intermediate result passed to the CNN-based classifier shown in the lower part of Fig. 2. Standard accuracy metrics are used to assess the performance of both the BSF algorithm and the CNN-based classifier. Thus, a separate control variable strategy is required. The size of the BSF indicates the feature size input into the classifier, which can be variant depending on the number of Window-Bs and the *N* value (in FFT) in the BSF algorithm. Hence, we fix the size (number of rows (h) and number of columns (w)) of the BSF (ϕ_{output}) to evaluate the BSF algorithm — this constrains the influence on the metrics of the first model (the BSF algorithm). (2) and (3) are the mathematical definitions. Here the BSF size is fixed to be $h = 10$ (also equal to N_{wb}) and $w = 1024$.

(1) we see that that the BSF algorithm's information loss is a dynamic variable that depends on l_{wa}, s_{wa}, l_{wb}, s_{wb}, and the *N*
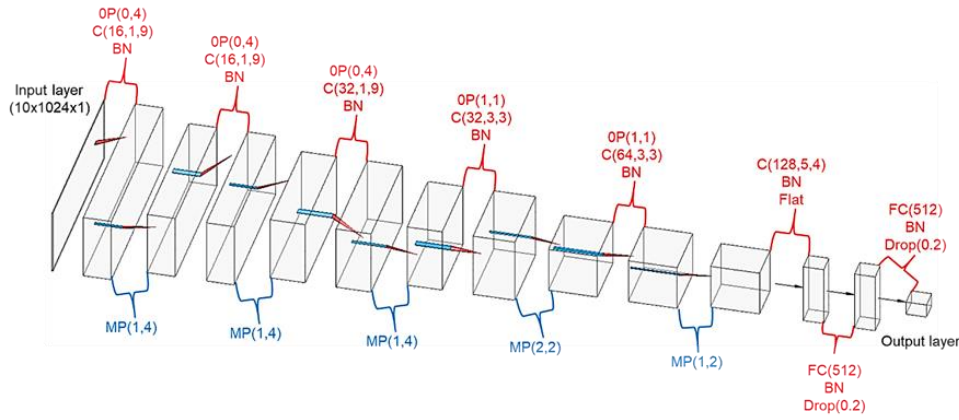
Fig. 5 The ConvNet classifier for flow regime classification. The grey cubes refer to the tensors. The blue cubes refer to the filter spaces. The red rectangular pyramids refer to the convolution or max-pooling layers, which are distinguished using red and blue texts. "*0P(w, h)*" refers to the zero-padding layer with the $w \times h$ padding size, "*C(c, w, h)*" refers to the 2D convolutional layer with the $w \times h$ convolutional kernel size and c channel, and "BN" refers to the batch normalization layer. "*MP(w, h)*" refers to the 2D max-pooling layer with $w \times h$ kernel size. "FC(d)" refers to the fully connected dense layer with d nodes, "Flat" refers to the flatten layer, and "Drop(x)" refers to the dropout layer with dropout ratio x.

value in FFT. TABLE II shows the specific experimental settings. Five values of l_{wa} were used ranging from 8,000 to 50,000 recorded data points. In line with the pipe scale and mixture flow velocity range, the highest frequency of the CWDU signals in this project was less than 2 kHz. According to the Nyquist theorem [38], the sample size must be at least twice (in practices at least four times) that determined by the maximum frequency value (this research used 8,000 recorded data points as the minimum l_{wa}). The TFE algorithm in [24] used 50,000 recorded data points and obtained a superior result, so this research also used 50,000 recorded data points as the maximum l_{wa}. Three values, 1,250, 2,500, and 5,000 recorded data points, were selected for s_{wa}, which corresponded to approximately 15%, 30%, and 60% of the minimum l_{wa}, respectively, while l_{wb} was 3,000, 5,000, and 10,000 recorded data points. The 3,000 recorded data points scheme was the closest value to 2 kHz, with the smallest amplitude loss, but may not have covered a complete cycle. The 5,000 recorded data points scheme experienced a slight amplitude loss, but a complete cycle was covered. The 10,000 recorded data points scheme experienced a larger amplitude loss.

Note that the experimental designs used the controlled variable method. Experiments A1, A2, A3, A4, and A5 belonged to the first group of experiments, which mainly tested the effect of varying l_{wa}. Experiments A5, B1, and B2 belonged to the second group of experiments, which mainly tested the influence of the various values of s_{wa}. Experiments A5, C1, and C2 belonged to the third group of experiments, which mainly tested the influence of the various values of l_{wb}.

C. ConvNet classifier

1) The architecture of the ConvNet classifier

The ϕ_{output} for the entire BSF algorithm (the ϕ_{input} is all CWDU signals from Section II) is a large matrix with N_{BSF} rows and 1,024 columns, called the BSF data space. Each row for the BSF data space corresponds to an input sample for the deep learning classifier. The corresponding ground-truth labels (mentioned in Section III-B-1)) adopt the one-hot encoding

format, which can avoid intermediate value confusion (such as prediction A5 for labels one and two). The multi-classification task essentially transforms into four binary classification tasks.

Fig. 5 depicts the ConvNet-based classifier's architecture, which was designed in terms of the 2D BSFs with a low height-width ratio. It is clear that the existing ConvNet-based classifications (such as ResNet [42] and InceptionNet [43]) are not suitable for the BSFs. This research proposes a new ConvNet-based classifier, which interactively cooperates with 1D ConvNets, 2D ConvNets, and fully connected networks (FcNets). The ConvNet classifier adopts the supervised learning strategy, which inputs the BSFs (Section III-B-1)) and then compares the predictions to the one-hot-formed flow regime labels for gradient descent.

In this paper, the multidimensional data array, or tensor, processed inside the ConvNet-based classifier is 4-dimensional (order 4), with dimension sizes batch, b, height, h, width, w, and depth, d. Thus, the tensor has dimension $b \times h \times w \times d$. The input tensor explained as an example in this section has dimension $128 \times 10 \times 1024 \times 1$. A batch-based gradient descent, which corresponds to the first dimension of the input tensor, is used. The batch-based strategy can significantly decrease unit-computational consumption and avoid the problem of local minima. The batch dimension comes from a concatenate operation, which groups multiple BSFs together and increases the dimension of the first axis, b. The second and third dimensions, h and w correspond to the same orientation as the convolutional kernel's height and width. The tensor height value of 10 is chosen as a suitable control variable to cope with the signal length and maintain the distinction (Section Section III-B-1)). The final depth value depends on the corresponding depth (the number of channels) of the convolutional kernel within the ConvNets.

