UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

JEFFERSON DOS SANTOS AMBROSIO

ADVANCEMENTS IN TWO-PHASE FLOW REGIME CLASSIFICATION: A COMPARATIVE STUDY OF MACHINE LEARNING AND DEEP LEARNING APPROACHES USING WIRE-MESH SENSOR DATA

TESE

CURITIBA
2024

JEFFERSON DOS SANTOS AMBROSIO

ADVANCEMENTS IN TWO-PHASE FLOW REGIME CLASSIFICATION: A COMPARATIVE STUDY OF MACHINE LEARNING AND DEEP LEARNING APPROACHES USING WIRE-MESH SENSOR DATA

Avanços na classificação de escoamentos bifásicos: um estudo comparativo de abordagens de aprendizado de máquina e aprendizado profundo usando dados de sensores *Wire-Mesh*

Tese apresentada como requisito parcial para obtenção do título de Doutor em Engenharia Elétrica, do Programa de Pós-Graduação em Engenharia Elétrica, da Universidade Tecnológica Federal do Paraná (UTFPR).

Orientador: Prof. Dr. André Eugenio Lazzaretti
Coorientador: Prof. Dr. Marco José da Silva

CURITIBA

2024

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Curitiba

JEFFERSON DOS SANTOS AMBROSIO

ADVANCEMENTS IN TWO-PHASE FLOW REGIME CLASSIFICATION: A COMPARATIVE STUDY OF MACHINE LEARNING AND DEEP LEARNING APPROACHES USING WIRE-MESH SENSOR DATA

Trabalho de pesquisa de doutorado apresentado como requisito para obtenção do título de Doutor Em Ciências da Universidade Tecnológica Federal do Paraná (UTFPR). Área de concentração: Engenharia De Automação E Sistemas .

Data de aprovação: 16 de Dezembro de 2024

Dr. Adriano Todorovic Fabro, Doutorado - Universidade de Brasília (Unb)

Dr. Daniel Rodrigues Pipa, Doutorado - Universidade Tecnológica Federal do Paraná

Everton Luiz De Aguiar, - Universidade Tecnológica Federal do Paraná

Dr. Marco Jose Da Silva, Doutorado - Johannes Kepler Universität Linz (Jku)

Dr. Silvio De Barros Melo, Doutorado - Universidade Federal de Pernambuco (Ufpe)

Documento gerado pelo Sistema Acadêmico da UTFPR a partir dos dados da Ata de Defesa em 12/02/2025.

I dedicate this work to my friends, whose
unwavering support was crucial in maintaining
my mental well-being and motivation
throughout this journey. In particular, I dedicate
it to the memory of Weslei Vasconcelos, who
was like a brother to me and left this world far
too soon. His friendship was invaluable, and his
presence will always be deeply missed.

ACKNOWLEDGEMENTS

I express my deepest gratitude to my supervisors, Prof. Dr. Marco José da Silva and Prof. Dr. André Eugênio Lazzaretti and Prof. Dr. Daniel Pipa, who guided me during my master's degree and part of my doctoral studies. I am sincerely grateful for their patience, dedication, and invaluable guidance. From them, I also learned that pursuing high-quality research with passion while maintaining humanity and respect for others is possible—an invaluable lesson that I intend to carry with me throughout my career.

I am also immensely grateful to my partner, Daniely Lopes Silva, and my stepson, Eduardo Simeão Pallaro, for their patience and understanding when research and work became overwhelming. Their presence brought me joy and balance throughout this demanding process.

I extend my heartfelt thanks to all my friends, whose names I will not attempt to list to avoid the risk of unintentionally omitting anyone. Each has been truly special and played a vital role in this journey.

I am deeply thankful to my family for their unwavering support, care, and encouragement, which are essential in helping me reach this milestone.

Finally, I express my gratitude to the Federal University of Technology – Paraná and my colleagues in the graduate program for all the enriching discussions, insightful advice, and shared ideas that contributed to the development of this work.

“Logic will get you from A to B. Imagination
will take you everywhere."
(Albert Einstein)

RESUMO

AMBROSIO, Jefferson dos Santos. **Avanços na classificação de escoamentos bifásicos: um estudo comparativo de abordagens de aprendizado de máquina e aprendizado profundo usando dados de sensores *Wire-Mesh***. 2024. 101 f. Tese (Doutorado em Engenharia Elétrica) – Universidade Tecnológica Federal do Paraná. Curitiba, 2024.

A classificação do regime de escoamento é essencial para analisar e modelar escoamentos bifásicos, pois demarca o comportamento do escoamento e influencia a seleção de modelos preditivos apropriados. Abordagens baseadas em aprendizado de máquina ganharam relevância na pesquisa de classificação de regime de escoamento nos últimos anos. No entanto, elas ainda são solidamente baseadas na construção e definição cuidadosa de características. Abordagens de aprendizado profundo, por outro lado, podem fornecer soluções mais robustas e completas. No entanto, elas são pouco exploradas e não avaliaram a generalização dos modelos para outros dados ou sistemas de aquisição. Nesta tese, comparamos duas abordagens diferentes para classificar padrões de escoamento (agitação, bolhas e *slug*) usando séries temporais de fração de vazio de um sensor *wire-mesh*. Na primeira abordagem, definida como *MdG+MVS*, a série temporal é modelada como um processo estocástico de amostras independentes e distribuídas de forma idêntica com função de densidade de probabilidade descrita por um modelo de mistura gaussiana (MdG). Os parâmetros estimados da mistura são então alimentados em uma Máquina de Vetor de Suporte (MVS), produzindo a classificação do padrão de escoamento. Na segunda, definida como *EDA-AP*, propomos usar métodos de classificação de séries temporais de baseados em aprendizado profundo (ResNet, LSTM-FCN e TSTPlus) para padrões de escoamento de duas fases. Também apresentamos a análise de generalização dos modelos com experimentos entre conjuntos de dados, treinando o modelo com um conjunto de dados e testando-o com outro conjunto de dados coletados em outro sistema para dois conjuntos de dados, aqui definidos como HZDR e TUD. Os resultados demonstram que a abordagem baseada em aprendizado profundo (*EDA-AP*) apresenta métricas de classificação superiores em todos os casos avaliados, particularmente em experimentos entre conjuntos de dados. Com os métodos propostos, todas as métricas avaliadas (precisão e F1-Score) consistentemente ultrapassam 90% em todos os casos, enquanto o método *MdG+MVS* pode diminuir o desempenho abaixo de 80%. Isso demonstra a relevância da análise proposta aqui para a literatura de classificação de regime de escoamento e abre um novo conjunto de possibilidades para pesquisa nesta área, visando soluções robustas que sejam viáveis para uso prático.

