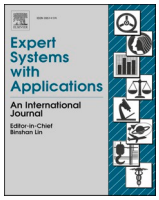
Expert Systems With Applications 236 (2024) 121414

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Self-supervised learning-based two-phase flow regime identification using ultrasonic sensors in an S-shape riser

Boyu Kuang [a,1], Somtochukwu G. Nnabuife [b,2], James F. Whidborne [a,*,3], Shuang Sun [c,4], Junjie Zhao [a,5], Karl Jenkins [a,6]

[a] School of Aerospace, Transport, and Manufacturing, Cranfield University, Cranfield MK43 0AL, United Kingdom
[b] School of Water, Energy, and Environment, Cranfield University, Cranfield MK43 0AL, United Kingdom
[c] College of Aviation Engineering, Civil Aviation University of China, 2898 Jinbei Road, Dongli District, Tianjin 300300, China

ARTICLE INFO

Keywords:
Weak-supervised learning
Multiphase flow
Classification
Feature extraction
Complexity

ABSTRACT

Two-phase flow regime identification is an essential transdisciplinary topic that spans digital signal processing, artificial intelligence, chemical engineering, and energy. Multiphase flow systems significantly impact pipeline safety, heat transfer, and pressure drop; therefore, precisely identifying the governing flow regime is crucial for effective modeling and design. However, it is challenging due to the geometrical complexity of flow regimes in multiphase flow. With the advances in sensor measurement and machine learning, applying non-destructive tests and self-supervised learning to practical industrial problems has become technically feasible and cost-effective. This study applies a weak-supervised learning-based two-phase flow regime identification solution using a non-destructive tests ultrasonic sensor in an S-shape riser experimental bed by proposing a self-supervised feature extraction algorithm. The proposed self-supervised feature extraction algorithm reduces time/labor consumption and human error in data annotation using SSL, which provides full supervision without manual annotation. The self-supervised feature extraction algorithm uses a bottlenecked neural network and encoder-decoder structure to extract compact features. The self-supervised feature extraction algorithm performance is evaluated using an established convolutional neural network-based classifier. The source data was collected from a 10 × 50 m riser experimental rig. The dataset is made available to the community as part of this study. The performance of the approach is comparable with state-of-the-art methods and is also the first successful attempt to apply self-supervised learning to multiphase flow regime ultrasonic signal identification. This study achieved 98.84%, 0.000663, 0.00312, and 7.71×10^5 in accuracy, root mean square error, categorical cross-entropy, and model complexity, respectively. The practical experiment justifies the robustness, fairness, and practicability in the practical application environment. The proposed self-supervised feature extraction brings new approaches and inspirations for the feature extraction step in identifying a two-phase flow regime, and it will be beneficial to generalize this study in different riser shapes in the future.

1. Introduction

Multiphase flow is a term used to describe the flow of two or more fluid phases, such as gas and liquid, within a single pipeline or vessel (Nazeer et al., 2022). These systems are ubiquitous in various industrial processes, such as oil and gas production, chemical processing, and nuclear reactors, and pose several challenges in terms of modeling and design (Balachandar et al., 2020; Lube et al., 2020; Roshani et al., 2021;

* Corresponding author.
E-mail addresses: neil.kuang@cranfield.ac.uk (B. Kuang), g.nnabuife@alumni2015.cranfield.ac.uk (S.G. Nnabuife), j.f.whidborne@cranfield.ac.uk (J.F. Whidborne), junjie.zhao@cranfield.ac.uk (J. Zhao), k.w.jenkins@cranfield.ac.uk (K. Jenkins).
[1] ORCID 0000-0003-1828-7663.
[2] ORCID 0000-0002-4050-5375.
[3] ORCID 0000-0002-6310-8946.
[4] ORCID 0000-0002-6644-5278.
[5] ORCID 0000-0002-1791-6061.
[6] ORCID 0009-0006-4618-5786.

https://doi.org/10.1016/j.eswa.2023.121414
Received 28 March 2023; Received in revised form 15 August 2023; Accepted 30 August 2023
Available online 3 September 2023

Shi et al., 2021). Two-phase flow, a specific type of multiphase flow, refers to the simultaneous flow of two immiscible fluids, such as gas and liquid (Nazeer et al., 2022). Identifying the two-phase flow regime is essential for predicting system behavior, optimizing performance, and avoiding potential hazards. For example, it affects heat transfer, pressure drop, system stability, and other key parameters. In recent years, there has been a growing interest in developing novel techniques and methods for two-phase flow regime identification, leveraging advances in sensor measurement and machine learning (Hammad et al., 2021; Z. Lin et al., 2020; Shen & Hibiki, 2021; H. Xu et al., 2022).

Multiphase flow is common in under-sea reservoirs for offshore oil and gas production (Nnabuife et al., 2022). Pipeline-riser systems are usually used to transport the fluids to offshore facilities. However, this process can be complicated because of slugging, which is characterized by large-amplitude pressure instabilities that can pose a significant safety risk to pipeline operations. Slugging is typically caused by low multiphase flow rates in pipeline risers, and accurately identifying multiphase flow regimes and their correlations with flow parameters is essential for maintaining pipeline safety (Nnabuife et al., 2022). Previous research has explored various methods for addressing this issue, but the complexity of multiphase flow regimes in pipeline-riser systems has limited these methods. In this study, we propose a novel self-supervised learning (SSL) approach for identifying two-phase flow regimes using ultrasonic signals, which we believe can improve current methods and reduce the risk of pipeline instability and failure.

Flow regime identification is essential for multiphase flow systems modeling and design. However, it is complicated because of the movement of different phases like gas, liquid, and solid particles (Holland & Bragg, 1995). In pipeline-riser systems, it is especially difficult because of the distinctive geometrical complexity of the flow regimes, which is influenced by multiphase interactions, pipeline configuration, flow properties, and operating conditions. Understanding the inner correlations between multiphase flow regimes and flow parameters is important to the pipeline-riser system (Buscher, 2019; Gupta et al., 2022). Flow regime refers to the patterns of the two phases flowing through the pipeline, impacting the safety, heat transfer, and pressure drops (Holland & Bragg, 1995). Machine learning has shown promising achievements in flow regime identification, providing faster and more accurate alternatives than traditional methods (Z. Lin et al., 2020; Q. Xu et al., 2021).

Considering the geometrical complexity of the flow regimes in multiphase flow, precisely identifying the governing flow regime is difficult (Nnabuife, Pilario, et al., 2019). According to Zhang et al. (2020), the stages of automatic recognition commonly include data acquisition, feature extraction, and machine learning classification. Feature extraction can be generally grouped into the heuristic and metaheuristic approaches (Arif et al., 2015). Heuristic approaches rely on iterative algorithms or techniques that aim to find a satisfactory solution to a problem without guaranteeing optimality. The typical heuristic approaches include sequential forward selection, randomized and enumeration-based algorithms, scatter criterion, symmetric uncertainty, fast correlation-based filter, minimal-redundancy-maximal-relevance, non-dominated sorting genetic algorithms, and neural network-based deep learning (Arif et al., 2015). Metaheuristic approaches aim to explore search spaces and find optimal or near-optimal solutions by iteratively improving candidate solutions. The typical metaheuristic approaches include genetic algorithms, simulated annealing, artificial bee colony, nearest neighbor, multi-cluster feature selection, and axiomatic fuzzy set (Arif et al., 2015). Power spectral density (Nnabuife, Pilario, et al., 2019), discrete wavelet transform (Abbagoni & Yeung, 2016), empirical mode decomposition, and statistical features are a few of the methods applied in previous studies for feature extraction, which is selected depending on the specific type of sensor used to acquire the data (Hanus et al., 2018). The essence of feature extraction in signal processing is to transform raw input signals into a more compact and representative set of features that capture the relevant information for further analysis or classification tasks (T. Pan et al., 2020). Dimensional reduction approaches, like principal components analysis (Nnabuife, Pilario, et al., 2019; Q. Xu et al., 2020) and kernel principal components analysis, can be used for further processing and analysis since these features could be highly dimensional and highly cross-correlated. Finally, many machine-learning techniques have been reported to have satisfactory performance for the classification process, including support vector machines, artificial neural networks (Hanus et al., 2018), and gradient-boosted trees (Z. Lin et al., 2020).

Advances in sensor technology, data science, and artificial intelligence (AI) make the two-phase flow identification increasingly practical, real-time, and efficient (Huang et al., 2002). Since the success of ImageNet in 2012, AI solutions have made an increasingly significant difference in various industrial products (Bécue et al., 2021; Tsang & Lee, 2022). However, most AI solutions apply supervised learning, which requires substantial human effort and cost to provide sufficient annotations (Swan et al., 2021; Tsai & Chang, 2021; Villalobos et al., 2022). Furthermore, manual annotation introduces human error (Swan et al., 2021). Therefore, adopting self-supervised or unsupervised AI solutions can reduce human labor and time costs. State-of-the-art AI solutions for flow regime identification utilize supervised learning, in which manual annotation is essential. This study explores the feasibility of SSL in feature extraction for the two-phase flow regime identification, which can significantly reduce human error, effort, and time when conducting manual annotation. A benchmark (Kuang et al., 2021; Nnabuife, Kuang, Rana, et al., 2021; Nnabuife, Kuang, Whidborne, et al., 2021a, 2021b; Nnabuife, Pilario, et al., 2019; Nnabuife, Whidborne, et al., 2019) for flow regime identification is used to verify the effectiveness of the proposed self-supervised feature extraction (SSFE) algorithm.

Machine learning methods can be divided into two categories, supervised learning and weak-supervised learning (S. J. Pan & Yang, 2010). Supervised learning depends on the difference between annotations and predictions (Jiang et al., 2020). However, precise annotations for practical problems are challenging.

The research question is, “Can self-supervised learning overcome time, labor, and human error challenges in manual data annotation in two-phase flow regime identification using a non-destructive testing ultrasonic sensor while keeping an accurate and competitive performance?” The motivation behind this study stems from the importance and significant impact of two-phase flow regime systems on pipeline safety, heat transfer, and pressure drop. However, the geometrical complexity of flow regimes brings considerable challenges to precisely identifying the two-phase flow regime. The motivation is further driven by the advancements in sensor measurement and machine learning techniques. The study recognizes the technical feasibility and cost-effectiveness of applying non-destructive tests and self-supervised learning to practical industrial problems. Therefore, this study targets to reduce time, labor, and human error associated with manual data annotation while keeping an accurate and competitive performance.

The contributions of this study are summarized as follows. (i) This study proposes an innovative Self-Supervised Feature Extraction (SSFE) algorithm that reduces the time, labor, and potential human error through self-supervision while achieving comparable performance to state-of-the-art methods. (ii) To promote research and collaboration, this study generates a new self-supervised features (SSFs) dataset[7] and makes it available to the community (Kuang, 2023). (iii) As far as the authors are aware, this study is the first successful application of Self-supervised Learning (SSL) to two-phase flow regime identification with an ultrasonic sensor, providing valuable insights for future research and practical implementation. The proposed SSFE algorithm has the

[7] The self-supervised features (SSFs) dataset is made available to the community, which can be accessed at: https://doi.org/10.17862/cranfield.rd.22241530.v1.

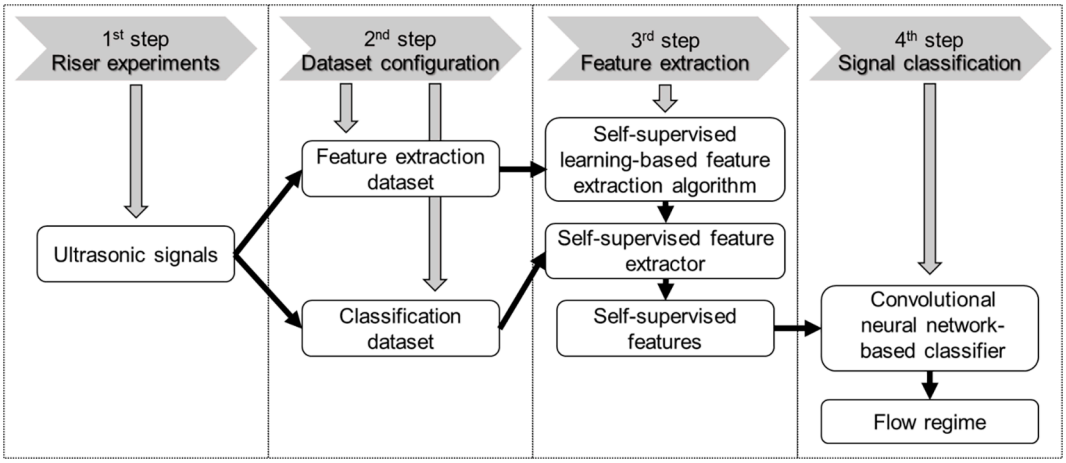


Fig. 1. The overall process of this study.

potential to significantly advance the understanding of two-phase flow and improve industrial process design and optimization.