The first part of the ConvNet-based flow regime classifier consists of six convolutional blocks and an FcNets block. Each convolutional block further consists of a 2D zero-padding layer (Zerop2D), a 2D convolutional layer (Conv2D), a "relu" activation layer (ReLU), a batch-normalized layer (BN), and a
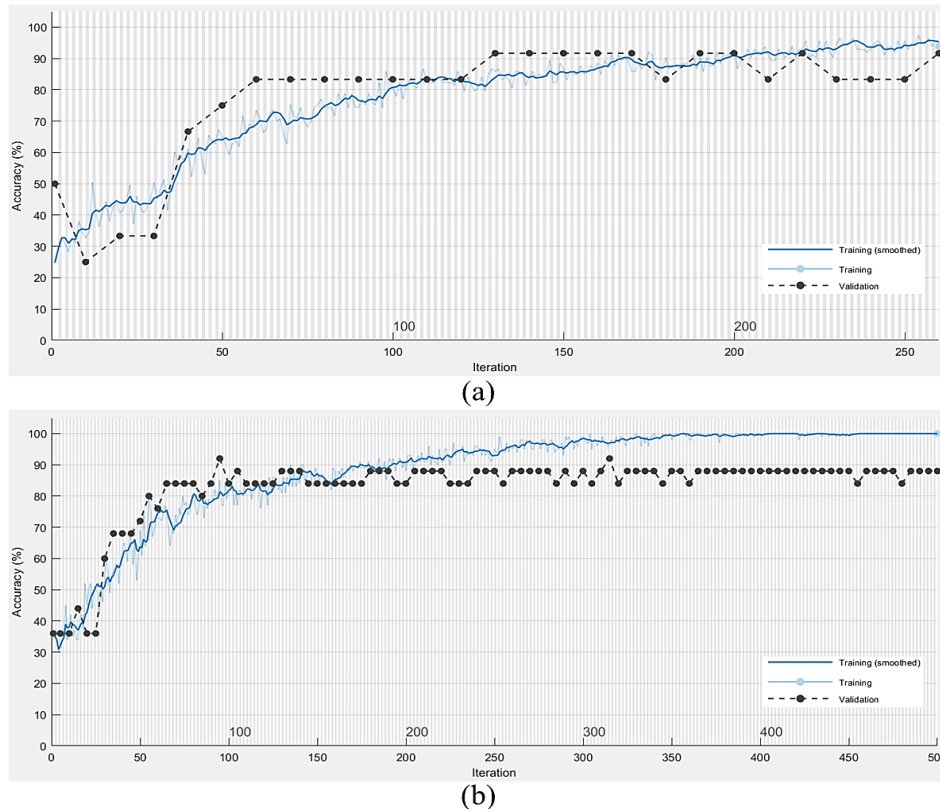
Fig. 6 The training and testing accuracy curves of Experiments E1 and E2 (TABLE III) in (a) and (b) respectively. The light blue curves give the real-time training accuracy without local smoothing. The adopted batch gradient descent can cause a highly oscillating curve, so the smoothed curves (the dark blue curves) depict the general trend instead of the oscillation. The black dots-curves refer to the testing accuracy. The x and y axes respectively refer to iteration epochs and accuracy.

2D max-pooling layer (Maxp2D). The parameter-sharing property of ConvNets can extract a compacted feature without a significant increase in computational consumption, then input to the FcNets block. The FcNets block contains two hidden layers, and each hidden layer consists of a fully connected layer (512 neurons), a ReLU, and a dropout layer. The proposed classifier's output layer is the fully connected layer (four neurons) with the "softmax" activation.

Notably, the in/out tensor dimensions of 1D ConvNets, 2D ConvNets, and FcNets respectively correspond to three, four, and two. Therefore, this research adjusted the kernel height to simulate the 1D ConvNets via 2D ConvNets. For example, the 2D convolutional kernel of 1×4 could replace the 1D convolution kernel of 4. This replacement could bypass the complicated tensor dimension change. Specifically, the kernel height and width in this research separately correspond to 1×9, 1×9, 1×9, 3×3, 3×3, and 5×4, and their kernel depth separately corresponds to 16, 16, 32, "32", 64, and 128 (the red notations in Fig. 5). The blue values in Fig. 5 depict the max-pooling layer, which separately utilizes 1×4, 1×4, 1×4, 2×2, and 1×4 as the max-pooling strides. It is clear that the tensor height and width progressively decrease while the tensor depth increases, which can be understood as a process by slowly transferring the 2D information into the depth axis. The output of the ConvNet-based block then inputs to the FcNets after flattening. Notably, the dropout layer uses a ratio of 0.2, which deactivates 20% of neurons in the fully connected layer. This can be understood to add extra random noises into

the neural network to improve its general-ability and suppress the overfitting risk [44].

2) The hyperparameters of the proposed ConvNet classifier

The loss function used in this research was the categorical cross-entropy. The optimizer was the Adam optimizer, the learning rate was 0.0001, the exponential decay rate for the first and second moment estimates were respectively 0.9 and 0.999, the epsilon value was 0.001, and the decay value was 0.0001. The training batch size was 128 samples. The division ratio of the training set, testing set, and validation set was 6:2:2.

Three precautions to guarantee the reliability of the results in Section II were adopted. Firstly, the BSF algorithm could produce more than 30k samples for the ConvNet training so that the training, testing, and validation sets respectively assigned more than 18k, 6k, and 6k samples. It was a sufficient sample size for the proposed classifier, being over 35 times the number of neurons in FcNets. Secondly, a random shuffle to the BSF sample space was adopted, guaranteeing high robustness to the potential data skew. Finally, the loss was weighted by a scalar coefficient list to suppress the data skew, which was calculated according to the categorical weights. Furthermore, only one result is displayed, which is located in the converge range. The aim was to achieve the best model for varying BSF schemes rather than creating a highly robust ConvNet-based classifier structure. Moreover, the statistic records can bring ambiguity to the metrics. The only two conditions for the deep learning model are converged and non-converged. The non-converged model is not fit for the target. For the converged condition, the
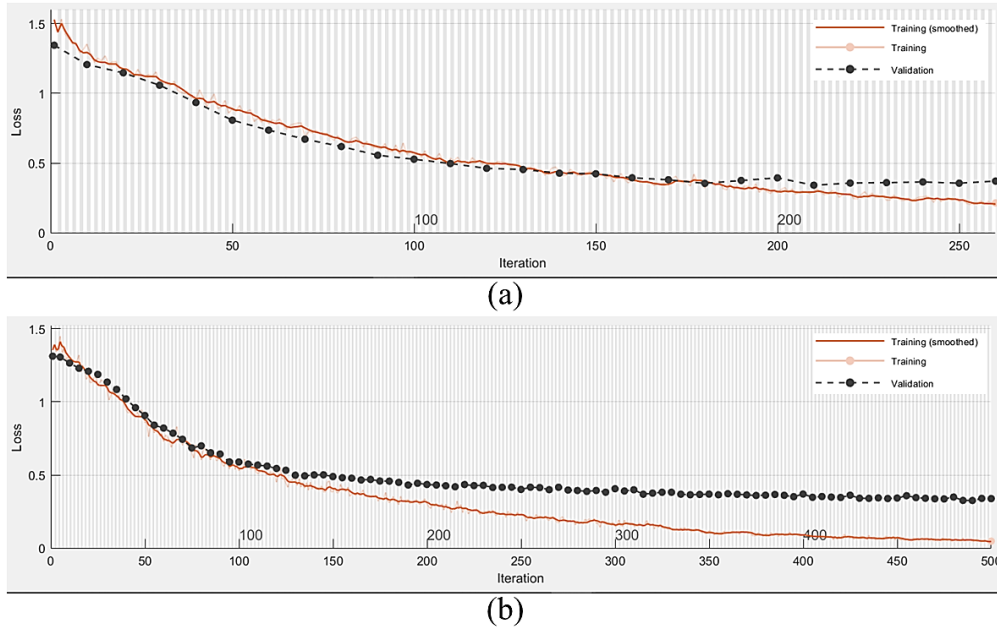
Fig. 7 The training and testing loss curves of Experiment E1 and Experiment E2 (TABLE III) in (a) and (b) respectively. The light orange curves depict the real-time loss without local smoothing. The dark orange curves are the smoothed loss. The black dot curves refer to the testing loss. The x and y-axis respectively refer to iteration epochs and categorical cross-entropy loss.

model can converge to either a local minimum or the global minimum. A local minimum performance is worse than the global minimum, so it is not necessary to list these results.