Palavras-chave: Máquina de vetores de suporte. Escoamento bifásico. Aprendizado profundo. Séries Temporais. Modelo de mistura de Gaussianas.

ABSTRACT

AMBROSIO, Jefferson dos Santos. **Advancements in Two-Phase Flow Regime Classification: A Comparative Study of Machine Learning and Deep Learning Approaches Using Wire-Mesh Sensor Data**. 2024. 101 p. Thesis (PhD in electrical and computer engineering) – Universidade Tecnológica Federal do Paraná. Curitiba, 2024.

Flow regime classification is essential for analyzing and modeling two-phase flows, as it demarcates the flow behavior and influences the selection of appropriate predictive models. Machine learning-based approaches have gained relevance in flow regime classification research in the last few years. However, they are still solidly based on the construction and careful definition of hand-crafted features. Deep learning approaches, on the other hand, can provide more robust and end-to-end solutions. However, they are underexplored and have not evaluated the generalization of the models to other data or acquisition systems. In this thesis, we compare two different approaches for classifying flow patterns (churn, bubbly, and slug) using time series of void fraction from a wire-mesh sensor. In the first, defined as *MoG+SVM*, the time series is modeled as a stochastic process of independent and identically distributed samples with probability density function described by a Mixture of Gaussian (MoG) model. The estimated parameters of the mixture are then fed into a Support Vector Machine (SVM), yielding the flow pattern classification. The second, defined as *SOTA-DL*, we propose using end-to-end state-of-the-art (SOTA) time-series classification methods (ResNet, LSTM-FCN, and TSTPlus) for two-phase flow patterns. We also present the generalization analysis of the models with cross-dataset experiments, training the model with one dataset and testing it with another dataset collected in another system for two datasets, here defined as HZDR and TUD. The results demonstrate that the deep learning-based approach (*SOTA-DL*) presents superior classification metrics in all cases evaluated, particularly in cross-dataset experiments. With the proposed SOTA methods, all the evaluated metrics (accuracy and F1-Score) consistently surpass 90% in all cases, while the *MoG+SVM* method can decrease the performance under 80%. This demonstrates the relevance of the analysis proposed here for flow regime classification literature and opens up a new set of possibilities for research in this area, aiming at robust solutions that are viable for practical use.

Keywords: Support Vector Machine. Two-phase flow. Deep learning. Time series. Gaussian mixture model.

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

INITIALISM

ANN	Artificial Neural Networks
ANFIS	Adaptive Neuro-Fuzzy Inference System
BiLSTM	Bidirectional Long Short-Term Memory
BN	Batch Normalization
CFD	Computational Fluid Dynamics
CHMM	Continuous Hidden Markov Models
CNN	Convolutional Neural Networks
CPU	Central Processing Unit
CSV	Comma Separated Values
CV	Cross-Validation
DL	Deep Learning
DNN	Deep Neural Networks
EM	Expectation-Maximization Algorithm
FCN	Fully Convolutional Networks
GA	Genetic Algorithms
GMM	Gaussian Mixture Models
GPU	Graphics Processing Unit
HMM	Hidden Markov Models
HZDR	Helmholtz-Zentrum Dresden-Rossendorf
LSTM	Long Short-Term Memory
MID	Magnetic-Inductive Flowmeters
MLE	Maximum Likelihood Estimator
MLP	Multi-Layer Perceptron
MoG	Mixture of Gaussians
PCA	Principal Component Analysis
PDF	Probability Density Function
PLC	Programmable Logic Controller
RBF	Radial Basis Function
ResNet	Residual Network
SOTA	State of the Art
SVM	Support Vector Machine
TOPFLOW	Transient Two-Phase Flow
TSTPlus	Time Series Transformer Plus
TUD	Technische Universität Dresden

WMS	Wire-Mesh Sensor

CONTENTS

1 INTRODUCTION

Two-phase flow, the simultaneous flow of two immiscible substances, such as gas and liquid, is commonly found in various industries, including petroleum, chemical, and nuclear sectors. One of the most significant types of two-phase flow is liquid-gas flow. Accurate measurement and estimation of flow parameters, such as void fraction, bubble size distribution, phase velocities, and pressure drop, are critical for efficient system performance and process safety (GIOIA *et al.*, 2009; POELMA, 2020; NUNNO *et al.*, 2020; Kumar Vandrangi *et al.*, 2022). Of these parameters, the void fraction, defined as the fraction of the pipe area occupied by the gas phase, is significant in the study of two-phase flow systems.