The structure of this paper as follows. Section 2 discusses related works from two aspects. Section 3 describes the methodology used in this study, which includes riser experiments to collect the source signals, dataset configuration, feature extraction based on SSL, and signal classification based on convolutional neural networks (CNN). Section 4 depicts the experimental design of this study. Section 5 discusses the results of self-supervised feature and flow regime signal classification. Finally, conclusions and future works are summarized in Section 6.

2. Related works

This section discusses related study from two aspects additional to the research questions and contributions mentioned in Section 1. The first aspect focuses on the concept applied in this study, self-supervised learning (SSL). An explanation about using SSL by comparing various weak-supervised learning approaches is conducted. The second aspect focuses on the task, the proposed feature extraction solution. A comparison among different feature extraction methods in existing solutions highlights the necessity of SSL-based feature extraction.

Weak-supervised learning overcomes the challenge by reducing manual intervention. Weak-supervised learning consists of coarse supervision, incomplete supervision, noise supervision, and self-supervision (Qian et al., 2019; Zhou, 2018). Coarse supervision uses a coarse annotation for rough supervision (like bonding box and target labels) (Hashmi et al., 2022). Incomplete supervision only labels a part of the samples or domains (Zhou, 2018). Noise supervision contains misannotations (Q. Li et al., 2020). Self-supervised learning, where models learn from intrinsic structures and patterns in data without manual labeling, has shown great success recently in natural language processing through models like BERT (Y. Li et al., 2021) and GPT (W. Lin et al., 2021) for language modeling. Similarly, self-supervised approaches have achieved state-of-the-art results in computer vision tasks like image classification (Reed et al., 2021) using transformations and augmentations as supervisory signals. The promise of self-supervision across domains motivates the exploration of its potential for multiphase flow applications. The high cost of supervised learning is the key motivation to explore the possibility of weak-supervised learning. However, incomplete supervision is difficult for flow regime identification because of the high similarity between different flow regime signals. Thus, SSL becomes a very attractive option.

The feature extraction in the related study can be broadly divided into traditional non-data-driven solutions and supervised learning solutions. Traditional non-data-driven solutions like principal component analysis (Nnabuife, Pilario, et al., 2019) or ideal low-pass filter (Kuang et al., 2021; Nnabuife, Kuang, Rana, et al., 2021; Nnabuife, Kuang, Whidborne, et al., 2021a, 2021b) have been applied for multiphase flow characterization using extracted statistical features. However, these hand-crafted features are limited in handling complex flow dynamics. More recently, (Tan et al., 2021) proposed using sparse batch normalization convolutional neural network for ultrasonic signal feature learning in a gas–liquid stratified flow. They showed improved flow pattern recognition accuracy compared to statistical features. However, their supervised autoencoder model requires large labeled data clusters to train properly. Supervised learning solutions usually use a 1D convolutional neural network model with front layers for automatic feature extraction from ultrasonic signals and back layers for flow regime classification. Their model achieved approximately 95–95% testing accuracy on classifying bubble, slug, and annular flows. However, a key limitation is the need for a large annotated dataset to train the supervised model, which can be impractical to obtain for new fluid conditions and flow loops (Kuang et al., 2021; Nnabuife, Kuang, Rana, et al., 2021; Nnabuife, Kuang, Whidborne, et al., 2021b, 2021a). Furthermore, the supervised learning solutions bring high model complexity because of the high compression ratio from the input signals to output classes. Kuang et al. (2022) systematically discuss the model complexity with different neural network designs.

Ultrasonic sensors have become widely used for non-intrusive characterization and measurement of multiphase flows across domains like oil and gas, nuclear, and chemical engineering. Their popularity stems from their ability to penetrate optically opaque mixtures and provide spatiotemporal information on velocity and phase profiles (Murai et al., 2010). Based on the limitations of existing traditional non-data-driven and supervised feature extraction methods for multiphase flows, the key objectives of the proposed SSFE are to 1) develop a customized self-supervised learning approach for ultrasonic signal feature extraction that does not require manual labels and 2) demonstrate its efficacy in improving flow regime classification accuracy. The novelty lies in harnessing self-supervision to exploit the rich information contained in ultrasonic measurements as an alternative supervisory signal.

3. Method

The overall process of this study is illustrated in Fig. 1 and consists of four steps, riser experiment, dataset construction, feature extraction, and signal classification. In the 1st step, the riser experiment refers to the original ultrasonic signal collection step, which used a noninvasive Doppler ultrasound sensor to collect ultrasound signals from an S-shaped riser (see Section 3.1). In the 2nd step, the dataset configuration step reconfigures the rise signals into two independent sub-datasets, the

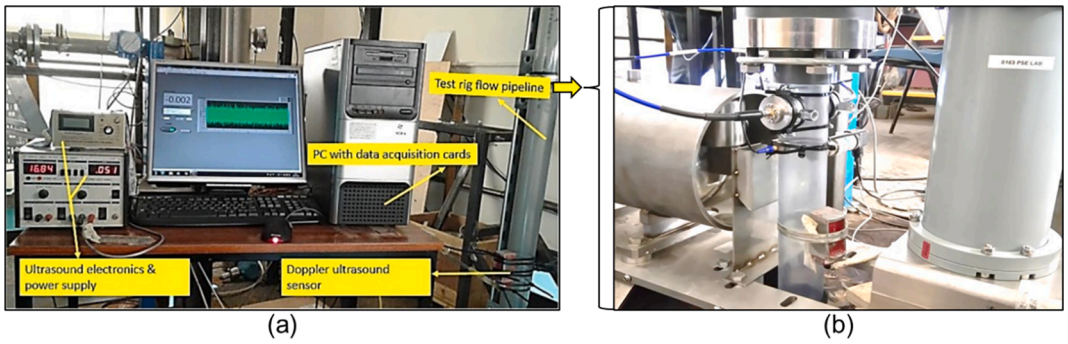


Fig. 2. The photos of the two-phase flow test riser.

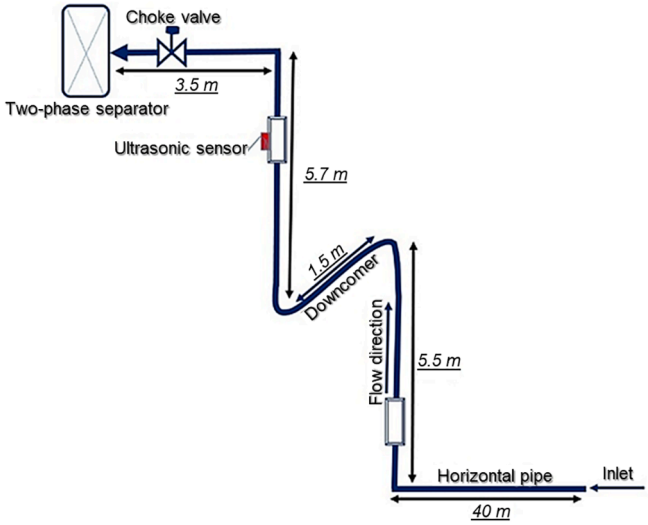


Fig. 3. The schematic diagram of the S-shape riser.

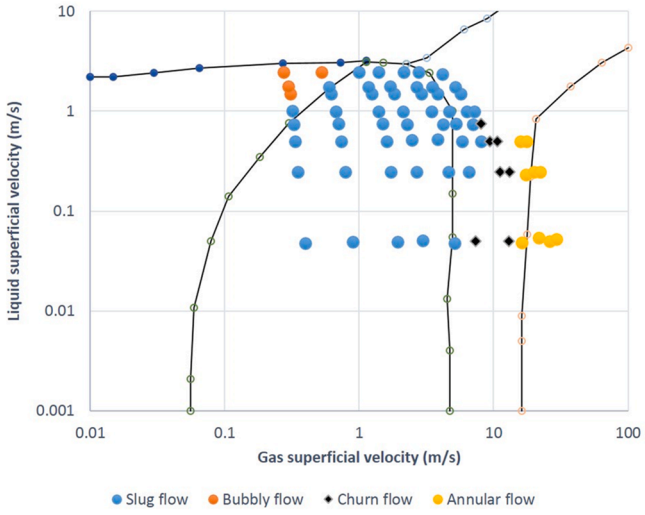


Fig. 4. The flow regime diagram.

feature extraction dataset and the classification dataset (see Section 3.2). In the 3rd step, the feature extraction step proposes an SSFE algorithm following an encoder-decoder structure. The SSFE algorithm is trained using the feature extraction dataset (see Section 3.3). In the 4th step, the self-supervised feature extractor achieves the self-supervised features (SSF) from the classification dataset. In the signal classification step, a CNN-based classifier is applied, which is used to verify the feasibility of the proposed SSFE and the achieved SSF (see Section 3.4).

3.1. The riser experiments

The source data used in this study comes from water-air two-phase flow experiments in an S-shape riser performed at the Process Systems Engineering Laboratory at Cranfield University. This source data has been extensively used in many related studies (Kuang et al., 2022; Nnabuife, Kuang, Rana, et al., 2021; Nnabuife, Kuang, Whidborne, et al., 2021a, 2021b; Nnabuife, Pilario, et al., 2019; Nnabuife, Sharma, et al., 2021; Roxas II et al., 2022), which justifies the data on validity, accuracy, and reliability.

The riser experiment is conducted on an S-shaped riser test platform with a height of 11 m and a length of 50 m, which can transport 140–1,400 cubic meters of water or oil per hour, and the air supply capacity/velocity can reach 40 m per second. The compressor of the riser test platform can provide a maximum of 20 barg. The two-phase flow test riser consists of three parts, the standpipe, the ultrasonic sensor, and the signal processing unit. Fig. 2(a) and (b) refer to the photos of the two-phase flow test riser, and Fig. 3 refers to a schematic diagram. The riser experiments used a 2-inch S-shaped riser system with

Table 1
The records of the riser experiments.

Flow regime annotation	Record file	Ultrasonic signal recording point			
		1	2	… …	1,300,000
1	5Air1.5Water	0.068359	−0.400391	… …	−0.756836
	…	…	…	… …	…
	120Air5Water	0.849609	0.214844	… …	1.672363
2	4Air3Water	2.478027	2.370605	… …	1.948242
	…	…	…	… …	…
	200Air4.5Water	1.643066	1.130371	… …	1.557617
3	10Air3.5Water	1.342773	1.308594	… …	1.513672
	…	…	…	… …	…
	300Air3.5Water	2.211914	1.818848	… …	2.255859
4	70Air0.1Water	−0.822754	−0.881348	… …	0.124512
	…	…	…	… …	…
	300Air1.5Water	0.075684	−0.107422	… …	0.617676

an internal diameter of 54.8 mm. The S-shaped pipeline system can be divided into five parts according to the direction, a 40-meter-long horizontal pipeline, a 5.5-meter vertical rising pipeline, a 1.5-meter downcomer pipeline, a 5.7-meter vertical rising pipeline (ultrasonic sensors are installed here), and 3.5 m of horizontal piping (a choke valve is installed here as a two-phase flow separator) (Nnabuife, Pilario, et al., 2019).

The riser experiment controls the flow regime in the S-shaped riser by changing the input of gas and liquid (air and water). The ultrasonic sensor measures the ultrasonic signal from the different two-phase flows in the S-shape riser. The corresponding flow regime is annotated according to the flow regime diagram (see Fig. 4). One hundred and twenty-five riser experiments were conducted to obtain the source data, covering four flow regimes, annular flow, churn flow, slug flow, and bubbly flow. The duration of each experiment is the same, which has approached the maximum capability of the riser bench with the safety requirements. The data records for the experiments are 1.3 million record points, far beyond the minimum record length for flow regime identification (this has been fully demonstrated in the study (Nnabuife, Kuang, Whidborne, et al., 2021a)). The records of the riser experiments are shown in Table 1, the first column refers to the flow regime annotation, and columns 2 to 1,300,001 are the recording points.

3.2. Dataset configuration

The data construction aims to prepare the datasets for the following deep learning procedures. The datasets used in feature extraction and signal classification are named the "feature extraction dataset" and the "classification dataset" in this study. The feature extraction dataset and the classification dataset are both configured using the process of Algorithm 1 and the source data from the riser experiment. Notably, feature extraction and signal classification are independent deep learning processes, so information leakage does not exist between the processes. Algorithm 1 contains a random shuffle, which makes the final feature extraction and classification datasets different datasets.

Algorithm 1 implements three functions. Firstly, Algorithm 1 divides every single experimental record into multiple short signal segments using the twin-window feature extraction algorithm (Nnabuife, Kuang, Whidborne, et al., 2021a) to achieve the effect of data enhancement. Kuang et al. (2022), Kuang, Nnabuife, et al. (2021), and Nnabuife, Kuang, Whidborne, et al. (2021b, 2021a) have shown that the information in a single experimental record from the riser experiments is more redundant than necessary for detecting the flow regime. Secondly, Algorithm 1 uses the fast Fourier transformation (FFT) algorithm to transfer the divided short signal segments from the time domain to the frequency domain. Kuang, Nnabuife, et al. (2021), and Nnabuife, Kuang, Whidborne, et al. (2021a) claim that the frequency domain transformation makes the flow regime characteristics too obvious. Thirdly, Algorithm 1 randomly distributes different flow regimes in the dataset through random shuffling. The random shuffle avoids the clustering of categories because the source data records adjacently for the same flow regime (Nnabuife, Pilario, et al., 2019; Nnabuife, Whidborne, et al., 2019). The air–water ratio in the riser experiments was adjusted from the minimum to the maximum, so a similar air–liquid ratio groups the adjacent records. However, such group patterns violate the robustness and generalizability requirement of the learning process. Fourthly, Algorithm 1 further divides the augmented short signal segments into the training, testing, and verification sets according to the ratio of 70%, 15%, and 15%.