IV. RESULTS AND DISCUSSION

A. Test without pre-processing

This research utilizes the comparative experiments, Experiments E1 and E2 (without the BSF pre-processing), to justify the BSF pre-process necessity. A CWDU signal with a single row and A3 million columns is segmented into 130 segments according to 10,000 recorded data points. The 130 segments of data are spliced row by row into a 2D array with 130 rows and 10,000 columns. The FFT is adopted to each row with N_{fft} equaling 2,048 so that the CWDU becomes a 2D array with 130 rows and 1,024 (the symmetry of FFT) columns, known as the 2D reshaped CWDU data space in this research. It is clear that the 2D reshaped CWDU data space has no data augmentation, so this research only divides it into two subsets: the training set and the testing set. Admittedly, excluding the validation set significantly decreases the reliability of deep learning training. Fig. 6, Fig. 7, and Fig. 8 respectively depict the training and testing accuracy curves, the loss curves, and the confusion matrices, where (a) and (b) respectively correspond to Experiments E1 and E2.

Fig. 6 (a) and Fig. 7 (a) illustrate that the testing loss of Experiment E1 converged at around 180 epochs, with the training accuracy curve and loss curve still under the learning track. The training and testing accuracy respectively arrived at 96% and 91.70%, as illustrated in Fig. 6 (a). Fig. 8 (a) shows that Experiment E1 achieved a superior flow regime identification performance, while TABLE III illustrates that Experiment E1 achieved better performance than [22]. Fig. 6 (b), Fig. 7 (b), and Fig. 8 (b) depict another result by adopting a different training-testing set ratio (Experiment E2). TABLE

III shows that Experiment E2 respectively achieved 100% and 88% for training and testing accuracy, which improves on [22] but is worse than Experiment E1. It is interesting that only changing the 10% data segmentation ratio causes such a significant change between Experiments E1 and E2. Technically, the subset strategy must guarantee each subset can represent the sample distribution over the whole data space [24], [25]. However, Experiments E1 and E2 reveal that the 2D reshaped CWDU data space (without pre-processing) cannot represent the sample distribution. Further evidence of the low representability is that the accuracy curves in Fig. 6 display a significant discontinuous form. Hence the notation of accuracy resolution (see (4)) is proposed to quantitively evaluate the result's reliability for all the experiments in this research, where $R_{accuracy}$, N_{sample}, and η_{train} respectively refer to the accuracy resolution, the number of samples, and the testing set ratio over the entire dataset. Experiments E1 and E2 respectively hold 8% and 4% as $R_{accuracy}$.

$$R_{accuracy} = \frac{100\%}{N_{sample} \times \eta_{train}} \quad (4)$$

B. Experiments and analysis with the BSF pre-processing data

TABLE III depicts the experimental results from TABLE II along with the comparison results. The first two rows refer to the comparative experimental results without pre-processing, and the last three rows list the state-of-the-art results from [22], [24], [25]. Experiments A1, A2, A3, A4, A5, B1, B2, C1, and C2 correspond to the experiments in TABLE II. The highest training accuracy was 97.40% (Experiment A5), and the lowest training loss was 0.0704 (Experiment A5). The highest testing accuracy was 94.57% (Experiment B2) and the lowest testing loss 0.1423 (Experiment B2). The highest validation accuracy was 94.94% (Experiment B2) and the lowest validation loss 0.0720 (Experiment A3).

The results of Experiments A1, A2, A3, A4, A5, B1, C1, and C2 are presented in the Appendix and respectively correspond to Fig. 10–Fig. 17. Note that this research conducted training three to five times and determined an approximate converge range first, then recorded an extra experimental record, located in the converged range. Therefore, the results in TABLE III are based on statistical analysis.

Fig. 10 presents the results of Experiment A1. The accuracy curves (Fig. 10 [a]) and the loss curves (Fig. 10 [b]) demonstrate that the converging of Experiment A1 reached about 45 epochs. The curves in Fig. 10 are continuous compared with Experiment E1 and E2. The BSF pre-processing algorithm augmented the data space to 64,500 samples for Experiment A1, which was 512 times larger than Experiments E1 and E2. Notably, $R_{accuracy}$ was 0.0078% for Experiment A1. Fig. 10 (c) and (d) depict the testing and validation confusion matrix. The identification accuracy of flow regimes 1 and 3 was about 75%. The length of Window A was 10,000 recorded data points for Experiment A1, which caused insufficient information feed-in to the BSF algorithm. Therefore, the experiment group Ax tested the influence with various l_{wa}. Experiments A2, A4, and A5 used a wider Window A, and Experiment A3 used a narrower Window A.

Fig. 11 illustrates that the accuracy of Experiment A2 was slightly improved, and the loss was slightly reduced. Fig. 11 (a) and (b) show that the classifier reached the converged point in about 45 epochs. However, compared with the result of Experiment A1, the training and testing curves of Experiment A2 are smoother, indicating that the increased information feed-in in Window A made the model more stable. $R_{accuracy}$ was also 0.0078% for Experiment A2. Moreover, Fig. 11 (c) and (d) illustrate that the classification accuracy of classes 1 and 3 improved. This indicates that increasing the information feed-in into the BSF algorithm improves classification accuracy.

Fig. 12 depicts the results of Experiment A3. Experiment A3 used the shortest Window A in TABLE II (8,000 recorded data points). However, Fig. 12 (a) and (b) still converge around the 45th epoch, which suggests that increasing the amount of information feed-in to the BSF algorithm did not make the model converge faster. However, TABLE III and Fig. 12 (c) and (d) demonstrate that reducing the length of Window A greatly reduced the classifier's accuracy, verifying the assumption in Section III that Window A's length was directly

related to the BSF's information capacity. Notably, $R_{accuracy}$ equaled 0.008% for Experiment A2.

Fig. 13 depicts the results of Experiment A4. Experiment A4 used the l_{wa} of 50,000 recorded data points, which equals the l_{wa} in the TFE algorithm. Fig. 13 (a) and (b) display significant smoothed curves. However, the model still reached the best training point at 45 epochs, which demonstrates that increasing l_{wa} did not speed up the convergence speed. The information capacity of the BSF feature mainly affected the smoothness of the training curve. $R_{accuracy}$ equaled 0.008% for Experiment A4. Fig. 13 (c) and (d) depict that the flow regime's accuracy was improved, suggesting that the BSF feature's performance increased as more information fed in.