Depending on the geometrical distribution of the gaseous and liquid phases, various flow regimes or patterns can be identified (HEWITT, 2010). Identification of these patterns is one of the fundamental challenges in two-phase flow studies (YAQUB *et al.*, 2020). Historically, the flow regime was identified by visual observation in transparent tubes, and the classification was based on subjective manual work (HEWITT, 1978). While these early approaches provided valuable insights, they were inherently limited by subjectivity, potentially leading to incorrect classification and, thus, inaccurate estimation of flow parameters, compromising both system safety and efficiency (AZZOPARDI; HILLS, 2003).

Machine learning techniques have been increasingly applied to address this challenge by automating the classification process (BAHRAMI *et al.*, 2024; BAO *et al.*, 2024; KHAN *et al.*, 2024). For instance, Li *et al.* (2016a) demonstrated the use of Fisher Discriminant Analysis (FDA) first to identify flow patterns, followed by the application of a void fraction calculation model based on Support Vector Machines (SVMs). However, incorrect pattern identification can result in suboptimal model selection and, subsequently, failed predictions. Moreover, as Yan *et al.* (2018) surveyed, while machine learning techniques have been widely applied to measure flow parameters, flow regime identification remains a specific challenge that requires more robust and generalizable solutions.

Several studies have addressed flow pattern classification using various techniques. Costigan and Whalley (1997) pioneered using the void fraction signal's probability density function (PDF) to relate flow regimes, although their work focused on qualitative analysis rather than direct classification. Rosa *et al.* (2010) applied artificial neural networks (ANNs) and expert systems for flow regime classification, concluding that while ANNs were effective, expert systems were better suited—albeit more complex and cumbersome. Similarly, Al-Naser *et al.* (2016) employed ANNs with preprocessing through Natural Logarithmic Normalization, enhancing the separability of data for better classification performance. However, these approaches also required difficult-to-obtain parameters like superficial liquid and gas velocities, limiting their applicability in real-time monitoring.

Further exploring void fraction analysis, Benito and Mureithi (2017) used fiber optic sensors to calculate statistical moments of the phase indicator function, which were then used as input for an SVM classifier. Mahvash and Ross (2008) utilized hidden Markov models (HMMs) for flow regime identification using similar data. Litak *et al.* (2017) applied recurrence plots and quantification analysis for classification. However, these studies relied on fragile or limited sensor technologies like fiber optics and high-speed cameras, making them impractical for widespread industrial use.

Despite some success, many existing approaches face a critical limitation: they depend on hand-crafted features, requiring detailed knowledge of the flow regimes. Such reliance on domain-specific knowledge makes it difficult to generalize these methods to different datasets or data collection systems. For example, while Shanthi and Pappa (2017) combined fuzzy logic with SVMs and Principal Component Analysis (PCA) for improved accuracy, these methods were tailored to specific data sets, and there was no guarantee they would perform equally well in different settings. Additionally, Ambrosio *et al.* (2022) applied Gaussian mixture models as feature extractors with SVM classifiers, achieving high accuracy, but the reliance on hand-crafted features limited the potential for broader application. To address this limitation, some recent studies have explored end-to-end machine learning methods that train the feature extractor and the classifier within the same architecture. Deep learning, in particular, offers a promising solution for two-phase flow classification by removing the need for manual feature extraction. For example, OuYang *et al.* (2022) developed a deep neural network (DNN) framework for multiphase classification, while Zhang *et al.* (2020a) applied convolutional neural networks (CNNs) for real-time flow regime identification. However, even these approaches have limitations, as their deep learning models were primarily adapted from vision-based applications rather than being specifically designed for time-series data, which is the format typical in two-phase flow studies.

This thesis presents two approaches for two-phase flow classification using void fraction data acquired by a wire-mesh sensor (WMS). The WMS is widely used in two-phase flow research due to its robustness, high spatial and temporal resolution, and cost-effectiveness (KIPPING *et al.*, 2016; SILVA *et al.*, 2007). Based on hand-crafted features, the first approach uses a Gaussian Mixture Model (GMM) to extract features from the void fraction time series, which are then classified using a Support Vector Machine (SVM). This approach leverages simpler, more interpretable models and avoids the hyperparameter complexities of deep learning techniques. Furthermore, it provides a robust solution for industrial environments, where the WMS can be used without requiring a transparent pipe for visual access. This first approach is defined as *MoG+SVM* in this work.

The second approach, an end-to-end approach, explores the application of state-of-the-art (SOTA) time-series classification methods for flow pattern recognition. These methods—Residual Networks (ResNet) (WANG *et al.*, 2017), LSTM Fully Convolutional Networks (LSTM-FCN) (KARIM *et al.*, 2018), and Time Series Transformers (TSTPlus) (ZERVEAS *et al.*, 2021)—are specifically designed for time-series classification tasks and eliminate the need for hand-crafted feature extraction. This second approach is defined as *SOTA-DL* in this work.

In addition to applying these advanced architectures, this thesis introduces a novel aspect of cross-dataset testing. By training models on one dataset and testing them on another, we evaluate the generalization capability of the models—a crucial but underexplored aspect in the related literature.

Through both single-dataset and cross-dataset tests, we compare the performance and generalization capacity of the two approaches. This research addresses the specific problem of flow regime identification. It contributes to the broader discussion of how machine learning methods can be applied to industrial flow metering, offering robust, generalizable solutions applicable across different datasets and acquisition systems.

1.1 OBJECTIVES

1.1.1 General Objective

The main objective of this thesis is to propose and evaluate two different approaches (*MoG+SVM* and *SOTA-DL*) for classifying two-phase flow patterns using void fraction data from wire-mesh sensors. The ultimate goal is to identify the most robust method for flow regime classification that can be applied in both academic research and industrial settings.

1.1.2 Specific Objectives

To achieve the general objective, the specific objectives of this research are as follows:

1. **Development of a Hand-Crafted Approach:** Design a methodology that extracts relevant features from the void fraction data using a Gaussian Mixture Model (GMM). These features will serve as inputs to a Support Vector Machine (SVM) classifier to perform flow regime identification. The output of this step will be a fully functional feature extraction pipeline integrated with the SVM classifier.