The pseudo-code is shown in Algorithm 1. The input $\boldsymbol{R}$ represents the overall riser experiment record (the source signal), a 2D matrix with a dimension of $125 \times 1,300,000$. The r_x represents a single experimental record (a single row in $\boldsymbol{R}$), and the x subscript refers to the experiment index (the row index). The first digit of r_x is the flow regime annotation, and the riser experiment uses the continue integers (1, 2, 3, and 4) to represent the four flow regimes (annular, churn, slug, and bubbly flow), respectively. Parameters l_{window}, l_{step}, and l_{signal} refer to the window length, step length of the window shift, and signal length of a single experimental record, respectively. The output $\boldsymbol{X}$ represents the short signal segment converted to the frequency domain, and $\boldsymbol{Y}$ represents the corresponding flow regime annotation. The output annotation adopts the one-hot encoding method, and 1, 2, 3, and 4 are encoded as $[1,0,0,0]$, $[0,1,0,0]$, $[0,0,1,0]$, and $[0,0,0,1]$, respectively. One-hot encoding turns the multi-class classification into multiple binary classifications, which can avoid the gradient confusion caused by intermediate values between the adjacent integers. The subscripts of $\boldsymbol{X}$ and $\boldsymbol{Y}$ refer to the training set, testing set, and validation set. The training set is used for model training, the testing set supports the training process to overcome overfitting and underfitting, and the validation set does not participate in the training process. Here is a detailed description of the algorithm:

(1) $\boldsymbol{R}$ is firstly divided into $\boldsymbol{L}$ and $\boldsymbol{S}$. The $\boldsymbol{L}$ refers to the annotation corresponding to r_x ($\boldsymbol{L} = [l_1, l_2, ..., l_{125}]$), and $\boldsymbol{S}$ represents the experimental signal corresponding to r_x ($S = [s_1, s_2, ..., s_{125}]$).
(2) Find the numbers of the four flow regimes from $\boldsymbol{L}$ and save them in rg_1, rg_2, rg_3, and rg_4, respectively.
(3) Shuffle rg_1, rg_2, rg_3, and rg_4 to get (rg_1', rg_2', rg_3', and rg_4'). The air–water change in the riser experiment changes gradually, and the shuffle can increase the robustness in terms of input diversity.
(4) Divide (rg_1', rg_2', rg_3', and rg_4') into three parts according to the 70%, 15%, and 15% ratio, respectively. Then, combine all the first parts into the training set, all the second parts into the testing set, and all the third parts into the validation set. Notably, the operation here is only the number recorded by the riser.
(5) Iterate over each riser record in the training set, testing set, and validation set.
(6) Read the annotations and source records from the corresponding riser experiment, and save them as l_{curr} and s_{curr}.
(7) Round the ratio of l_{record} to l_{window} to obtain the number of short signals (windows) divided into a single experimental record.
(8) Traverse each short signal that has been segmented.
(9) Convert the short signals from the time domain to the frequency domain using the FFT algorithm.
(10) Save the training, testing, and validation sets.

Algorithm 1: Dataset configuration

Inputs: $\boldsymbol{R} = [r_1, r_2, ..., r_{125}]$.
l_{window}, l_{step}, and l_{signal}.
Outputs: $\boldsymbol{X}_{train}$, Y_{train}, $\boldsymbol{X}_{test}$, Y_{test}, $\boldsymbol{X}_{valid}$, Y_{valid}.

$\boldsymbol{R} \rightarrow (\boldsymbol{L}, \boldsymbol{S})$;
$(rg_1, rg_2, rg_3, rg_4) \leftarrow$ *where* $\boldsymbol{L} = 1,2,3,4$, respectively;
$(rg_1', rg_2', rg_3', rg_4') \leftarrow$ *shuffle* (rg_1, rg_2, rg_3, rg_4);
divide & concatenate $\rightarrow (rg_{train}, rg_{test}, rg_{valid})$;
for rg_i *in* $(rg_{train}, rg_{test}, rg_{valid})$:
  $l_{curr} = \boldsymbol{L}[rg_i], s_{curr} = \boldsymbol{S}[rg_i]$;
  $n_{window} = [(l_{signal} - l_{window})/l_{step}]$;
  for i in range (n_{window}) :
    $s_{seg} = s_{curr}[i \times l_{step}, i \times l_{step} + l_{window}], f_{seg} \leftarrow FFT(s_{seg})$;
    According to $(rg_{train}, rg_{test}, rg_{valid})$:
      Save f_{seg} to corresponding $\boldsymbol{X}_{train}$, $\boldsymbol{X}_{test}$, or $\boldsymbol{X}_{valid}$;
      Save l_{curr} to corresponding Y_{train}, Y_{test}, or Y_{valid}.

The outcome of dataset configuration is a feature extraction dataset and a classification dataset. Both datasets are 2D matrices with a dimension of $125 \times n_{window} \times (l_{window}/2 + 1)$. The first column contains the flow regime annotation, and the frequency domain signal segment is between the second and the last columns. The overall dataset contains samples $n_{data} = [125 \times n_{window}]$. Thus, the training set contains $n_{train} = [n_{data} \times 70\%]$ samples, the testing set contains $n_{test} = [n_{data} \times 15\%]$ samples and the validation set contains $n_{valid} = [n_{data} - n_{train} - n_{test}]$ samples.

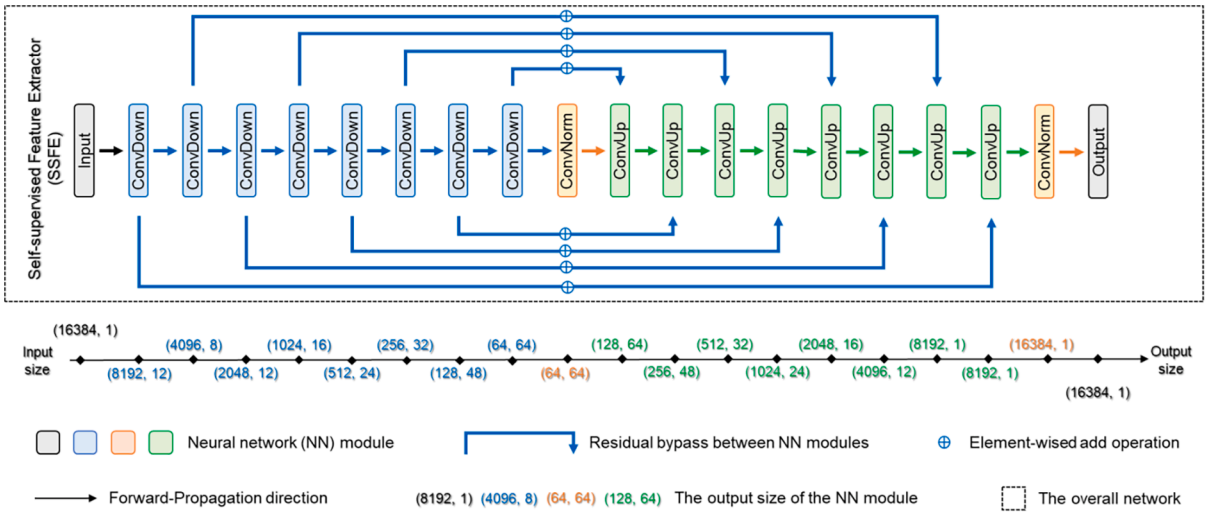


Fig. 5. The network structure of the proposed SSFE.

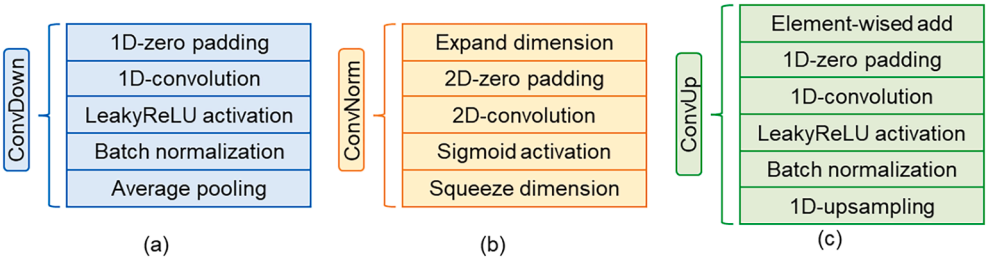


Fig. 6. The specific layouts of each module in the network of the proposed SSFE.

3.3. Feature extraction

Feature extraction is essential in performing deep learning-based classification tasks, usually known as feature engineering (Reid Turner et al., 1999). A superior feature extraction should have three characteristics: Firstly, it can reduce the dimension. Secondly, improve the learnability for the following classification tasks. Thirdly, keep valid information while excluding invalid information regarding the classification task. The feature extraction dataset and the classification dataset contain a large input dimension because the FFT in Algorithm 1 keeps the same output length as the l_{window} to avoid the necessary information loss. Nnabuife, Kuang, Whidborne, et al. (2021b, 2021a) have justified that a significant dimension change of the FFT input–output can significantly influence classification performance.

This study proposes a fully convolutional network and encoder-decoder-based SSFE algorithm with a U-Net-formed structure. A U-Net structure is an effective neural network widely used in many advanced studies (Reid Turner et al., 1999). The input and output are $\boldsymbol{X}_{train}$, $\boldsymbol{X}_{test}$, and $\boldsymbol{X}_{valid}$ in Algorithm 1. The SSFE takes the input as the output to implement the self-supervision-based feature extraction. Notably, the $\boldsymbol{Y}_{train}$, $\boldsymbol{Y}_{test}$, and $\boldsymbol{Y}_{valid}$ are not used in the SSFE process. Thus, the proposed SSFE does not require any annotation or human intervention. The feature extracted by the self-supervised feature extractor is the output of the encoder part, and each one-dimensional input is converted into a compact 2D feature with a dimension of 64×64.

Figs. 5 and 6 show the network structure and the specific layouts of each module in the network of the proposed SSFE algorithm, respectively. The SSFE algorithm consists of four modules, the input/output module, convolutional downsampling module (ConvDown), convolutional normalization module (ConvNorm), and convolutional upsampling module (ConvUp). The input/output module provides the input and output interfaces for the overall SSFE to connect with $\boldsymbol{X}_{train}$, $\boldsymbol{X}_{test}$, and $\boldsymbol{X}_{valid}$ in the feature extraction dataset. The convolutional downsampling module (see Fig. 6(a)) accordingly consists of a zero-padding layer, a one-dimensional (1D) convolution layer, a LeakyReLU activation layer, a batch normalization (BN) layer, and an average pooling layer. The zero-padding layer fills two zeros at both ends of the input. The convolution kernel is a vector with a length of five, and the depth increases to avoid a significant information loss between the input and output tensors of a module. Notably, the tensor dimension remains the same between the module input and output of the convolution layer. The downsampling layer uses two as the stride length, and the output tensor has half the input length. The dimension expansion layer in the convolution normalization module converts the 1D input into a 2D tensor by expanding a new dimension at the end, corresponding to the depth of the 2D tensor (see Fig. 6(b)). Both the zero-padding layer and the convolution layer shift to the 2D version. The padding layer fills a zero edge surrounding the 2D tensor, and the convolution kernel applies the size of 3×3. The convolutional normalization layer normalizes the output using a sigmoid activation. The convolutional normalization layer is used in the outputs of the SSFE and the overall neural network. The normalization layer normalizes the extracted features and the overall neural network output between zero and one. The squeeze layer converts the normalized tensors back into a 1D tensor by eliminating the expanded depth (because the expanded depth equals one). The convolutional upsampling module is equivalent to an inverse process of the convolutional downsampling module (see Fig. 6(c)). The convolutional upsampling module constructs a bypass highway of the gradient propagation between the shallow and deep networks through the element-wised add operation, which is similar to the bypass channel in the ResidualNet (Z. Zhang et al., 2021) and widely used in many studies (Marcinkiewicz et al., 2019; Qiu et al., 2021). The convolutional

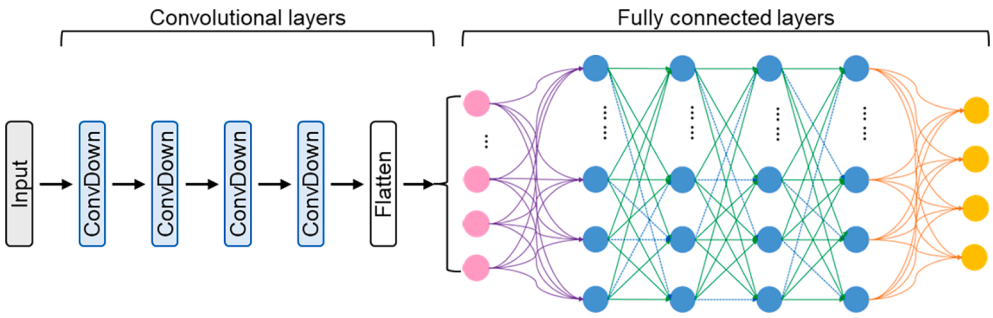


Fig. 7. The CNN-based flow regime classifier.

upsampling module then adopts an upsampling layer, and the size doubles and depth decreases to maintain the overall information. The lower part of Fig. 5 further depicts an example of dimension changes throughout the neural network. At the same time, the specific values adopt the condition of the experimental group (see Section 4.1 for the details of the experimental and control groups). It can be found that the input is a 1D tensor of 16,384 × 1 (the left black bracket), and the output of the feature extractor is a 2D tensor of 64 × 64 (the middle orange bracket).