Fig. 14 depicts the results of Experiment A5. Increasing l_{wa} also increased the challenge of the frequency domain transformation by Window B, which led to a higher risk of amplitude loss. Therefore, it was necessary to balance l_{wa} and amplitude loss. Experiment A5 used an l_{wa} of 30,000 recorded data points, which was between the values in Experiments A2 and A4. Fig. 14 (a), (b), (c), and (d) illustrate that Experiment A5 achieved results similar to Experiment A4 while adopting an l_{wa} of 30,000 recorded data points. Therefore, this research uses Experiment A5 as the reference experiment for Experiments Bx and Cx. Notably, $R_{accuracy}$ was 0.0078% for Experiment A5.

Fig. 15 illustrates the results of Experiment B1. Experiments A5, B1, and B2 focused on the influence of different s_{wa} on the BSF algorithm. Experiment B1 used twice the value of s_{wa} of Experiment A5. Fig. 15 (a) and (b) illustrate that the best training point was reached at the 25th epoch, while Fig. 15 (c) and (d) depict that the accuracy of Experiment B1 slightly decreased. According to (1), changing s_{wa} directly affects the number of samples generated by the BSF algorithm, which leads to a high risk of overfitting. $R_{accuracy}$ was 0.0157% for Experiment B1. This research further generated more samples in Experiment B2 to test the impact of the number of samples for training classifiers.

Fig. 9 displays the results of Experiment B2, which yielded the best results in this research. The l_{wa} used in Experiment B2 was 1,250 recorded data points so that the BSF algorithm could augment 127,000 samples. Fig. 9 (a) and (b) display the ideal training curves, which converge at approximately the 35th epoch. Fig. 9 (c) and (d) illustrate that the accuracy of

TABLE II
THE HYPER-PARAMETERS SETTINGS OF THE EXPERIMENTS FOR THE BSF ALGORITHM

index	l_{wa}	s_{wa}	N_{wa}	l_{wb}	s_{wb}	N_{wb}	N_{BSF}
unit	recorded data points		n/a	recorded data points		n/a	n/a
A1	**10,000**	2,500	516	5,000	500	10	64,500
A2	**20,000**	2,500	512	5,000	1,500	10	64,000
A3	**8,000**	2,584	500	5,000	300	10	62,500
A4	**50,000**	2,500	500	5,000	4,500	10	62,500
A5	**30,000**	2,540	500	5,000	2,500	10	62,500
B1	30,000	**5,000**	254	5,000	2,500	10	31,750
B2	30,000	**1,250**	1,016	5,000	2,500	10	127,000
C1	30,000	2,540	500	**3,000**	2,700	10	62,500
C2	30,000	2,540	500	**10,000**	2,000	10	62,500

The number l_{wa} refers to the length of Window A, s_{wa} refers to the length of the Window A step, N_{wa} refers to the number of Window A. Similarly, l_{wb} refers to the size of Window B, s_{wb} refers to the length of the Window B step, N_{wb} refers to the amount of Window B. The term N_{BSF} refers to the number of the final BSF features. The bold values are the control variables for the corresponding experiments
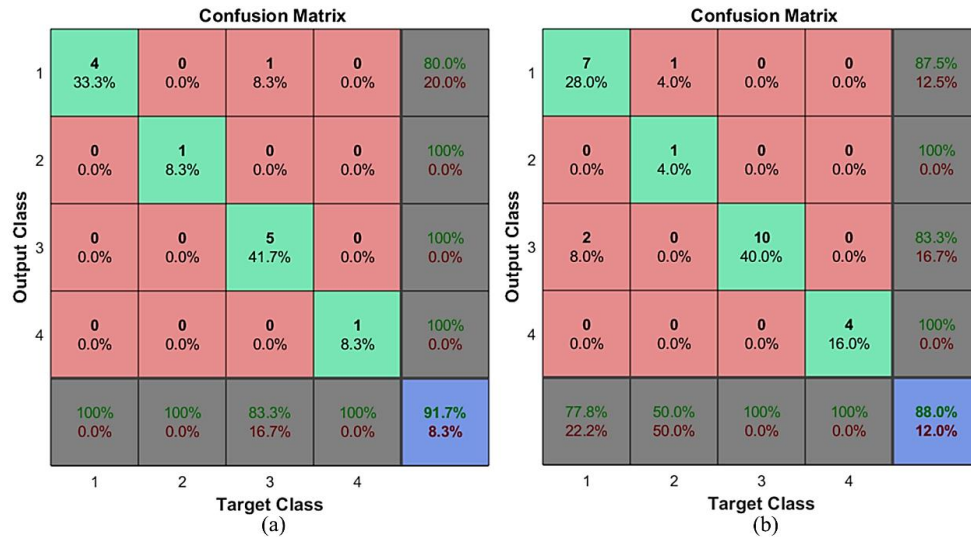
Fig. 8 The confusion matrix of Experiment E1 and Experiment E2 (TABLE III) in (a) and (b) respectively. The green squares refer to the true-positive predictions. The red squares refer to the false-positive, true-negative, and false-negative predictions. The grey squares refer to the average for each row and column. The green values refer to the average true-negative or false-positive predictions, and red values refer to the average false-positive or true-negative predictions. The blue squares refer to the overall average predictions. The green values refer to the average true-positive predictions, and the red values refer to the average false-negative predictions.

Experiment B2 was greatly improved. TABLE III further reveals that the testing accuracy, the testing loss, and the validation accuracy were the best in Experiment B2. This suggests that the BSF algorithm-based augmentation considerably improved the performance of the ConvNet classifier. Notably, $R_{accuracy}$ was 0.0157% for Experiment B2.

Fig. 16 illustrates the results of Experiment C1. Experiment C1 reduced the size of Window B to 3,000. The accuracy curves of Fig. 16 (a) demonstrate that the control variable of Experiment C1 hardly affected the accuracy. However, TABLE III and Fig. 16 (b) show that the converged loss of Experiment C1 was three times larger than Experiment A5. This suggests that the training in Experiment A5 was overfitted. Although Experiment A5 had the highest training accuracy, the generalization ability of the model was relatively weak. Experiment C1 reached convergence with only 14 epochs, which indicates that the low amplitude leakage caused by the

small Window B size increased the model's learning ability, thereby enhancing the robustness and generalizability.