2. **Implementation of an End-to-End Approach:** Evaluate state-of-the-art time-series classification methods, including Residual Network (ResNet), LSTM Fully Convolutional Networks (LSTM-FCN), and Time Series Transformer (TSTPlus). These methods will be applied directly to the void fraction time-series data for flow regime identification. This step will result in trained models for each method.

3. **Evaluation and Dataset Organization**: Evaluate and organize the two datasets used in this work to compare the methods and experiments. This organization involves visual analysis of the data, exclusion of problematic points or points with measurement failures, review and annotation of flow patterns, preprocessing, and other structuring necessary for subsequent objectives.

4. **Single-Dataset Performance Evaluation:** Train and test both the Hand-Crafted and End-to-End approaches on the same dataset to assess their performance under consistent data acquisition conditions. The result of this objective will be a performance analysis report that includes accuracy, precision, recall, and F1-score metrics.

5. **Cross-Dataset Generalization Tests:** Assess the generalization capabilities of both approaches by training the models on one dataset and testing them on a different dataset collected under different conditions. This step aims to evaluate the robustness of each method in handling different operating environments, with a comparison report as the outcome.

6. **Comparative Analysis of Approaches:** Conduct a detailed comparison of the Hand-Crafted and End-to-End approaches based on the single-dataset and cross-dataset tests. This

will provide insights into which method is the most robust for flow regime classification, along with the strengths and limitations of each approach.

1.2 PUBLICATIONS

During this doctoral project, the following publications directly related to the thesis were carried out:

- AMBROSIO, Jefferson dos Santos; LAZZARETTI, André Eugenio; PIPA, Daniel Rodrigues; SILVA, Marco Jose da. Two-phase flow pattern classification based on void fraction time series and machine learning. **Flow Measurement and Instrumentation**, Elsevier, v. 83, p. 102084, 2022.
- AMBROSIO, Jefferson dos Santos; SILVA, Marco Jose; LAZZARETTI, André Eugenio. A deep learning-based approach for two-phase flow pattern classification using void fraction time series analysis. **IEEE Access**, IEEE, v. 13, p. 11778, 2025.

On the other hand, Table 1 provides insight into the repositories associated with the thesis, offering access to additional resources and materials related to the conducted research. Each repository is accompanied by a related section in the document, facilitating the location of specific information.

Table 1 – Repositories related to this document.

Repository	Related Section
https://github.com/ambrosioj/GMM_Nested_CV	Section 3.3
https://github.com/ambrosioj/two-phase-time-series-deep-learning	Section 3.4

1.3 STRUCTURE OF THIS DOCUMENT

Chapter 1 sets the stage by introducing the research problem and objectives. It provides an overview of the study’s significance and outlines the structure of the thesis. Chapter 2 presents multiphase concepts and essential aspects of the machine learning models. Also, a review of flow pattern classification is presented, detailing classical, machine, and deep learning-based methods. Chapter 3 details the two datasets used in this work, the *MoG+SVM* and the *SOTA-DL*. Additionally, preprocessing stages and experiment procedures are discussed to provide a better understanding of the implementations. Chapter 4 details the results of each experiment, the generalization analysis, and the advantages and limitations of each procedure, indicating a possible most appropriate solution for the pattern classification problem. Chapter 5 presents an analysis of the research findings and their implications. This section acts as the apex of the thesis, providing valuable conclusions derived from the investigation. The discussion elaborates on attaining each objective within the thesis and highlights compelling avenues for future research to broaden and enhance the study's scope.

2 MULTIPHASE FLOW AND PATTERN CLASSIFICATION

This chapter will present some fundamental concepts related to the study of multiphase flow. A literature review will also be provided to contextualize the discussions presented in this document.

2.1 MULTIPHASE FLOW

Multiphase flow can be defined as the simultaneous flow between different fluids in various physical states. For example, there is liquid-gas, liquid-liquid, and liquid-liquid-gas flow, among others. These flows can also involve the same substance in different physical states, as in the liquid-gas flow with water and water vapor, which appears in several industrial applications. Another example is liquid-liquid-gas-solid flow (sand-oil-water-natural gas), often encountered in oil extraction (GIOIA *et al.*, 2009). In the course of this work, the focus will be on liquid-gas two-phase flow in vertical pipes.

2.1.1 Vertical Two-Phase Flow

One of the most critical issues in studying two-phase flows is the flow patterns. These patterns are the structures that form at the interfaces that arise between fluids. These interfaces depend on several factors, such as the fraction and velocity of each phase, surface tension, operating pressure and temperature, pipe diameter, and inclination, as well as density, viscosity, and other properties. As can be seen, the classification of patterns depends on many variables, making it a very complex and sometimes subjective issue. Figure 1 illustrates one way this differentiation is made, based on the work of Taitel *et al.* (1980). A brief description of the patterns is provided below.

Bubbly Flow: This pattern can be observed as tiny bubbles randomly distributed in the liquid throughout the pipe, where the liquid phase is the continuous phase and the gas is the dispersed phase.

Slug Flow: With the increase in gas flow rate, some tiny bubbles tend to coalesce, resulting in elongated bubbles known as Taylor bubbles, which are followed by a liquid piston that may contain dispersed gas bubbles.