3.4. Signal classification

The flow regime classification applies supervised learning, which aims to evaluate the performance of the achieved SSFs from the proposed SSFE. The flow regime supervision is the one-hot annotations in the classification dataset (see Section 3.2). The flow regime classifier utilizes a CNN-formed classifier, a proven effective flow regime classifier in Nnabuife, Kuang, Whidborne, et al. (2021b). The CNN classifier is not the focus of this study, so this study does not extend the discussion of the CNN classifier. The CNN classifier aims to implement two functions. Firstly, this study used the well-proved CNN-based flow regime classifier to evaluate the effectiveness of the proposed SSFE and achieved SSF, which can significantly strengthen the reliability of the evaluation results. Secondly, this study evaluates the improvement in computational complexity by comparing the proposed and the state-of-the-art CNN classifiers, which can justify the advance and necessity of the proposed SSFE. This can also improve the feasibility of implementing an embedded device and the real-time performance of the solution.

The classifier in this study applies a CNN-based structure (see Fig. 7). The convolution part compresses the SSF into a 1D vector and then uses the fully connected layer to analyze the high-level information in the compressed 1D vector. The number of parameters indicates the computational complexity of the neural network (Kuang et al., 2022). The classifier contains four convolutional layers and three fully connected layers, corresponding to about 50,000 parameters. Kuang et al. (2022) systematically studied the relation among the parameter amount, model complexity, and classification performance. The number of parameters for the flow regime classifier in this study is lower than the lowest case in Neshatpour et al. (2018) and Radev et al. (2022). Section 4.2 has a detailed discussion of the number. Fig. 7 shows the neural network layout of the CNN-based flow regime classifier, and the ConvDown module shares the same architecture as in Fig. 6.

4. Experiments

The experiments can be divided into two parts, the riser experiments and the signal processing experiments. The riser experiment in this study uses ultrasonic non-destructive testing sensors and Continuous Wave Doppler Ultrasound (CWDU) signal processing software to record the time-domain Doppler flow regime signals (Nnabuife, Pilario, et al., 2019). The signal processing experiments concentrate on the signal processing of the ultrasonic signals from the riser experiments. The signal processing in this study can be two steps, the SSL-based feature extraction, and the CNN-based classification. The dataset construction and frequency domain transformation use Python-based NumPy and Panda libraries. All neural networks are implemented using the GPU version of TensorFlow 2.1.16 (Abadi et al., 2016) on Linux Ubuntu18.04, and the visualization uses the matplotlib library. The hardware is Lenovo Thinkstation, which has 32 GB memory, Core i7-7700 CPU, and NVIDIA RTX 1080.

This study separately employs one experimental experiment of the experimental group (*ex*) and seven control experiments of the control group (*ctr*) to test the proposed SSFE algorithm. The *ctr* is set for the window length (l_{window}) and window moving step ratio ($r_{s/w}$) in the self-supervised feature extractor. Table 2 shows the experimental settings for the *ex* and *ctr* groups. The value of l_{window} is obtained using Equation (1), and the number of self-supervised features (n_{SSF}) is obtained by Equation (2). Parameter l_{window} determines the length of the experimental riser data segmented in one feature extraction, which refers to the overall amount of information used in one feature extraction. The value $r_{s/w}$

Table 2
The experimental settings for the experimental and control groups.

Group		Index	l_{Window}	$r_{s/w}$	l_{step}	n_{SSF}	d_{SSF}	$r_{Tr/Te/Va}$
experiment		*ex*	65,536	1/4	16,384	9,418	64 × 64	70% / 15% / 15%
control	*ctr-A*	*ctr-Aa*	16,384	1/4	4,096	39,125	64 × 64	
		ctr-Ab	32,768	1/4	8,192	19,250	64 × 64	
		ctr-Ac	131,072	1/4	32,768	4,375	64 × 64	
	ctr-B	*ctr-Ba*	65,536	1/2	32,768	4,625	64 × 64	
		ctr-Bb	65,536	1/8	4,096	18,750	64 × 64	
	ctr-C	*ctr-Ca*	65,536	1/4	16,384	9,418	32 × 32	
		ctr-Cb	65,536	1/4	16,384	9,418	128 × 128	

Table 3
The hyperparameters of the SSFE experiments.

Hyperparameter	Setting
Training, testing, and validation set ratio	70%, 15%, and 15%
Batch size	32 samples
Maximum training epoch	500 epochs
Callback condition	Continue testing loss increase over 10 epochs
Loss function	MSE
Evaluation metric	RMSE and MAE
Optimizer	Adam from TensorFlow

Table 4
The hyperparameters for the flow regime classification experiments.

Hyperparameter	Setting
Training, testing, and validation set ratio	70%, 15%, and 15%
Batch size	128 samples
Maximum training epoch	5000 epochs
Callback condition	Continue testing loss increase over 100 epochs
Loss function	CCE
Evaluation metric	Accuracy, RMSE, MSE, and MAE
Optimizer	Adam from TensorFlow

indicates the step length (l_{step}) regarding the window length (l_{window}), which refers to the overlap ratio of adjacent windows. The control group A (*ctr-A*) and *ex* study the effect of l_{window}, which adopts 1/4, 1/2, 1, and 2 times of l_{window} of the experimental group (*ex*), respectively. It is noteworthy that l_{window} determines the overall information amount in not only one sample but also the real-time performance of the solution. For example, *ctr-Ac* means that it requires twice the length of l_{signal} as the experimental group, and its real-time performance is also lower. Control group B (*ctr-B*) studies the $r_{s/w}$. A larger value of $r_{s/w}$ refers to a higher overlapping ratio between adjacent windows. When increasing the number of samples, it also brings the increasing risk of similarity among samples. The control group C (*ctr-C*) studies the compression rate of the proposed SSFE, that is, the dimensions of the output SSF (d_{SSF}). A larger value of d_{SSF} refers to a lower compression rate and more information the SSF can contain. However, a large value of d_{SSF} is opposite to dimension reduction and information compression for feature extraction, and a large SSF means that the complexity of the classifier also becomes high.

The *ex* and *ctr* refer to the experimental and control groups. $r_{Tr/Te/Va}$ refers to the ratio of training, testing, and validation set.

$$l_{step} = l_{window} \times rS/W \tag{1}$$

$$n_{SSF} = \left[\frac{l_{signal} - l_{window}}{l_{step}}\right] \tag{2}$$

4.1. SSFE experiments

The SSFE algorithm extracts each frequency-domain signal in Section 3.2 into 2D self-supervised features (SSFs). The loss function for SSL is the mean square error (MSE) (see Eq. (3)). This study further applies the root mean square error (RMSE) and mean absolute error (MAE) as additional loss function supervision (see Eqs. (4) and (5)). In Eqs. (3), (4), and (5), $\widetilde{y}$ and y are the corresponding predictions and annotation values, and $\widetilde{y}_i$ and y_i are the i-th predictions value and true value, n is the number of y_i in y. RMSE is more sensitive to single outliers in the summation than MAE.

$$MSE(y,\widetilde{y}) = \frac{1}{n} \times \Sigma_{i=0}^{n-1}\left(y_i - \widetilde{y}_i\right)^2 \tag{3}$$

$$RMSE(y,\widetilde{y}) = \sqrt{MSE(y,\widetilde{y})} = \sqrt{\frac{1}{n} \times \sum_{i=0}^{n-1}\left(y_i - \widetilde{y}_i\right)^2} \tag{4}$$

$$MAE(y,\widetilde{y}) = \frac{1}{n} \times \Sigma_{i=0}^{n-1}\left|y_i - \widetilde{y}_i\right| \tag{5}$$

Table 3 lists the hyperparameters of the SSFE experiments. The training, testing, and validation sample numbers are 6648, 1423, and 1429, respectively. The batch size is based on the hardware capability used in this study. Theoretically, a larger batch can better represent the distribution of the overall dataset. The maximum training epoch is set to 500, but the callback condition can be triggered when testing loss continuously increases over ten epochs. When the callback condition is triggered, the training is stopped to prevent overfitting, and the model from ten epochs ago is saved. The reason for choosing RMSE and MAE as additional evaluation metrics is to avoid overfitting the model to the loss function (MSE). Equations (3) and (4) have pointed out the difference in calculation principles between MSE and MAPE. The model has reached the best convergence point. Adam is a very stable optimizer widely used in similar research (Kuang et al., 2021, 2022; Nnabuife, Kuang, Rana, et al., 2021; Nnabuife, Kuang, Whidborne, et al., 2021b, 2021a).

4.2. Flow regime classification experiments

Table 4 lists the hyperparameters for the flow regime classification experiments. The training, testing, and validation set use 70%, 15%, and 15% of the classification dataset configured in Section 3.2, respectively. Notably, although the subset division ratios in Table 3 and Table 4 are the same, they are not the same dataset (see discussion in Section 3.2). The batch size uses 128 samples per batch instead of 32 samples per batch in SSFE, which indicates that feature extraction can reduce the required computational cost. If the source data is used directly, 32 samples per batch have reached the hardware limitation applied in this study, while feature extraction increases the number of samples in each batch by four times. The more samples per batch, the better it can represent the specific distribution of the dataset, and the available computational cost limits the sample number per batch. Especially considering the potential onboard situation, the computational cost and energy consumption are very limited. An excellent feature extraction algorithm dramatically improves the embeddability and real-time performance of flow regime classification solutions.

CCE, RMSE, MSE, and MAE refer to categorical cross-entropy, root mean square error, and mean absolute error, respectively.

The data for flow regime classification are SSFs achieved from Section 4.1. The flow regime classification task is a multi-classification. Since the one-hot encoding is used, the training loss function uses categorical cross-entropy (see Eq. (6)). This study adopted Accuracy, RMSE, and MAE as evaluation metrics (see Eqs. (4) and (5)). The mathematical properties of RMSE and MAE have been described in Section 4.1. Compared with the feature extraction experiments, Accuracy is a new evaluation metric. Accuracy is a very intuitive evaluation criterion, and its value range is a percentage from 0% to 100%. The classifier's output uses a SoftMax activation function (see Eq. (7)), which limits the sum of all outputs to one, and a single output represents the possibility of the corresponding category. The lower the cross entropy value, the closer the output probability distribution is to the ground truth. The Accuracy selects the category corresponding to the largest output node as the predicted class, while the Accuracy is not very sensitive to the predicted value. In Eqs. (6) and (7), *CCE*, *log*, *softmax*, and *e* refer to the categorical cross-entropy loss function, logarithm, SoftMax activation function, and exponent, respectively.

$$CCE(y,\widetilde{y}) = -\frac{1}{n}\Sigma_{i=1}^{n}\left(y_i \times log\left(\widetilde{y}_i\right) + (1-y_i) \times log\left(1-\widetilde{y}_i\right)\right) \tag{6}$$

Table 5
The values of MSE, RMSE, and MAE in the training, testing, and validation sets.