Fig. 17 illustrates the results of Experiment C2. The length of Window B in Experiment C2 was 10,000 recorded data points larger than 2 kHz. A single Window B produced substantial amplitude loss, leading to a decrease in accuracy and an increase in loss. However, TABLE III demonstrates that the accuracy and loss of Experiment C2 were still excellent, and Fig. 17 (a) and (b) illustrate that Experiment C2 reached the best training point in the 25th epoch, which was better than Experiments Ax and Bx. Fig. 17 (c) and (d) also depict better results than Experiment Ax. This was due to the sliding window strategy; the next window captured the amplitude lost in a single window. Therefore, the BSF algorithm is a highly flexible algorithm, which can flexibly adjust its parameters according to the situation and achieve superior results compared to existing methods [22], [24], [25] (especially with

TABLE III
THE EXPERIMENTAL RESULTS OF THE PROPOSED CONVNET-BASED AND COMPARISON CLASSIFIERS

index	acc_{train}	$loss_{train}$	acc_{test}	$loss_{test}$	acc_{valid}	$loss_{valid}$
E1	96.00%	n/a	91.70%	n/a	n/a	n/a
E2	100.00%	n/a	88.00%	n/a	n/a	n/a
A1	90.14%	0.2404	85.68%	0.3764	81.17%	0.5663
A2	91.86%	0.1963	89.66%	0.2458	88.95%	0.2815
A3	87.08%	0.3155	83.20%	0.4106	80.06%	**0.0720**
A4	95.37%	0.1120	91.21%	0.2543	91.65%	0.2389
A5	**97.40%**	**0.0704**	91.78%	0.2491	92.32%	0.2417
B1	93.70%	0.1572	89.00%	0.2840	90.00%	0.2606
B2	96.28%	0.0952	**94.57%**	**0.1423**	**94.94%**	0.1287
C1	91.11%	0.2113	90.14%	0.2213	92.14%	0.1897
C2	92.68%	0.1865	90.34%	0.2279	91.28%	0.2100
[22]	85.70%	n/a	84.60%	n/a	n/a	n/a
[24]	99.01%	n/a	96.28%	n/a	96.35%	n/a
[25]	95.96%	0.1250	96.26%	0.0903	96.42%	0.0887

"Index" refers to the corresponding experimental index; the first number indicates the experiment group. The bold values refer to the best results among all experiments using the BSFs. "acc_{train}" refers to the training accuracy. "$loss_{train}$" refers to the training loss using the categorical cross-entropy. "acc_{test}" refers to the testing accuracy. "$loss_{test}$" refers to the testing loss using the categorical cross-entropy. "acc_{valid}" refers to the validation accuracy. "$loss_{valid}$" refers to validation loss. Rows denoted "[22]" and "[24]" respectively refer to the advanced results of Nnabuife et al. (2019) [22] and Nnabuife et al. (2020) [24]. "[25]" refers to the CNN-based result in the flow regime identification benchmark proposed by Kuang et al. (2020) [25].
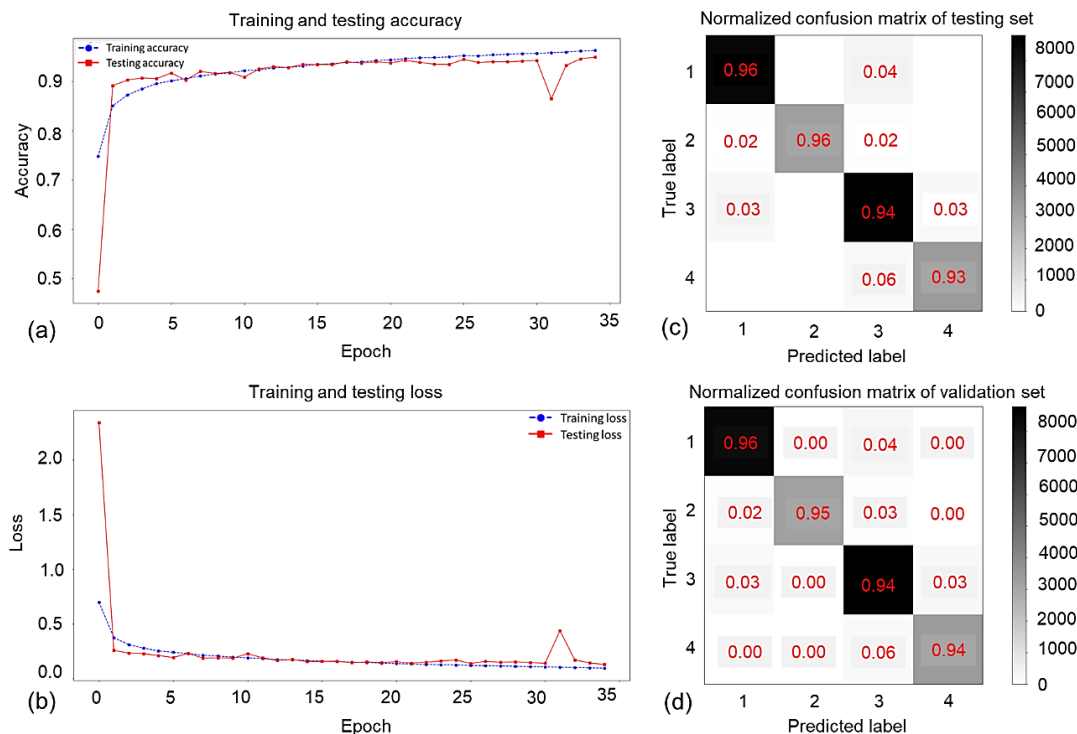
Fig. 9 The training, testing, and validation results of the Experiment B2 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The red solid lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.

limited computing power).

Notably, the BSF pre-processed experiments all achieved better performance than [22] but worse than [24] and [25]. However, the BSF used a smaller size compared to the TEF in [24], which simplifies the computational graph of the BSF-based ConvNet classifier. Furthermore, the best result in this research (Experiment B2) was progressively good; the call-back point was around 33 epochs. Correspondingly, [24] conducts more than 1,200 epochs and [25] conducts 300 epochs.

V. CONCLUSION

The BSF algorithm provides a new idea for one-dimensional signal feature coding through the belt-shaped configuration. Compared with the existing state-of-the-art methods, the results of the two-phase flow ultrasonic signal classification experiments suggest that BSF features are more flexible, robust, and generalized. It not only retains the flexibility of the twin-window signal processing method but further enhances the feature detail coding ability. It provides a more reliable and rich training environment for subsequent neural network classifiers.

The ConvNet implementation demonstrated the ability to identify key distinct features via a learning process that extracts relevant features from the raw 2D matrix processed through a ConvNet that consists of convolution/batch normalization/rectification/down-sampling layers. These features can then be used to separate data into distinct classes (in this case, slug, bubbly, churn, and annular flows) through a so-called fully connected neural layer. The confidence level of classification may be ascertained from the class probability

vectors produced by the final SoftMax layer.

In this work, a method using a Doppler ultrasonic sensor, BSF features, and ConvNets was developed for the objective classification of gas-liquid two-phase flow regimes. Using a convolution neural architecture, the features extracted were classified into one of the four flow regime categories: bubbly, slug, churn, and annular flow regimes. To improve the performance of the neural network flow regime classifier, the number of inputs or features and hidden neurons was increased with caution to avoid overfitting. In addition, the initial network biases and weights of the network were tuned accordingly. This approach demonstrates the applicability of ConvNets in flow regime classification for industrial applications using a clamp-on non-intrusive ultrasonic sensor.