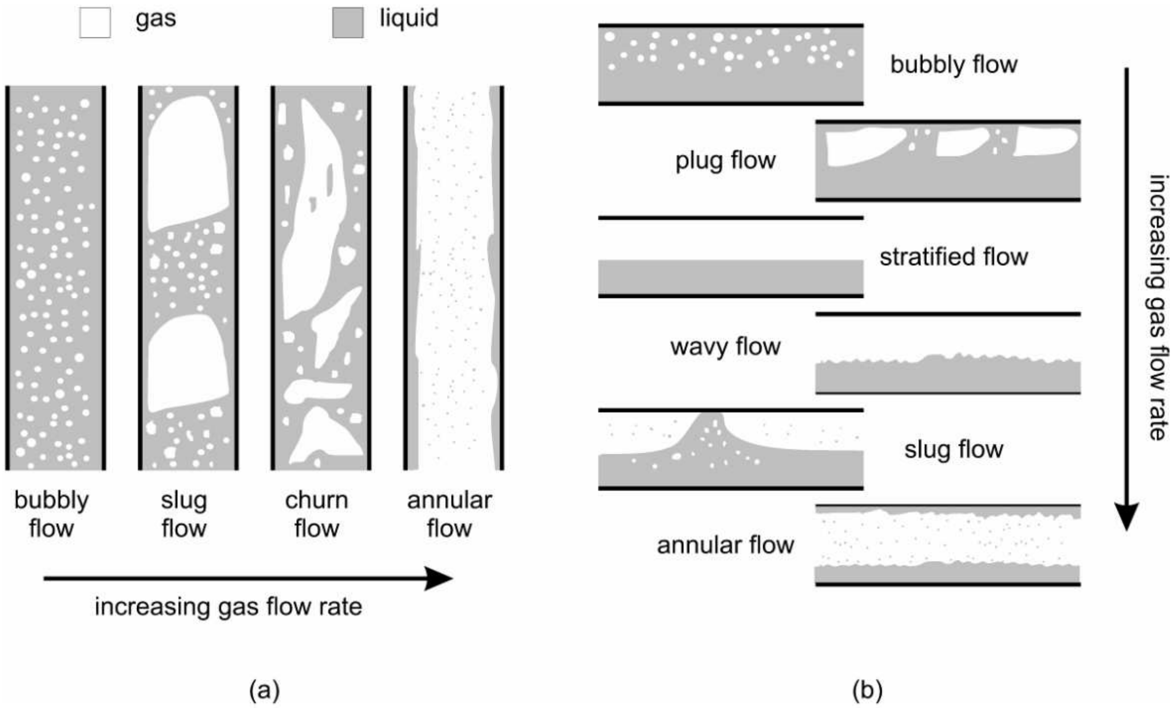


Figure 1 – Flow regimes in (a) vertical gas-liquid up-flows and (b) horizontal gas-liquid flows. Adapted from Silva (2008).

Churn Flow: With a further increase in gas flow rate, the Taylor bubbles collapse, giving rise to a more unstable and chaotic pattern.

Annular Flow: The annular pattern occurs when the gas flow rate is so high that it generates a central gas column with tiny dispersed liquid droplets.

2.1.2 Flow Parameters and Pattern Classification

Since, in most applications, it is impossible to visualize the flow pattern occurring in the pipe directly, approaches are needed where it is possible to relate the pattern to some measurable characteristic. One alternative is to use a void fraction, which indicates the amount of gas in the flow. First, it is necessary to define some concepts.

2.1.2.1 Phase Indicator Function and Void Fraction

The phase indicator function P_k, as described in the work of Bertola (2014), is a binary function that describes the presence or absence of phase k at a given point $\mathbf{x}$ and time t. Thus,

$$P_k(\mathbf{x},t) = \begin{cases} 1, if\ \mathbf{x} \in phase \quad k \\ 0, if\ \mathbf{x} \notin phase \quad k, \end{cases} \tag{1}$$

where $k = G, L$ for the gas or liquid phase, respectively.

From this definition, the average void fraction in the cross-section α is obtained from Equation 2:

$$\alpha(t) = \frac{1}{A}\int_A P_G(\mathbf{x},t)da = \frac{A_G}{A}, \tag{2}$$

where $P_G(\mathbf{x},t)$ is the gas phase indicator function, A is the total cross-sectional area of the pipe, and A_G is the area occupied by the gas phase. Sampling this average void fraction over a given time interval generates the time series of the void fraction, as shown in Figures 2 (a), 2 (c), and 2 (e).

2.1.2.2 Superficial Velocity

Next, it is necessary to define an essential parameter called superficial velocity. It is calculated considering that the phase flows through the pipe occupying its entire cross-sectional area, as shown in Equation 3:

$$j_k = \frac{Q_k}{A}, \tag{3}$$

where Q_k is the volumetric flow rate of phase k and A is the cross-sectional area of the pipe.

2.1.2.3 Pattern Classification

The time series is directly linked to the distribution and development of interfaces in two-phase flow. In works such as Jones and Zuber (1975) and Costigan and Whalley (1997), the authors propose a methodology in which the time series and its respective histogram are used to classify the flow pattern.

Using data from Wire-Mesh sensors, the average void fraction in the pipe cross-section can be calculated, generating a time series of the void fraction. Figures 2 (a) and 2 (b) illustrate the time series and histogram of a bubbly flow pattern. Due to the occurrence of tiny bubbles, this pattern shows low void fraction values in the cross-section, which can be observed in both the time series and the peak for low values in the histogram.

Figures 2 (c) and 2 (d) provide information about the slug flow pattern. In the time series, Taylor bubbles can be identified in the higher void fraction peaks, and liquid slugs, where the void fraction is lower, can also be observed. This is reflected in the bimodal behavior of the histogram, owing to the flow characteristics.