Metrics	loss (MSE)			RMSE			MAE		
Unit	10^{-5}			10^{-2}			10^{-3}		
Subset Index	Train	Test	Valid	Train	Test	Valid	Train	Test	Valid
ex	1.50	1.50	11.64	0.39	0.39	1.08	2.10	2.10	2.15
ctr-Aa	4.85	4.85	51.78	0.70	0.70	2.28	3.60	3.60	4.10
ctr-Ab	27.15	27.15	46.80	1.65	1.65	2.16	3.00	3.00	3.27
ctr-Ac	0.82	0.80	11.88	0.29	0.28	1.09	1.60	1.60	2.31
ctr-Ba	2.90	2.85	18.74	0.54	0.53	1.37	3.60	3.50	3.56
ctr-Bb	1.48	1.49	11.70	0.38	0.39	1.08	2.03	2.04	2.15
ctr-Ca	1.58	1.56	16.29	0.40	0.39	1.28	2.10	2.10	2.18
ctr-Cb	310.00	310.00	319.52	5.56	5.56	5.65	5.10	5.10	5.21

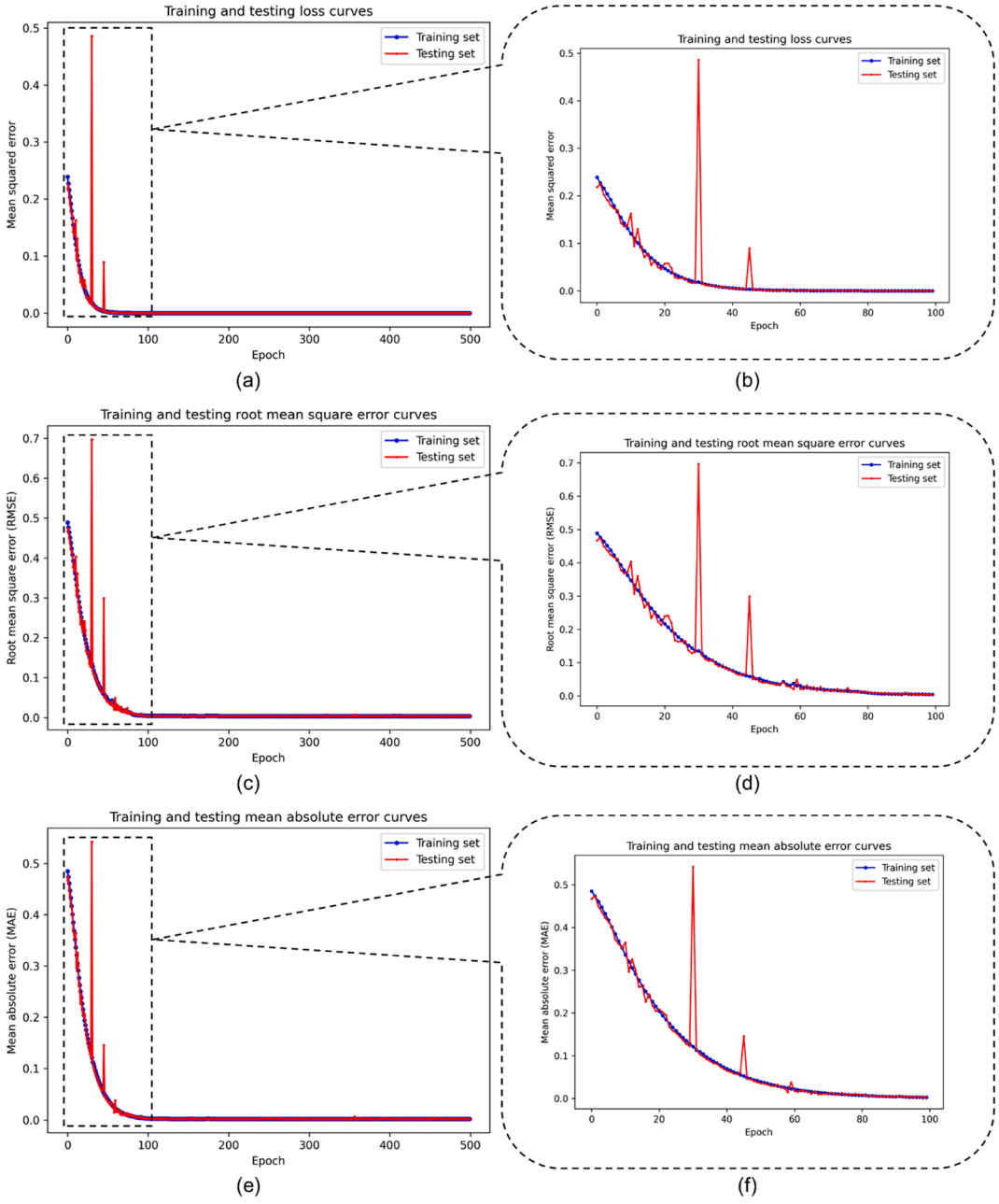


Fig. 8. The loss, RMSE, and MAE curves of the self-supervised training for SSFE.

Table 6
The experimental results of CCE, Accuracy, RMSE, and MAE of experimental and control groups.

Metric	Loss (CCE)			Accuracy			RMSE			MAE		
Unit	10^{-2}			%			10^{-3}			10^{-2}		
Index	Train	Test	Valid	Train	Test	Valid	Train	Test	Valid	Train	Test	Valid
ex	4.52	8.05	6.07	98.50	97.34	98.05	7.68	10.44	9.55	1.49	1.94	1.68
ctr-Aa	13.07	20.78	18.54	94.85	92.15	92.70	13.64	17.11	16.20	3.92	4.94	4.58
ctr-Ab	7.55	12.44	13.38	97.34	95.42	95.50	10.05	13.02	13.36	2.34	2.87	2.94
ctr-Ac	1.72	2.80	4.04	99.52	98.96	98.09	4.32	6.22	7.88	0.66	0.81	1.21
ctr-Ba	7.71	15.19	10.80	97.44	93.38	95.12	10.06	15.14	12.45	2.41	3.87	2.92
ctr-Bb	1.98	2.42	3.12	99.23	99.01	98.84	5.21	5.90	6.63	0.63	0.63	0.74
ctr-Ca	7.79	8.46	6.08	96.87	96.98	97.69	10.40	10.91	9.43	2.35	2.27	1.88
ctr-Cb	2.30	8.46	5.56	99.47	97.19	98.25	4.82	10.51	8.30	0.90	1.87	1.26

$$softmax(x) = \frac{1}{1+e^{x}} \tag{7}$$

5. Results and discussion

The results are divided into two parts for discussion. Section 4.1 discusses the results of the proposed SSFE algorithms. The results of the SSFE algorithm are discussed using the experimental group (*ex*) and the control groups (*ctr*) in Section 4. Section 5.2 further utilizes the flow regime classification to evaluate the performance of self-supervised features (SSFs). The experimental results are also compared between the experimental group and the control groups, and further comparison to the related advanced studies is made to verify the novelty of this study.

5.1. Self-supervised feature

Table 5 lists the MSE, RMSE, and MAE values in the training, testing, and validation sets. Table 5 characterizes the degree of information loss after the proposed SSFE algorithm. The original data of the riser experiment contains important irrelevant noise information. After the frequency domain transformation, the relevant flow regime information is concentrated in the low-frequency band, while the irrelevant noise is concentrated in the high-frequency band. Choosing a proper filter is challenging, significantly impacting the remaining relevant information and filtering out irrelevant information. Kuang, Nnabuife, et al. (2021), Nnabuife, Kuang, Rana, et al. (2021), and Nnabuife, Kuang, Whidborne, et al. (2021b, 2021a) use the low-pass filter to directly segment the frequency band, which lacks flexibility for various inputs. SSL realizes the filter functionality through an encoder-decoder design. Notably, in the frequency domain, the low-frequency bands have a higher value than the high-frequency bands, and their impact on the loss function is also higher. l_{window} of the *ctr-Aa*, *ctr-Ab*, *ctr-Ac*, and *ex* gradually increases, and *ex* obtains a loss much lower than *ctr-Aa* and *ctr-Ab*. However, although *ctr-Ac* obtains the lowest values on the training and testing sets, its validation loss is higher than the *ex*. Besides, SSFE also compresses the input while filtering out irrelevant information. Excessive compression can also cause relevant information to be lost. The length of the *ctr-Ac* input signal is 131,072, while the d_{SSF} is 64×64, and the compression rate is 32 times. However, the values in Table 5 can only represent the ability of SSFE to encode and decode signals, and the performance of the proposed SSFE algorithm requires further evaluation by the flow regime classification experiments in Section 5.2.

Fig. 8 illustrates the loss, RMSE, and MAE curves of the self-supervised training for SSFE. It can be found that the initial declines of the curves are rapid, and then they step into a slow learning process. Fig. 8(a), (c), and (e) are complete curves of loss, RMSE, and MAE, and Fig. 8(b), (d), and (f) is the enlarged curves between the No.0 and No.100 epoch. The increase in the loss curve indicates overfitting, which means that the generalization ability of the trained model has declined. However, determining the overfitting is a difficult problem because the training curve also has many local fluctuations with random gradient descent. For example, in Fig. 8, the learning curve seems flat since the No.100 epoch, while its trend is actually still slowly declining. Therefore, a callback condition is set in Section 4.1, which guarantees that the best model can be preserved when overfitting is sure to be detected. Fig. 8(b), (d), and (f) show that RMSE and MAE enter the slow learning condition later than MSE (loss). The RMSE and MAE curves start to be flattening at around the No.80 to 100 epoch, while the MSE is about the No.50 epoch. Besides, the RMSE is more sensitive to outliers during the accumulation (Σ), so RMSE's volatility in the testing set is stronger than in the MAE. In Fig. 8, the sharp fluctuations and spikes of the verification curve at the early stage of SSL can eliminate the potential risk of information leakage from the training set to the testing set. The values of the validation set in Table 5 differ significantly from the training and testing sets, which can also eliminate the risk of information leakage from the training set to the validation set.

The self-supervised feature extractor (SSFE) extracts the configured data in Section 3.2 to 2D SSFs. The loss function used for achieving SSFE is a mean square error (MSE) (see Equation (3)). This study further applies root mean square error (RMSE) and mean absolute error (MAE) as different evaluation metrics (see Eqs. (4) and (5)). Fig. 8(a), (b), and (c) illustrate the MSE, RMSE, and MAE curves. MSE becomes smoother than the RMSE and MAE curves, so utilizing MSE as the loss function can improve the convergence closer to the global minimum. Although Fig. 8 (b) and (c) seem similar, the RMSE is more sensitive to singular values than MAE. Notably, MSE, RMSE, and MAE refer to the function names; $\widetilde{y}$ and y refer to the predicted and real values, respectively; $\widetilde{y}_i$ and y_i refer to the ith predicted value and real value, respectively; and n refers to the number of $\widetilde{y}_i$ in $\widetilde{y}$.

Fig. A.1 in the Appendix shows some random visualization examples of the achieved self-supervised features (SSFs) from the validation set. Each row in Fig. A.1 is associated with a flow regime, corresponding to annotations 1, 2, 3, and 4 in Section 3.2. Notably, although Fig. A.1 visualizes self-supervised features, the SSFs are essentially compressed 2D matrixes that guarantee the minimum loss between SSFE's inputs and outputs. Quantitative interpretation of these 2D matrixes relates to another significant topic named interpretable artificial intelligence, while it is not the focus of this study and this manuscript does not further discuss. These SSFs correspond to the extension of the one-dimensional input through the convolutions in the SSFE's encoder part, so the visualized results appear as an obvious pattern of vertical stripes. This study is the first successful attempt to use an SSL strategy in the feature extraction of two-phase flow ultrasound signals. The compression process of feature extraction inevitably leads to information loss due to the data volume reduction. Compared with artificial features, self-supervised features can continuously encode and compress the features of the original signal through the average poolings in the neural network. The average pooling does not simply discard any input data but integrates it into the next layers. Section 5.2 uses a typical flow regime classification network to verify the performance of the proposed SSFE.

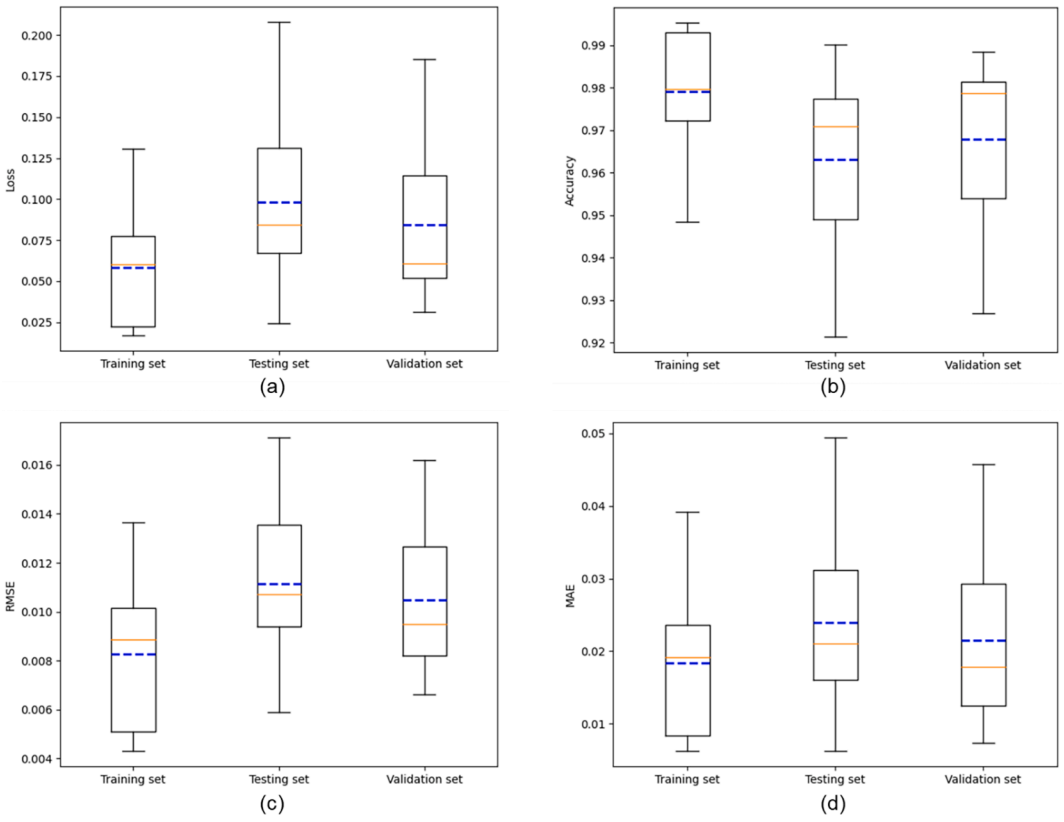


Fig. 9. The box and whiskers diagrams of experiment groups (ex) and control group (ctr) results. (a), (b), (c), and (d) refer to the box and whiskers diagrams of loss (CCE), Accuracy, RMSE, and MAE values in Table 6, respectively. Blue dash lines and solid orange lines indicate the mean and median values. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