VI. FURTHER WORK

As a sequel to this present work, we are developing a real-time identification of gas-liquid flow regime transitions using a non-invasive ultrasonic sensor, belt-shape features, and convolutional neural network in an s-shaped riser. In this work, different flow regime transitions will be dealt with, for example, flow transition from bubbly flow to slug flow.

Moreso, flow regime identification in an S-shaped riser using different image processing methods will be unraveled.
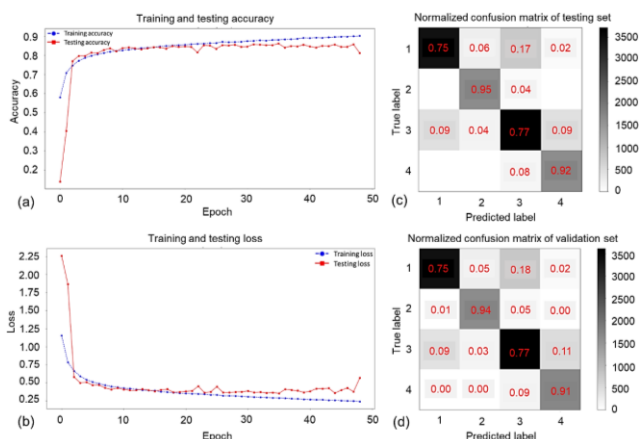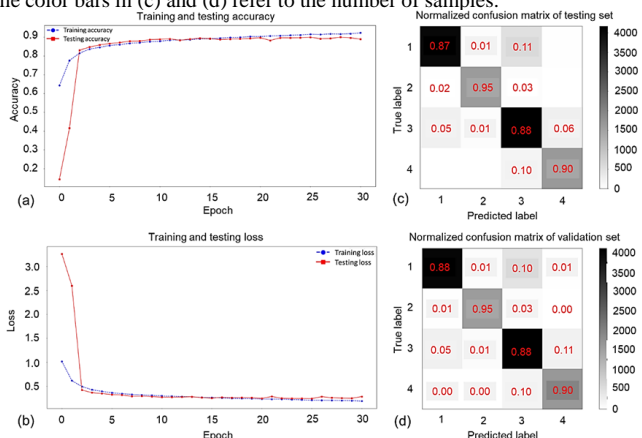
VII. APPENDIX

Fig. 10 The training, testing, and validation results of the Experiment A1 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.
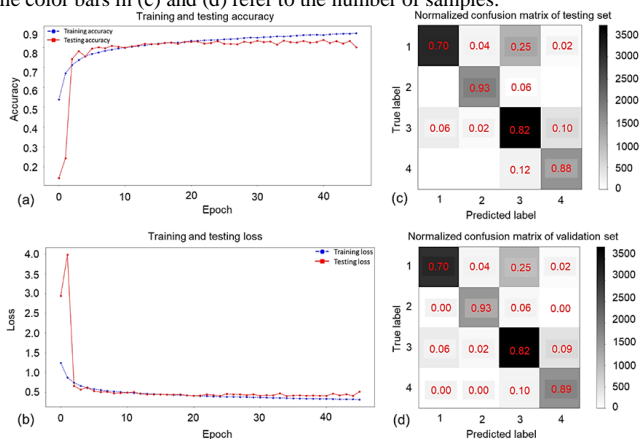

Fig. 11 The training, testing, and validation results of the Experiment A2 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.
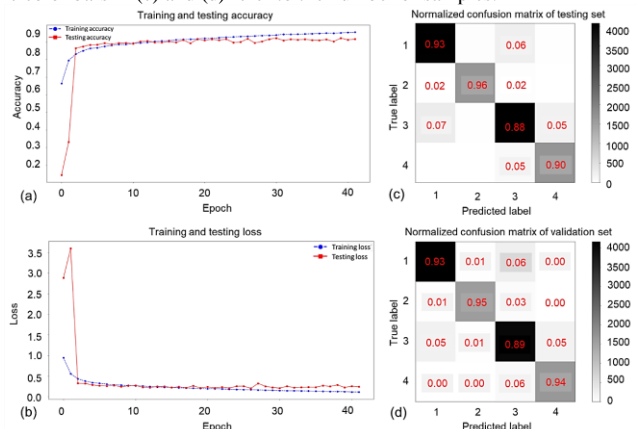

Fig. 12 The training, testing, and validation results of the Experiment A3 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing

process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.

Fig. 13 The training, testing, and validation results of the Experiment A4 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.
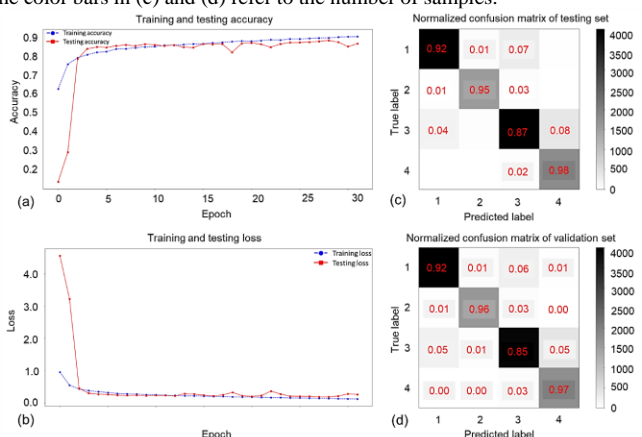

Fig. 14 The training, testing, and validation results of the Experiment A5 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.
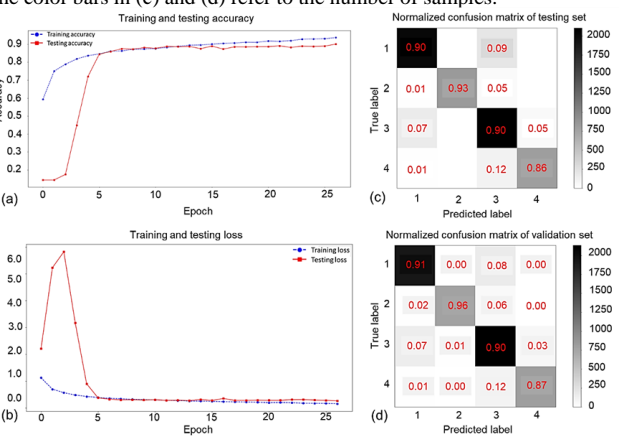

Fig. 15 The training, testing, and validation results of the Experiment B1 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing

process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the number of samples.
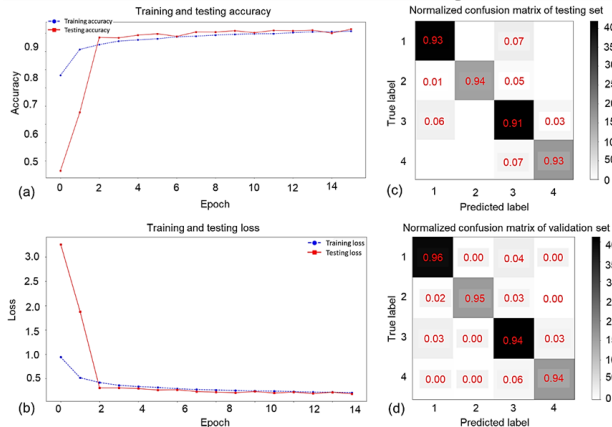