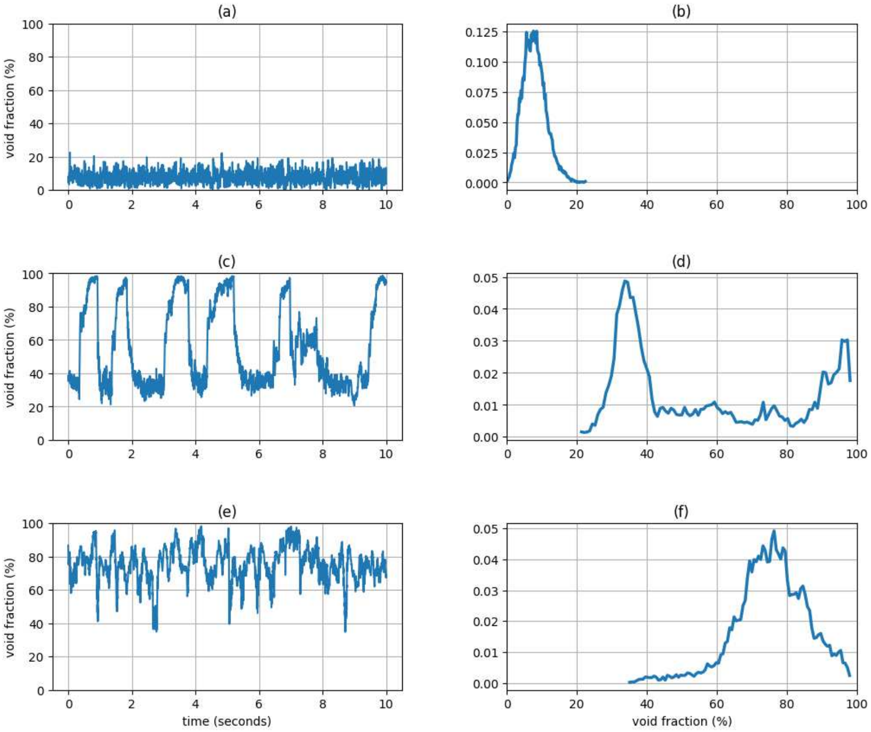


Figure 2 – (a) Time series of an operation point in bubbly flow; (b) Histogram of the time series in bubbly flow; (c) Time series of an operation point in slug flow; (d) Histogram of the time series in slug flow; (e) Time series of an operation point in churn flow; (f) Histogram of the time series in churn flow.

For churn flow, Figures 2 (e) and 2 (f) indicate the presence of a pattern where the void fraction is higher, as can be seen in Figure 1.

A widely used methodology in classification is relating the flow pattern to the superficial velocities of each phase, as shown in Figure 3, where j_g and j_l are the gas and liquid superficial velocities, respectively. In Taitel *et al.* (1980), a model is developed that defines transitions between patterns, considering superficial velocities, pipe diameter, and fluid properties.

2.2 WIRE-MESH SENSOR

The Wire-Mesh Sensor (WMS) is an advanced imaging device designed for multiphase flow measurements, combining the principles of intrusive electrode arrays with tomographic techniques. Initially proposed by Prasser *et al.* (1998), the sensor measures the local electri-

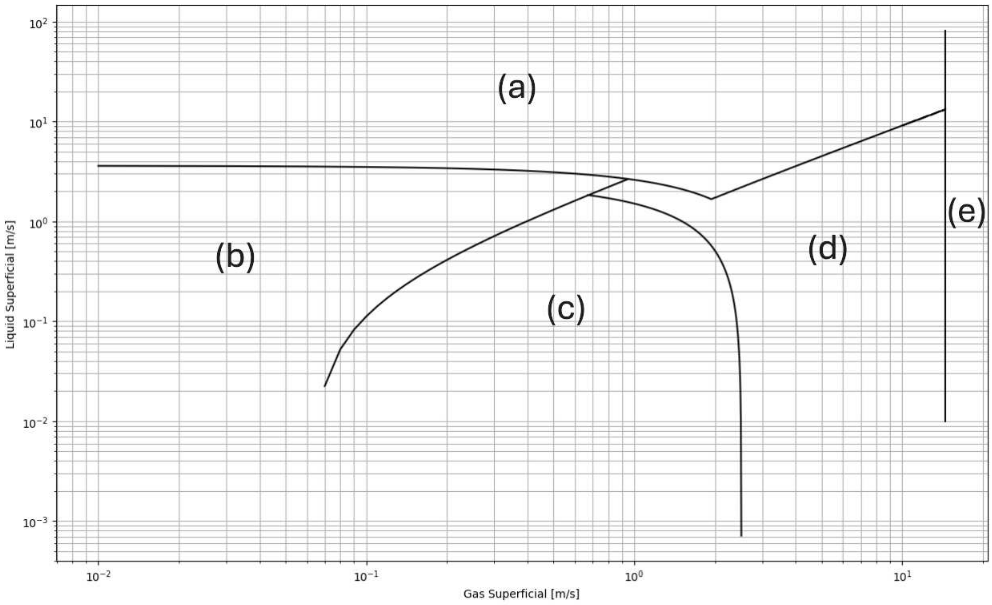


Figure 3 – Flow Pattern Map for Vertical Liquid-Gas Two-Phase Flow: This map, based on the model by Taitel *et al.* (1980), depicts the relationship between flow patterns and the superficial velocities of each phase. The flow patterns are categorized as follows: (a) Dispersed Bubbly, (b) Bubbly, (c) Slug, (d) Churn, and (e) Annular.

cal conductivity of a two-phase mixture, allowing for high-resolution visualization of phase distribution.

The WMS consists of two parallel electrode grids, where one set functions as transmitters and the other as receivers. When an electrical signal is applied to the transmitting electrodes, the receiving electrodes measure the resulting conductivity changes, enabling the reconstruction of spatial phase fraction distributions. This design provides high temporal and spatial resolution, making it a valuable tool for analyzing complex flow patterns.