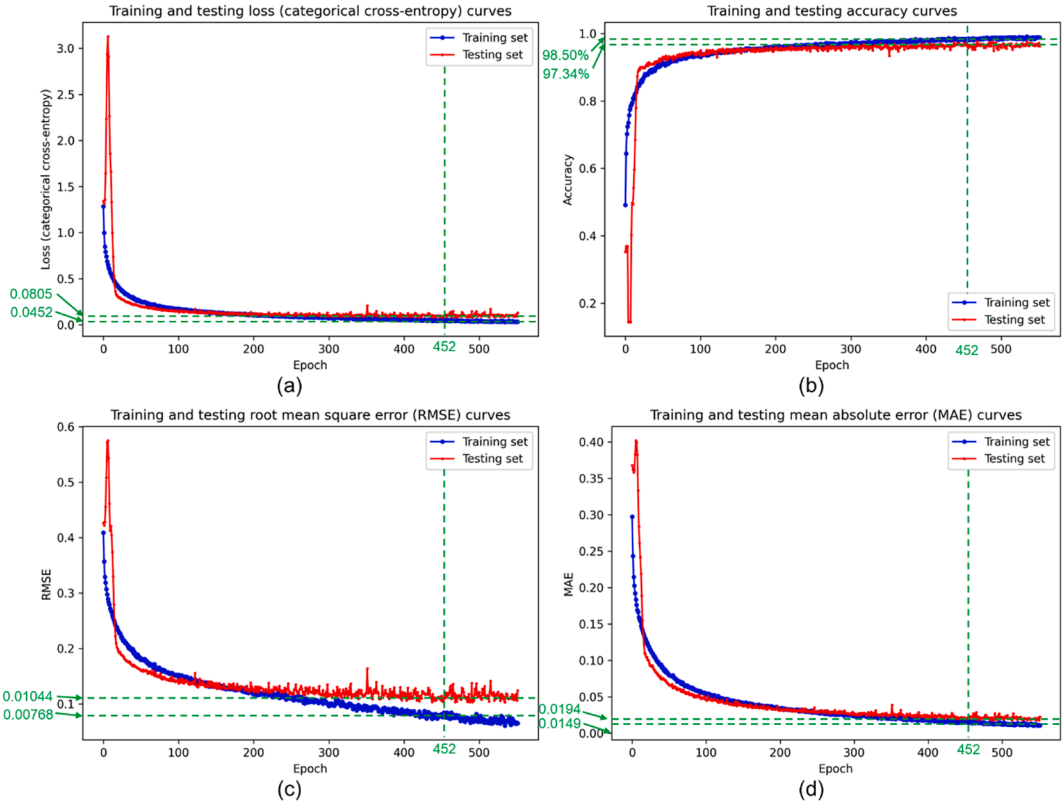


Fig. 10. The loss (CCE), accuracy, RMSE, and MAE curves for the training set and testing set.

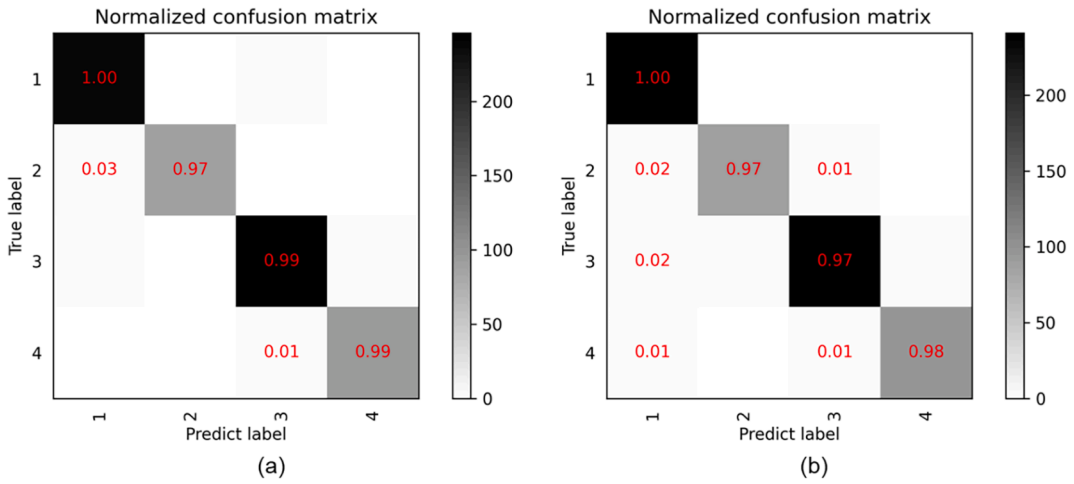


Fig. 11. The normalized confusion matrixes in the testing set and validation set. (a) and (b) refers to the normalized confusion matrixes in the testing set and validation set, respectively.

5.2. Flow regime classification

This study focuses on studying the potentiality and performance of SSL in the flow regime feature extraction with a competitive flow regime classification performance. Therefore, this study does not excavate the maximum performance of the proposed SSFE through parameter tuning, which may vary slightly in further implementation depending on the specific situation. This study explores the impact of different SSFE algorithm settings on the flow regime classification through the experimental group (*ex*) and the control groups (*ctr*).

Table 6 compares the experimental group (*ex*) and the control group (*ctr*) on the four metrics. Except for *ctr-Ab*, *ctr-Ac*, and *ctr-Ba*, the validation accuracies have reached more than 98%. When l_{window} goes to 65,536, the validation accuracy of *ctr-Bb* achieves the highest among all experiments. This shows that increasing the n_{SSF} for the SSL process can improve the flow regime classification accuracy to a certain extent. From *ctr-Ba* (4,625 samples) to the *ex* (9,418 samples), the validation accuracy increased by 2.93%. However, from the *ex* (9,418) to *ctr-Bb* (18,750), the validation accuracy only improves by 0.79%. However, although *ctr-Ab* and *ctr-Ac* have much higher sample sizes than other experiments, their validation accuracies are still low. This shows that the l_{window} significantly influences the flow regime classification performance, and *ctr-A* also corroborated this situation. However, the l_{window} of *ctr-Ac* is twice that of the *ex*, while an accuracy improvement is only 0.04%. This shows that the effect of further improving l_{window} is no longer significant. *ctr-C* shows that increasing the d_{SSF} can increase the flow regime classification performance, but this also reduces the information compression ability of feature extraction and increase the computing power requirements of the classification.

Fig. 9 applies box and whiskers diagrams to illustrate the value distribution of the four metrics among training, testing, and validation sets. The diagram are generated using the boxplot function of the Matplotlib library, and the parameters apply the default settings. The median and mean values are close to the minimums in loss (Fig. 9(a)), RMSE (Fig. 9(c)), and MAE (Fig. 9(d)), while the median and mean accuracies are close to the maximums in Fig. 9(b). The box position corresponding to the whiskers is close to the highest performance, which indicates the overall experiment optimization is in the correct direction to achieve convergence. The quartiles show the testing and validation results have more extensive data distribution and variability than the training results.

This study uses training curves and confusion matrices to evaluate

Table 7
The flow regime classification performance of the achieved self-supervised feature in recent studies, the representative experiments in the baseline studies, and this study.

Metric		CCE	Accuracy	RMSE	MSE	MAE	Complexity
	Unit						
Solution		10^{-2}	%	10^{-2}	10^{-3}	10^{-2}	10^{5}
Recent studies	(Nnabuife, Pilario, et al., 2019)	n/a	84.60	n/a	n/a	n/a	n/a
	(Nnabuife, Sharma, et al., 2021)	6.84	96.80	n/a	n/a	n/a	n/a
	(Nnabuife, Kuang, Whidborne, et al., 2021a)	n/a	96.35	12.61	0.0159	n/a	n/a
	(Nnabuife, Kuang, Whidborne, et al., 2021b)	12.87	96.94	n/a	n/a	n/a	n/a
Baseline studies	(Kuang et al., 2021)	1.32	99.54	4.12	0.0017	n/a	n/a
	Ex. No. 7 in (Kuang et al., 2022)	n/a	98.06	15.17	0.0230	n/a	171.10
	Ex. No. 11 in (Kuang et al., 2022)	n/a	93.06	21.56	0.0465	n/a	1.18
Our	**Our** *ctr-Aa*	18.54	92.70	16.20	1.62	4.58	7.71
	Our *ctr-Ab*	13.38	95.50	13.36	1.34	2.94	7.71
	Our *ctr-Ac*	4.04	98.09	7.88	6.21	1.21	7.71
	Our *ctr-Ba*	10.80	95.12	12.45	15.50	2.92	7.71
	Our *ctr-Bb*	3.12	98.84	6.63	4.40	0.74	7.71
	Our *ctr-Ca*	6.08	97.69	9.43	8.89	1.88	4.53
	Our *ctr-Cb*	5.56	98.25	8.30	6.89	1.26	8.46
	Our *ex*	6.07	98.05	9.55	9.12	1.68	7.71

Table 8
The number of **Our** experiments that reach the Alternative Hypothesis (*max-capability*, *75%-capability*, and *min-capability*).

Metric Hypothesis	CCE	Accuracy	RMSE	MSE	MAE	Complexity
maxcapability	1.32×10^{-2}	99.54%	4.12×10^{-2}	1.7×10^{-3}	n/a	1.18×10^{5}
N_{alter}	0	0	0	0	n/a	0
75 %capability	4.2075×10^{-2}	95.805%	8.49×10^{-2}	12.9×10^{-3}	n/a	43.58×10^{5}
N_{alter}	2	6	3	7	n/a	8
mincapability	12.87×10^{-2}	84.60%	21.6×10^{-2}	46.5×10^{-3}	n/a	171.10×10^{5}
N_{alter}	6	8	8	8	n/a	8

N_{alter} refers to the number of **Our** experiments that reach the *Alternative Hypothesis*.

the flow regime classification results. Fig. 9(a), (b), (c), and (d) show the loss (CCE), accuracy, RMSE, and MAE curves for the training set and testing set, respectively. The overall curves show a trend of rapid learning and then slow convergence. In the initial rapid learning phase, the testing curves all show violent fluctuations, which indicates that the testing and validation sets are not highly consistent in feature distribution. A significant challenge in dataset construction (Section 3.2) is the information leakage between training, testing, and validation sets, which can cause the feature distributions among training, testing, and validation sets to be highly consistent. A highly consistent feature distribution leads to a nearly consistent training trend across the training curves. However, the apparent difference between the training and testing curves in Fig. 10 can eliminate the potential risk of information leakage for unreliable testing and validation results. Regarding the callback condition in Section 4.2, Fig. 10 shows the convergence of the model training results using green dashed lines and values. The sharp fluctuation of the RMSE curves (Fig. 10(c)) comes from the influence of individual outliers in the accumulation process (Σ). Therefore, although the four criteria in Table 6 and Fig. 10 all show good numerical results, they can only refer to the overall performance.

This study demonstrates the performance of each flow regime on test and validation sets using confusion matrices (see Fig. 11). The annotations (“1″, ”2″, “3″, and ”4″) represent annular, churn, slug, and bubbly flow regimes, respectively. The proposed SSF achieves the highest Accuracy for the annular flow regime and the lowest Accuracy for the churn and slug flow regimes. However, the single-category Accuracy is higher than 97% in all four flow regimes. It is noteworthy that the false-positive rate is mainly concentrated on the annular flow regime, which also means that the overall classifier has a very slight data skew to the annular flow. At the same time, the trend is only 1–2%.

Table 7 utilizes six metrics to compare the *ex* and *ctrs* experiments in this study to the recent studies and the representative experiments in baseline studies. Three criteria restrict the baseline selected in this study. First, the baseline study must apply the same dataset as in this study because of the comparability. Second, the baseline study must contain systematic experiments design and evaluation metrics to provide a comprehensive investigation. Third, the baseline study must be peer-reviewed and accessible for technical details. Table 7 only archives the validation results. Although Kuang et al. (2022) do not claim to be a baseline, it meets all three baseline selection criteria. Furthermore, Kuang et al. (2022) systematically studied the parameter quantity among various classifiers, which provided an essential approach to evaluating the model complexity.