Fig. 16 The training, testing, and validation results of the Experiment C1 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the amount of sample.
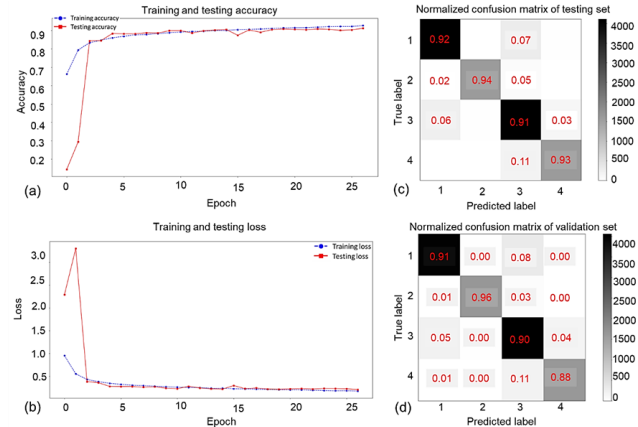

Fig. 17 The training, testing, and validation results of the Experiment C2 hyperparameter of BSF features. (a) refers to the training and testing accuracy chart, (b) refers to the training and testing loss chart, (c) refers to the normalized confusion matrix of the testing set, and (d) refers to the normalized confusion matrix of the validation set. The solid red lines in (a) and (b) refer to the testing process, while the blue dashed lines in (a) and (b) refer to the training process. The color bars in (c) and (d) refer to the amount of sample.

VIII. REFERENCES

[1]   M. Firouzi and S. Hashemabadi, "Analytical solution for newtonian laminar flow through the concave and convex ducts," *J. Fluids Eng.*, vol. 131, pp. 1–6, 2009.

[2]   L. Cheng, G. Ribatski, and J. R. Thome, "Two-phase flow patterns and flow-pattern maps: fundamentals and applications," *Appl. Mech. Rev.*, vol. 61, no. 5, p. 050802, 2008.

[3]   K. Muvvala, V. Kumar, B. C. Meikap, and S. Chakraborty, "Development of soft sensor to identify flow regimes in horizontal pipe using digital signal processing technique," *Ind. Eng. Chem. Res.*, vol. 49, no. 6, pp. 3001–3010, 2010.

[4]   A. Ogazi, Y. Cao, H. Yeung, and L. Lao, "Slug control with large valve openings to maximize oil production," *SPE J.*, vol. 15, no. 3, pp. 8–11, 2010.

[5]   M. Al-naser, M. Elshafei, and A. Al-sarkhi, "Artificial neural network application for multiphase flow patterns detection : A new approach," *J. Pet. Sci. Eng.*, vol. 145, pp. 548–564, 2016.

[6]   M. Alssayh, A. Addali, D. Mba, and T. Dao, "Identification of two phase flow regime using acoustic emission technology," *Int. J. Mech.*

Prod. Eng., vol. 1, no. 6, pp. 27–31, 2013.

[7]   R. Hanus, M. Zych, M. Kusy, M. Jaszczur, and L. Petryka, "Identification of liquid-gas flow regime in a pipeline using gamma-ray absorption technique and computational intelligence methods," *Flow Meas. Instrum.*, vol. 60, no. September 2017, pp. 17–23, 2018.

[8]   B. K. Arvoh, R. Hoffmann, and M. Halstensen, "Estimation of volume fractions and flow regime identification in multiphase flow based on gamma measurements and multivariate calibration," *Flow Meas. Instrum.*, vol. 23, no. 1, pp. 56–65, 2012.

[9]   R. Hanus, "Application of the Hilbert Transform to measurements of liquid-gas flow using gamma ray densitometry," *Int. J. Multiph. Flow*, vol. 72, pp. 210–217, 2015.

[10]   Y. Zhao, Q. Bi, and R. Hu, "Recognition and measurement in the flow pattern and void fraction of gas-liquid two-phase flow in vertical upward pipes using the gamma densitometer," *Appl. Therm. Eng.*, vol. 60, no. 1–2, pp. 398–410, 2013.

[11]   S. Fan and T. Yan, "Two-phase air-water slug flow measurement in horizontal pipe using conductance probes and neural network," *IEEE Trans. Instrum. Meas.*, vol. 63, no. 2, pp. 456–466, 2014.

[12]   E. D. Übeyli and I. Güler, "Improving medical diagnostic accuracy of ultrasound Doppler signals by combining neural network models," *Comput. Biol. Med.*, vol. 35, no. 6, pp. 533–554, 2005.

[13]   D. Kouam, J. Girault, and J. Remenieras, "High resolution processing techniques for ultrasound doppler velocimetry in the presence of colored noise . Part II : Multiplephase pipe-flow velocity measurement," *IEEE Trans. Ultrason. Ferroelec. Frequ. Control*, vol. 50, no. 3, pp. 267–278, 2003.

[14]   E. S. Rosa, R. M. Salgado, T. Ohishi, and N. Mastelari, "Performance comparison of artificial neural networks and expert systems applied to flow pattern identification in vertical ascendant gas-liquid flows," *Int. J. Multiph. Flow*, vol. 36, no. 9, pp. 738–754, 2010.

[15]   J. L. Goncalves, T. A. Paiva, R. D. M. Carvalho, O. J. Venturini, and U. F. De Itajuba, "Application of the ultrasonic technique for monitoring and measuring the phase fractions of liquid-gas-solid mixtures," in *Brasil Offshore*, 2011, p. SPE-143153-MS.

[16]   S. Cai, H. Toral, J. Qiu, and J. Archer, "Neural network based objective flow regime identification in air-water two phase flow," *Can. J. Chem. Eng.*, vol. 72, no. 3, pp. 440–445, 1994.

[17]   Y. Mi, M. Ishii, and L. H. Tsoukalas, "Vertical two-phase flow identification using advanced instrumentation and neural networks," *Nucl. Eng. Des.*, vol. 184, no. 2–3, pp. 409–420, 1998.

[18]   Y. Mi, M. Ishii, and L. H. Tsoukalas, "Flow regime identification methodology with neural networks and two-phase flow models," *Nucl. Eng. Des.*, vol. 204, no. 1–3, pp. 87–100, 2001.

[19]   J. Y. Lee, M. Ishii, and 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.*, vol. 51, no. 13–14, pp. 3442–3459, 2008.

[20]   J. Enrique, L. Yang, P. Sidharth, and M. Ishii, "Upward vertical two-phase flow local flow regime identification," *Nucl. Eng. Des.*, vol. 238, pp. 156–169, 2008.

[21]   T. Tambouratzis and I. Pázsit, "Non-invasive on-line two-phase flow regime identification employing artificial neural networks," *Ann. Nucl. Energy*, vol. 36, no. 4, pp. 464–469, 2009.

[22]   S. G. Nnabuife, K. E. S. Pilario, L. Lao, Y. Cao, and M. Shafiee, "Identification of gas-liquid flow regimes using a non-intrusive Doppler ultrasonic sensor and virtual flow regime maps," *Flow Meas. Instrum.*, vol. 68, no. October 2018, 2019.