Compared to X-ray and gamma-ray tomography, which offer excellent spatial resolution but at a significantly higher cost, the WMS presents a cost-effective alternative while still achieving remarkable measurement accuracy (SILVA *et al.*, 2007). Additionally, due to its rapid acquisition rate, the sensor is particularly suited for studying transient flow dynamics in industrial applications.

A simplified schematic of a WMS installed in a pipeline is shown in Figure 4(a), while Figure 4(b) presents an image of a sensor developed at the Helmholtz-Zentrum Dresden-Rossendorf (HZDR).

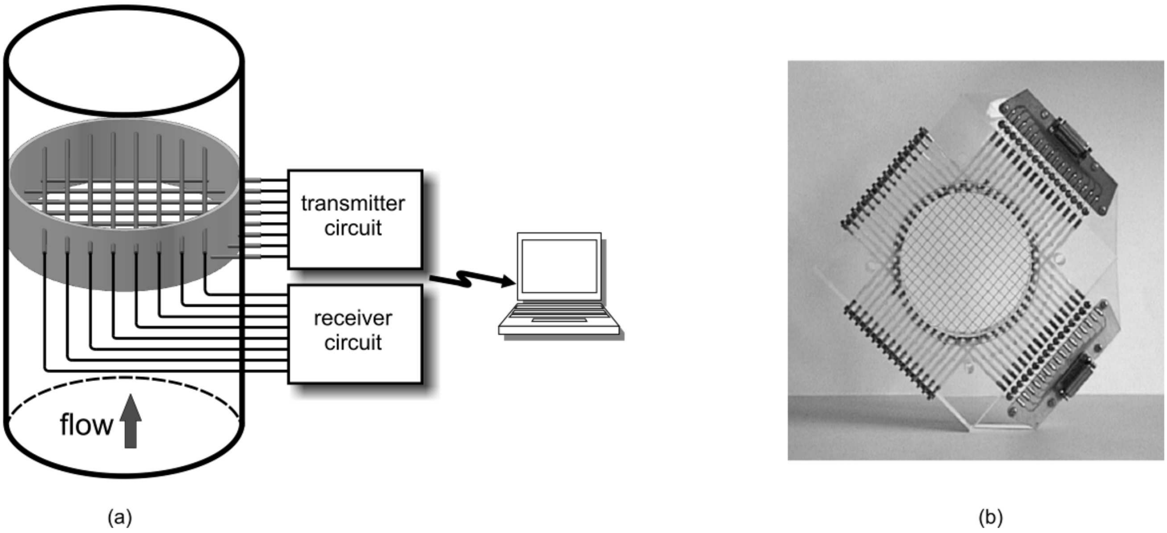


Figure 4 – (a) schematic representation of a Wire-Mesh Sensor; (b) photography of a typical sensor developed at HZDR. Adapted from Silva (2008).

2.3 RELATED WORKS

This section reviews related works on two-phase flow pattern identification and void fraction measurement, categorizing them into classical approaches, machine learning methods, and deep learning techniques. The discussion focuses on key findings and the instrumentation used, including experimental setups, sensor technologies, and data preprocessing techniques. By examining these aspects, this review contextualizes existing methodologies, highlighting their contributions, limitations, and relevance to the present study.

2.3.1 Classical Approaches

2.3.1.1 Theoretical and Mechanistic Modeling

Taitel and Dukler (1976) presents a theoretical model for predicting flow regime transitions in horizontal gas-liquid two-phase flow. The authors develop analytical criteria for transitions between five basic flow regimes: smooth stratified, wavy stratified, intermittent (slug and plug), annular with dispersed liquid, and dispersed bubble. The key achievement is a fully predictive model based on physical concepts, without relying on empirical flow regime data. The authors derive dimensionless equations and present a generalized flow regime map. Good agreement is reported between theoretical predictions and latest available experimental flow regime maps. The model successfully predicts shifts in transition boundaries for larger pipe sizes, high-pressure gases, and slight pipe inclinations. A strength of the work is its basis in physical mechanisms and first principles. However, the paper lacks an extensive discussion of instrumentation and experimental validation, relying mainly on comparisons with existing maps. There is little mention of generalization tests across a wide range of conditions. The authors acknowledge that some transitions, such as smooth to wavy stratified flow, use approximate models that could be refined with additional data. Despite these limitations, the paper presents a significant theoretical advance in predicting gas-liquid flow regimes, providing a foundation for predicting transitions across various conditions and addressing limitations of purely empirical maps.

Jones and Zuber (1975) presents a significant advancement in the study of two-phase flow patterns. The authors developed a fast-response X-ray void measurement system to obtain statistical measurements in air-water flow within a rectangular channel. Their key achievement was demonstrating that the probability density function (PDF) of void fraction fluctuations could serve as an objective and quantitative flow pattern discriminator for bubbly, slug, and annular flow regimes. The study covered a wide range of mixture velocities from 0.0 to $37m/s$. The authors provided detailed insights into flow characteristics, including film thickness calculations, slug length and residence time ratios, and spectral density measurements. The instrumentation used was highly sophisticated for its time, featuring a modified X-ray diffraction unit with fast response time and high accuracy. The system's response time to a 100% step change in void fraction was less than $1m/s$. However, the study has some limitations in terms of generalization. The experiments were conducted in a specific rectangular channel geometry, and the authors acknowledge that developing flow effects were observed in bubbly flow cases. Additionally, the bubbly flow runs were not contiguous with the slug and annular flow runs in flow parameters due to experimental constraints. Despite these limitations, the paper provides a robust methodology for objective flow pattern determination and offers valuable insights into the structure of two-phase flows, particularly the concept of slug flow as a periodic time combination of bubbly and annular flows.