Table 7 shows that the proposed SSFE algorithm has achieved competitive performance to the supervised learning algorithms in recent studies and baseline studies. *ctr-Aa* (with the lowest Accuracy) also surpassed the results in Nnabuife, Pilario, et al. (2019), while the *ex* and *ctr-Bb* achieved better performance than Nnabuife, Kuang, Whidborne, et al. (2021b, 2021a), Nnabuife, Pilario, et al. (2019), and Nnabuife, Sharma, et al. (2021) and the experiment No. 11 in Kuang et al., (2022). The *ex* is only 0.01% different from the Accuracy of experiment No. 7 in Kuang et al. (2022) and 1.49% different from Kuang et al. (2021). The complexity of the experiment No. 7 in Kuang et al. (2022) is 22 times that of the *ex*. Although Kuang et al. (2021) does not mention its complexity, according to the discussion in Kuang et al. (2022), the complexity of Kuang et al. (2021) classifier is very high (Kuang et al., 2022). Besides, this study uses SSL, which has a lower supervision level and minimum annotation requirements. Without supervision, this study has achieved very competitive flow classification results (with a difference of less than 1.5% of the accuracy difference and a very low classifier complexity). The proposed solution intentionally focused on exploring an unsupervised approach, aiming to address the challenge without relying on supervision. This approach allows for greater flexibility and potential scalability in real-world applications where labeled data might be limited or costly. The novelty and potential advantages of an unsupervised solution in terms of reduced dependency on labeled data is the significant contribution of this study.

This study further conducts the nonparametric statistical analysis to evaluate the competitiveness of the proposed solution, and the process follows the tutorial from Derrac et al. (2011). The specific method applied is the *Sign test* of the *Pairwise comparisons*. This study applies three hypothesis standards to nonparametric statistical analysis of competitiveness, which are named *maxcapability*, *75 %capability*, and *mincapability*. The *maxcapability* refers to “the proposed solution defeats the maximum capability compared to recent and baseline studies.” The *75 %capability* refers to “the proposed solution exceeding 75% of the maximum capability compared to recent and baseline studies.”, which is calculated through Equation (8). The *mincapability* refers to “the proposed solution achieving the minimum capability compared to recent and baseline studies.” Considering the recent studies and baseline studies are all selected from recent and advanced studies, this study defines the competitiveness with the *Null Hypothesis* and *Alternative Hypothesis*. The alternative hypothesis refers to “if any of the *ctr* or *ex* results defeat the hypothesis standards, the proposed solution achieves the competitive performance.”, while the *Null Hypothesis* is otherwise. Table 8 depicts the number of **Our** experiments that reach the *Alternative Hypothesis*. For example, the value of N_{alter} of Accuracy of *75 %capability* is 6, which means six experiments reached the *75 %capability* hypothesis. According to the critical values in Derrac et al. (2011), if N_{alter} is not less than 7, the proposed solution is significant for the two-tailed sign test at $\alpha = 0.05$. Table 8 first shows that the proposed solution is not significant for *maxcapability*. So, the proposed solution does not defeat the highest achievement in recent and baseline studies. Second, the proposed solution shows significance in terms of MSE and Complexity for *75 %capability*. So, the proposed solution achieves 75% performance in terms of the MSE and Complexity compared to the recent and baseline studies. Third, the proposed solution shows significance in terms of all metrics for *mincapability*. So, the proposed solution achieves better performance than the minimum capability compared to the recent and baseline studies. Therefore, the proposed solution achieves competitive performance compared to advanced studies.

$$75\%capability = 75\% \times (|maxcapability - mincapability|) + \{maxcapability, mincapability\}_{min} \quad (8)$$

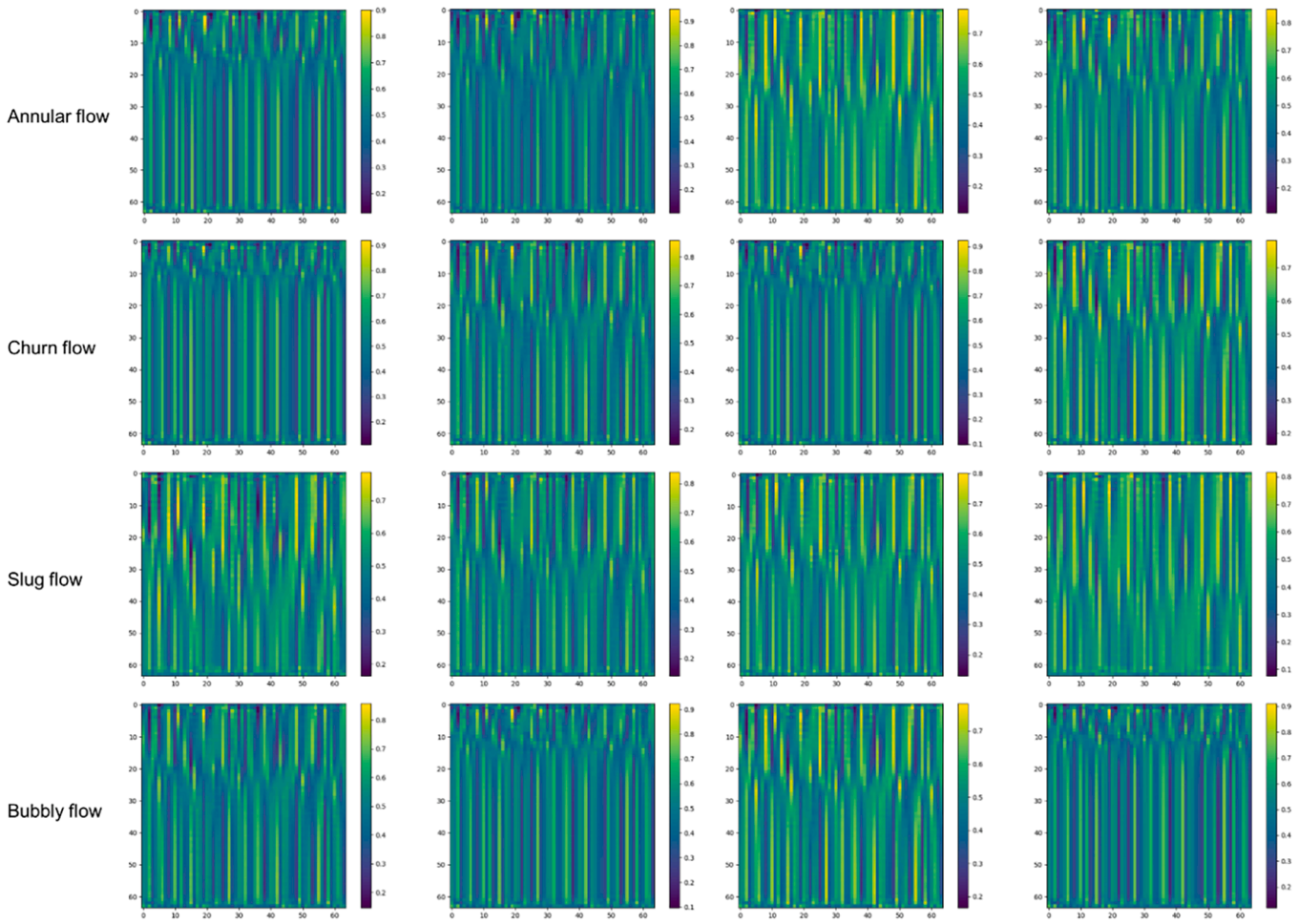


Fig. 12. Some random visualization examples of the achieved self-supervised features (SSFs) from the validation set in the proposed SSFs dataset [1] (Kuang, 2023).

6. Conclusion and future work

This study demonstrates the feasibility and advantages of SSL for ultrasonic signal processing in flow regime identification. By leveraging the benefits of SSL, this study bypasses manual annotation for the feature extraction step, which can effectively save time/labor costs and avoid human error. It makes this approach an attractive option for practical engineering problems that lack sufficient annotations. This study discussed the feasibility and advantage of SSL for flow regime identification using ultrasonic signals through a pipeline of riser experiments, dataset construction, feature extraction, and signal classification. This study proposes a self-supervised feature extraction (SSFE) algorithm to extract the self-supervised features (SSFs), and SSFs are opened to the community. A CNN-based classifier is then applied to evaluate the SSFE performance systematically. Our study shows that an SSFE algorithm significantly improves feature extraction and achieves competitive performance in flow regime classification. In terms of the results, the evaluation metrics, including Accuracy, CCE, RMSE, MSE, MAE, and individual Accuracy of each flow regime, are reported. The results show that the SSFs achieved an accuracy of 98.84%, CCE of 0.00312, RMSE of 0.00663, MSE of 0.00044, and MAE of 0.00074, and individual accuracies of each flow regime were above 97%.

However, some limitations exist in the proposed study. First, the proposed solution has a restricted number of riser experiments and experimental settings. The practical experiment requires a costly compressor, riser system, and experiment site, which is challenging for individual teams to obtain data from a wide variety of riser shapes and gas–liquid ratios. Increased access to diverse datasets can support further research and enable cross-cooperation among different groups, which can help overcome the limitations faced by individual research teams. Second, the proposed solution did not implement a complete self-supervision solution for flow regime identification. Although the proposed solution executes self-supervision in the feature extraction step, the overall flow regime classification still applies supervised learning. However, a complete self-supervision solution can bypass manual intervention entirely.

Future works can explore the model transfer capability of ultrasonic signals across various riser shapes, such as vertical or horizontal risers. Techniques like transfer learning and multi-domain adaptation (MDA) could be employed to investigate the applicability and performance of existing models trained on S-shaped risers to these different configurations. This study suggests that more diverse and comprehensive datasets should be made available to the community for two-phase flow regime classification. This includes datasets encompassing a wide range of gas–liquid ratios, flow rates, and various pipe geometries. Access to such datasets would enable researchers to conduct extensive testing and comparison of different classification models, facilitating the development of more accurate and robust solutions. Furthermore, it would be intriguing to investigate other self-supervised learning (SSL) techniques, such as contrastive learning or generative models, for flow regime identification tasks. Contrastive learning can facilitate the learning of meaningful representations by contrasting positive and negative samples, while generative models can capture the underlying data distribution. These approaches hold promise in enhancing ultrasonic signal processing and classification by providing additional insights and improved performance.

CRediT authorship contribution statement

Boyu Kuang: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Writing – review & editing, Visualization. **Somtochukwu G. Nnabuife:** Investigation, Resources, Data curation, Writing – original draft, Writing – review & editing, Visualization. **James F. Whidborne:** Writing – review & editing, Supervision, Project administration. **Shuang Sun:** Formal analysis, Investigation, Writing – review & editing. **Junjie Zhao:** Validation, Formal analysis, Writing – review & editing. **Karl Jenkins:** Resources, Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

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.

Data availability

Data available in the Cranfield University Research Data repository - doi included in list of references

Appendix

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., … Zheng, X. (2016). TensorFlow: Large-scale machine learning on heterogeneous distributed systems. *The Library, s4-X*(3), 339. http://arxiv.org/abs/1603.04467.

Abbagoni, B. M., & Yeung, H. (2016). Non-invasive classification of gas–liquid two-phase horizontal flow regimes using an ultrasonic Doppler sensor and a neural network. *Measurement Science and Technology, 27*(8), Article 084002. https://doi.org/10.1088/0957-0233/27/8/084002

Arif, M., Hassan, H., Nasien, D., & Haron, H. (2015). A review on feature extraction and feature selection for handwritten character recognition. *International Journal of Advanced Computer Science and Applications, 6*(2). https://doi.org/10.14569/IJACSA.2015.060230

Balachandar, S., Zaleski, S., Soldati, A., Ahmadi, G., & Bourouiba, L. (2020). Host-to-host airborne transmission as a multiphase flow problem for science-based social distance guidelines. *International Journal of Multiphase Flow, 132*, Article 103439. https://doi.org/10.1016/j.ijmultiphaseflow.2020.103439

Bécue, A., Praça, I., & Gama, J. (2021). Artificial intelligence, cyber-threats and Industry 4.0: Challenges and opportunities. *Artificial Intelligence Review, 54*(5), 3849–3886. https://doi.org/10.1007/s10462-020-09942-2

Buscher, S. (2019). Visualization and modelling of flow pattern transitions in a cross-corrugated plate heat exchanger channel with uniform two-phase distribution. *International Journal of Heat and Mass Transfer, 144*, Article 118643. https://doi.org/10.1016/j.ijheatmasstransfer.2019.118643

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation, 1*(1), 3–18. https://doi.org/10.1016/j.swevo.2011.02.002

Gupta, P., Kumar, P., & Rao, S. M. V. (2022). Artificial neural network based shape optimization of supersonic ejectors in the critical flow regime. *Applied Thermal Engineering, 216*, Article 119046. https://doi.org/10.1016/j.applthermaleng.2022.119046

Hammad, F. A., Sun, K., Che, Z., Jedelsky, J., & Wang, T. (2021). Internal two-phase flow and spray characteristics of outside-in-liquid twin-fluid atomizers. *Applied Thermal Engineering, 187*(January), Article 116555. https://doi.org/10.1016/j.applthermaleng.2021.116555

Hanus, R., Zych, M., Kusy, M., Jaszczur, M., & Petryka, L. (2018). Identification of liquid-gas flow regime in a pipeline using gamma-ray absorption technique and computational intelligence methods. *Flow Measurement and Instrumentation, 60*, 17–23. https://doi.org/10.1016/j.flowmeasinst.2018.02.008

Hashmi, K. A., Pagani, A., Stricker, D., & Afzal, M. Z. (2022). *BoxMask: Revisiting bounding box supervision for video object detection*. http://arxiv.org/abs/2210.06008.