[23]   E. N. Eyo, K. E. S. Pilario, L. Lao, and G. Falcone, "Development of a Real-Time Objective Gas–Liquid Flow Regime Identifier Using Kernel Methods," *IEEE Trans. Cybern.*, vol. 51, no. 5, pp. 2688–2698, May 2021.

[24]   S. Godfrey Nnabuife, B. Kuang, J. F. Whidborne, and 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," *Chem. Eng. J.*, vol. 403, no. July 2020, p. 126401, 2020.

[25]   B. Kuang, S. G. Nnabuife, and Z. Rana, "Pseudo-image-feature-based Identification Benchmark for Multi-phase Flow Regimes," *Chem. Eng. J. Adv.*, p. 100060, Dec. 2020.

[26]   J. Zhang, "Improved on-line process fault diagnosis through information fusion in multiple neural networks," *Comput. Chem. Eng.*, vol. 30, no. 3, pp. 558–571, 2006.

[27]   N. Bhat and T. J. McAvoy, "Use of neural nets for dynamic modeling and control of chemical process systems," *Comput. Chem. Eng.*, vol. 14, no. 4–5, pp. 573–582, 1990.

[28] W. Yu, C. Zhao, and B. Huang, "MoniNet with concurrent analytics of temporal and spatial information for fault detection in industrial processes," *IEEE Trans. Cybern.*, pp. 1–12, 2021.

[29] Y. Xie, W. Dai, Z. Hu, Y. Liu, C. Li, and X. Pu, "A Novel Convolutional Neural Network Architecture for SAR Target Recognition," *J. Sensors*, vol. 2019, p. 1246548, 2019.

[30] Y. Zhang, M. J. Er, R. Zhao, and M. Pratama, "Multiview Convolutional Neural Networks for Multidocument Extractive Summarization," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3230–3242, Oct. 2017.

[31] L. Wu, J. Z. Cheng, S. Li, B. Lei, T. Wang, and D. Ni, "FUIQA: Fetal ultrasound image quality assessment with deep convolutional networks," *IEEE Trans. Cybern.*, vol. 47, no. 5, pp. 1336–1349, 2017.

[32] R. Zhao, M. Pang, and J. Wang, "Classifying airborne LiDAR point clouds via deep features learned by a multi-scale convolutional neural network," *Int. J. Geogr. Inf. Sci.*, vol. 32, no. 5, pp. 960–979, 2018.

[33] J. C. Chen, A. Kumar, R. Ranjan, V. M. Patel, A. Alavi, and R. Chellappa, "A cascaded convolutional neural network for age estimation of unconstrained faces," in *IEEE 8th International Conference on Biometrics Theory, Applications and Systems, BTAS*, 2016, pp. 1–8.

[34] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2012, pp. 3642–3649.

[35] X. Bangquan and W. X. Xiong, "Real-time embedded traffic sign recognition using efficient convolutional neural network," *IEEE Access*, vol. 7, pp. 53330–53346, 2019.

[36] S. G. Nnabuife, J. Whidborne, and 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.

[37] X. Dong, C. Tan, Y. Yuan, and F. Dong, "Oil-water two-phase flow velocity measurement with continuous wave ultrasound Doppler," *Chem. Eng. Sci.*, vol. 135, no. 2, pp. 155–165, 2015.

[38] H. NYQUIST, "Certain Topics in Telegraph Transmission Theory," *Trans. Am. Inst. Electr. Eng.*, vol. 47, no. 2, pp. 617–644, 1928.

[39] M. Abadi *et al.*, "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," *Library (Lond).*, vol. s4-X, no. 3, pp. 339–339, Mar. 2016.

[40] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 39, no. 2014, pp. i–ii, Jan. 2012.

[41] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007.

[42] K. He, X. Zhang, R. Shaoqing, and J. Sun, "Deep Residual Learning for Image Recognition," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015.

[43] C. Szegedy *et al.*, "Going deeper with convolutions," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 07-12-June, pp. 1–9, 2015.

[44] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," *J. Mach. Learn. Res.*, vol. 15, pp. 1929–1958, Jan. 2014.

Nnabuife Somtochukwu is a Chartered Engineer with extensive offshore oil and gas expertise as a flow assurance/process engineer. He is experienced in interfacing with and integrating across various disciplines (topsides, subsea, subsurface, and geotechnical) and delivering systems and process designs (topsides and subsea) for a wide range of development types such as deep-water, gas storage, heavy oil, gas condensate, and HP/HT.

He earned a B.Sc. in Chemical Engineering from Anambra State University, an M.Sc. in Petroleum and Gas Engineering from the University of Salford in Manchester, UK, and a Ph.D. in Energy from Cranfield University, UK. His research interests include signal processing, process optimization, multiphase flow, sustainable energy, machine learning, and data analytics.

Boyu Kuang received the B.E. in aerospace engineering from the Civil University of China, Tianjin, China, in 2014 and the MSc. in Thermal Power from Cranfield University, United Kingdom, in 2017. He is currently pursuing the Ph.D. degree with the Centre for Computational Engineering Sciences (CES), Cranfield University.

His experience involves surface inspection, image semantic segmentation for planetary exploration, and ultrasonic signal processing. His research interests include computer vision, signal processing, image processing, machine learning, and the planetary rover's visual odometry.

James F. Whidborne received his BA in engineering from the Cambridge University, United Kingdom, and MSc and Ph.D. in systems and control from the University of Manchester Institute of Science and Technology (UMIST), UK. From 1991 to 1994, he held a position of Research Associate with the Department of Engineering, University of Leicester, UK. From 1994 to 2003, he was a Lecturer, then Senior Lecturer with the Department of Mechanical Engineering, Kings College London. He is currently the Head of the Dynamics Simulation and Control Group in the Centre for Aeronautics at Cranfield University, UK.

He has almost 200 refereed research publications, including three books. He is a Chartered Engineer, a Member of the IET, and a Senior Member of the IEEE. His research interests are in the theory and application of advanced control, including fluid flow control, multiobjective robust control, finite precision controller implementation problems, directional drilling and flight control.

Zeeshan Rana has several years of industrial and academic experience in the vision-based system. He obtained his BSc in Mechanical Engineering from Pakistan and then progressed to MSc in Computer Vision from the UK. Later he earned his Ph.D. in Computational Engineering from Cranfield University, UK. His focus has been on the development of complex algorithms for various engineering applications including stereo vision-based inspection of cosmetically sensitive surfaces. In Cranfield, he has been the option leader for the MSc in Computer and Machine Vision and teaches various modules related to the topic. He is currently PI on research projects in the area of computer vision and AI, such as object tracking, drone detection using stereo vision, sign language identification using computer vision. Dr Rana has provided consultancy work to the industry on various projects and recently won funding from the Department of Transport (DfT) for research in the area of small UAV detection using stereo vision. Dr Rana is a Fellow of HEA, a Member of IET, course director of the MSc course, and reviewer of high-impact international journals.