Holland, F. A., & Bragg, R. (1995). Fluid flow for chemical engineers. *Elsevier*. https://doi.org/10.1016/B978-0-340-61058-9.X5000-2

Huang, Z., Wang, B., & Li, H. (2002). An intelligent measurement system for powder flowrate measurement in pneumatic conveying system. *IEEE Transactions on Instrumentation and Measurement, 51*(4), 700–703. https://doi.org/10.1109/TIM.2002.803296

Jiang, T., Gradus, J. L., & Rosellini, A. J. (2020). Supervised machine learning: A brief primer. *Behavior Therapy, 51*(5), 675–687. https://doi.org/10.1016/j.beth.2020.05.002

Kuang, B. (2023). SSFs dataset: Self-supervised features dataset of the ultrasonic signals of two-phase flow in an S-shaped riser. Cranfield Online Research Data (CORD). Cranfield University. https://dx.org/10.17862/cranfield.rd.22241530.v1.

Kuang, B., Nnabuife, S. G., & Rana, Z. (2021). Pseudo-image-feature-based identification benchmark for multi-phase flow regimes. *Chemical Engineering Journal Advances, 5*, Article 100060. https://doi.org/10.1016/j.ceja.2020.100060

Kuang, B., Nnabuife, S. G., Sun, S., Whidborne, J. F., & Rana, Z. A. (2022). Gas-liquid flow regimes identification using non-intrusive Doppler ultrasonic sensor and convolutional recurrent neural networks in an s-shaped riser. *Digital Chemical Engineering, 2*, Article 100012. https://doi.org/10.1016/j.dche.2022.100012

Li, Q., Peng, X., Cao, L., Du, W., Xing, H., Qiao, Y., & Peng, Q. (2020). Product image recognition with guidance learning and noisy supervision. *Computer Vision and Image Understanding, 196*, Article 102963. https://doi.org/10.1016/j.cviu.2020.102963

Li, Y., Shetty, P., Liu, L., Zhang, C., & Song, L. (2021). BERTifying the hidden Markov model for multi-source weakly supervised named entity recognition. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (Volume 1: Long Papers)*, 6178–6190. 10.18653/v1/2021.acl-long.482.

Lin, W., Tseng, B.-H., & Byrne, B. (2021). *Knowledge-aware graph-enhanced GPT-2 for dialogue state tracking*. http://arxiv.org/abs/2104.04466.

Lin, Z., Liu, X., Lao, L., & Liu, H. (2020). Prediction of two-phase flow patterns in upward inclined pipes via deep learning. *Energy, 210*, Article 118541. https://doi.org/10.1016/j.energy.2020.118541

Lube, G., Breard, E. C. P., Esposti-Ongaro, T., Dufek, J., & Brand, B. (2020). Multiphase flow behaviour and hazard prediction of pyroclastic density currents. *Nature Reviews Earth & Environment, 1*(7), 348–365. https://doi.org/10.1038/s43017-020-0064-8

Marcinkiewicz, M., Nalepa, J., Lorenzo, P. R., Dudzik, W., & Mrukwa, G. (2019). Segmenting brain tumors from MRI using cascaded multi-modal U-Nets. In *Lecture notes in computer science* (Vol. 11384, pp. 13–24). 10.1007/978-3-030-11726-9_2.

Murai, Y., Tasaka, Y., Nambu, Y., Takeda, Y., & Gonzalez, A. (2010). Ultrasonic detection of moving interfaces in gas–liquid two-phase flow. *Flow Measurement and Instrumentation, 21*(3), 356–366. https://doi.org/10.1016/j.flowmeasinst.2010.03.007

Nazeer, M., Hussain, F., Khan, M. I., & Khalid, K. (2022). Theoretical analysis of electrical double layer effects on the multiphase flow of Jeffrey fluid through a divergent channel with lubricated walls. *Waves in Random and Complex Media, 1–15*. https://doi.org/10.1080/17455030.2022.2126025

Neshatpour, K., Behnia, F., Homayoun, H., & Sasan, A. (2018). ICNN: An iterative implementation of convolutional neural networks to enable energy and computational complexity aware dynamic approximation. In *2018 Design, automation & test in europe conference & exhibition (DATE)*, 551–556. 10.23919/DATE.2018.8342068.

Nnabuife, S. G., Kuang, B., Rana, Z. A., & Whidborne, J. (2021). Classification of flow regimes using a neural network and a non-invasive ultrasonic sensor in an S-shaped pipeline-riser system. *Chemical Engineering Journal Advances.* , Article 100215. https://doi.org/10.1016/j.ceja.2021.100215

Nnabuife, S. G., Kuang, B., Whidborne, J. F., & Rana, Z. (2021a). 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, 403*, Article 126401. https://doi.org/10.1016/j.cej.2020.126401

Nnabuife, S. G., Kuang, B., Whidborne, J. F., & Rana, Z. A. (2021b). Development of gas-liquid flow regimes identification using a noninvasive ultrasonic sensor, belt-shape features, and convolutional neural network in an S-shaped riser. *IEEE Transactions on Cybernetics, 1–15*. https://doi.org/10.1109/TCYB.2021.3084860

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

Nnabuife, S. G., Sharma, P., Iyore Aburime, E., Lokidor, P. L., & Bello, A. (2021). Development of gas-liquid slug flow measurement using continuous-wave Doppler ultrasound and bandpass power spectral density. *ChemEngineering, 5*(1), 2. https://doi.org/10.3390/chemengineering5010002

Nnabuife, S. G., Tandoh, H., & Whidborne, J. F. (2022). Slug flow control using topside measurements: A review. *Chemical Engineering Journal Advances, 9*, Article 100204. https://doi.org/10.1016/j.ceja.2021.100204

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

Pan, S. J., & Yang, Q. (2010). A survey on transfer learning. In *IEEE transactions on knowledge and data engineering* (Vol. 22, Issue 10, pp. 1345–1359). 10.1109/TKDE.2009.191.

Pan, T., Chen, J., Pan, J., & Zhou, Z. (2020). A deep learning network via shunt-wound restricted Boltzmann machines using raw data for fault detection. *IEEE Transactions on Instrumentation and Measurement, 69*(7), 4852–4862. https://doi.org/10.1109/TIM.2019.2953436

Qian, M., Qi, J., Zhang, L., Feng, M., & Lu, H. (2019). Language-aware weak supervision for salient object detection. *Pattern Recognition, 96*, Article 106955. https://doi.org/10.1016/j.patcog.2019.06.021

Qiu, L., Li, H., Li, Z., & Wang, C. (2021). Residual grounding transformer network for terrain recognition on the lunar surface. *Applied Optics, 60*(21), 6002. https://doi.org/10.1364/AO.428232

Radev, S. T., Mertens, U. K., Voss, A., Ardizzone, L., & Kothe, U. (2022). BayesFlow: Learning complex stochastic models with invertible neural networks. *IEEE Transactions on Neural Networks and Learning Systems, 33*(4), 1452–1466. https://doi.org/10.1109/TNNLS.2020.3042395

Reed, C. J., Metzger, S., Srinivas, A., Darrell, T., & Keutzer, K. (2021). SelfAugment: Automatic augmentation policies for self-supervised learning. *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2021*, 2673–2682. https://doi.org/10.1109/CVPR46437.2021.00270

Reid Turner, C., Fuggetta, A., Lavazza, L., & Wolf, A. L. (1999). A conceptual basis for feature engineering. *Journal of Systems and Software, 49*(1), 3–15. https://doi.org/10.1016/S0164-1212(99)00062-X

Roshani, M., Phan, G. T. T., Jammal Muhammad Ali, P., Hossein Roshani, G., Hanus, R., Duong, T., Corniani, E., Nazemi, E., & Mostafa Kalmoun, E. (2021). Evaluation of flow pattern recognition and void fraction measurement in two phase flow independent of oil pipeline's scale layer thickness. *Alexandria Engineering Journal, 60*(1), 1955–1966. https://doi.org/10.1016/j.aej.2020.11.043

Roxas, R., II, Evangelista, M. A., Sombillo, J. A., Nnabuife, S. G., & Pilario, K. E. (2022). Machine learning based flow regime identification using ultrasonic Doppler data and feature relevance determination. *Digital Chemical Engineering, 3*, Article 100024. https://doi.org/10.1016/j.dche.2022.100024

Shen, X., & Hibiki, T. (2021). Distribution parameter and drift velocity for upward gas-liquid metal two-phase flow. *Applied Thermal Engineering, 184*(October 2020), Article 116242. https://doi.org/10.1016/j.applthermaleng.2020.116242

Shi, X., Tan, C., Dong, F., dos Santos, E. N., & da Silva, M. J. (2021). Conductance sensors for multiphase flow measurement: A review. *IEEE Sensors Journal, 21*(11), 12913–12925. https://doi.org/10.1109/JSEN.2020.3042206

Swan, R. M., Atha, D., Leopold, H. A., Gildner, M., Oij, S., Chiu, C., & Ono, M. (2021). AI4MARS: A dataset for terrain-aware autonomous driving on Mars. In *2021 IEEE/CVF conference on computer vision and pattern recognition workshops (CVPRW)* (pp. 1982–1991). https://doi.org/10.1109/CVPRW53098.2021.00226

Tan, C., Li, F., Lv, S., Yang, Y., & Dong, F. (2021). Gas-liquid two-phase stratified flow interface reconstruction with sparse batch normalization convolutional neural network. *IEEE Sensors Journal, 21*(15), 17076–17084. https://doi.org/10.1109/JSEN.2021.3081432

Tsai, C.-H., & Chang, J.-Y. (2021). A new approach to enhance artificial intelligence for robot picking system using auto picking point annotation. In *ASME 2021 30th conference on information storage and processing systems*. 10.1115/ISPS2021-65218.

Tsang, Y. P., & Lee, C. K. M. (2022). Artificial intelligence in industrial design: A semi-automated literature survey. *Engineering Applications of Artificial Intelligence, 112*, Article 104884. https://doi.org/10.1016/j.engappai.2022.104884

Villalobos, P., Sevilla, J., Heim, L., Besiroglu, T., Hobbhahn, M., & Ho, A. (2022). *Will we run out of data? An analysis of the limits of scaling datasets in Machine Learning*. 10.48550/arXiv.2211.04325.

Xu, H., Tang, T., Zhang, B., & Liu, Y. (2022). Identification of two-phase flow regime in the energy industry based on modified convolutional neural network. *Progress in Nuclear Energy, 147*, Article 104191. https://doi.org/10.1016/j.pnucene.2022.104191

Xu, Q., Li, W., Liu, W., Zhang, X., Yang, C., & Guo, L. (2020). Intelligent recognition of severe slugging in a long-distance pipeline-riser system. *Experimental Thermal and Fluid Science, 113*, Article 110022. https://doi.org/10.1016/j.expthermflusci.2019.110022

Xu, Q., Liu, C., Wang, X., Cao, Y., Yu, H., Li, W., & Guo, L. (2021). Machine learning classification of flow regimes in a long pipeline-riser system with differential pressure signal. *Chemical Engineering Science, 233*, Article 116402. https://doi.org/10.1016/j.ces.2020.116402

Zhang, Y., Azman, A. N., Xu, K.-W., Kang, C., & Kim, H.-B. (2020). Two-phase flow regime identification based on the liquid-phase velocity information and machine learning. *Experiments in Fluids, 61*(10), 212. https://doi.org/10.1007/s00348-020-03046-x

Zhang, Z., Tang, Z., Wang, Y., Zhang, Z., Zhan, C., Zha, Z., & Wang, M. (2021). Dense Residual Network: Enhancing global dense feature flow for character recognition. *Neural Networks, 139*, 77–85. https://doi.org/10.1016/j.neunet.2021.02.005

Zhou, Z.-H. (2018). A brief introduction to weakly supervised learning. *National Science Review, 5*(1), 44–53. https://doi.org/10.1093/nsr/nwx106

2023-09-07

Self-supervised learning-based two-phase flow regime identification using ultrasonic sensors in an S-shape riser

Kuang, Boyu

Elsevier

Kuang B, Nnabuife SG, Whidborne JF, et al., (2024) Self-supervised learning-based two-phase flow regime identification using ultrasonic sensors in an S-shape riser, Expert Systems with Applications, Volume 236, February 2024, Article Number 121414

https://doi.org/10.1016/j.eswa.2023.121414