2.3.1.2 Void Fraction Measurement Techniques

Li *et al.* (2016b) developed a new void fraction measurement method for gas-liquid two-phase flow in small channels using a laser diode, photodiode array sensor, and machine learning techniques. They first identified the flow pattern using Fisher Discriminant Analysis, then selected a relevant Support Vector Machine model to measure void fraction. Experiments were conducted in four different small channels with diameters ranging from 1.08 to 4.22 mm. The method achieved a maximum absolute error of less than 7% compared to reference void fractions. Key strengths include the low cost and simple structure of the system compared to conventional laser measurement systems. The use of machine learning allowed them to overcome the influence of flow patterns on void fraction measurement. However, the authors note that the number of sensing elements used was not optimal, and further research is needed to determine the ideal configuration. While they tested the system in multiple channel sizes, there is no explicit discussion of generalization tests beyond the experimental setup described.

Silva *et al.* (2007) introduced a new wire-mesh sensor based on capacitance measurements for fast measurement of phase fraction distributions in multiphase flows. Their prototype sensor comprised two planes of 16 wires each, achieving a time resolution of 625 frames per second. The sensor showed good stability and accuracy in capacitance measurements, with uncertainties in the femtofarad range. They demonstrated the sensor's ability to distinguish between substances with slight differences in relative permittivity, such as air and silicone oil. Instrumental noise analysis, substance discrimination tests, and time response assessments evaluated the system's performance. The sensor's ability to measure in non-conducting or slightly conducting fluids is a key strength. However, the authors note that the data acquisition card limits the current frame rate and could be increased to above 10,000 fps with improved electronics. While they tested the sensor with various substances and in a silicone oil/water two-phase flow, there is no explicit discussion of generalization tests beyond these experiments.

Costigan and Whalley (1997) developed and tested a conductivity probe for measuring void fraction in vertical air-water flows. They conducted experiments in a $32mm$ diameter vertical tube, covering water superficial velocities from 0 to $1.0m/s$ and air superficial velocities from 0.05 to $37m/s$. The probe showed good accuracy, with measurements within 0.1 of reference values over the full range of void fractions. Examining void fraction traces and probability distribution functions, they identified six flow regimes: discrete bubble, spherical cap bubble, stable slug, unstable slug, churn, and annular flow. They provided statistical data on slug and bubble lengths through cross-correlation of signals from two probes. A key achievement was the identification of two different flooding mechanisms associated with transitions to unstable slug flow and churn flow . The instrumentation appears robust, with constant air flows ensured by critical flow nozzles. However, the authors note that their conductivity-based method is primarily responsive to conductivity rather than capacitance, which could limit its applicability to non-conductive fluids. While they tested the system over a wide range of flow conditions, there is no explicit discussion of generalization tests beyond their experimental setup.

2.3.1.3 Statistical and Signal Analysis Methods

Litak *et al.* (2017) investigated two-phase air-water flow patterns in a mini channel using recurrence plots and recurrence quantification analysis. The authors observed the evolution of air bubbles and slugs by changing air and water flow rates. They used a digital camera for qualitative visualization and a laser-phototransistor sensor to record light transmission time series for detailed analysis. The study achieved the identification of particular flow patterns using recurrence quantifiers obtained from the laser time series. The instrumentation included a high-speed camera and laser sensor, providing visual and quantitative data. However, the paper does not mention generalization tests for other flow conditions or geometries.

Figueiredo *et al.* (2020) proposed a method to classify flow patterns in two-phase vertical water-air flows using a single ultrasonic transducer. The authors developed a classifier based on the coefficient of variation of the energy of ultrasonic pulses reflected by the dispersed phase. The technique achieved high recognition rates, with 100% accuracy for dispersed bubbles and 98.9% for slug and churn flows. The instrumentation included a single ultrasonic transducer, making it a relatively simple and non-intrusive setup. However, the study was limited to specific pipe diameters and fluid properties, and the authors acknowledge that further tests are needed to confirm the applicability to different operational conditions.

Vieira *et al.* (2021) presented a method for determining parameters in two-phase gas-liquid flow, precisely the velocity of Taylor Bubbles, using differential pressure measurements. The authors analyzed a normalized signal's time and frequency signature and its coupling with void fraction waves. They proposed a phase difference estimator based on the cross-spectra between differential pressure signals. The study demonstrated a coupling between differential pressure signals and void fraction kinematic waves in a specific frequency region. The instrumentation included differential pressure sensors, providing non-intrusive measurements. However, the study focused only on intermittent slug flow, and the authors note that further work is needed to extend the approach to different flow patterns.

Poelma (2020) This review paper discussed various measurement techniques for characterizing dispersed multiphase flows that are not accessible by conventional optical methods. The author highlighted several techniques, including ultrasound velocimetry, magnetic resonance velocimetry, and X-ray imaging. The paper emphasized the potential of these techniques for providing time-averaged and time-resolved velocity fields in opaque flows. However, the author noted that the impact of these techniques will depend on the development of acquisition and measurement protocols specifically for fluid mechanics, rather than clinical imaging. The review stressed the need for systematic development and careful validation experiments. A key limitation highlighted was the lack of standardized 'benchmark' flows for multiphase systems, which are necessary for proper validation of these techniques.

2.3.2 Machine Learning Approaches

Shanthi and Pappa (2017) developed an artificial intelligence system to classify two-phase flow patterns using image processing and machine learning techniques. They extracted textural and shape features from images of six different flow patterns and used fuzzy logic, support vector machine (SVM), and SVM with principal component analysis (PCA) for classification. Their best results were achieved using SVM with PCA, obtaining classification accuracy of over 98%. However, the study was limited to a single experimental setup with fixed pipe dimensions, and no generalization tests were reported for different flow conditions or geometries.

Mahvash and Ross (2008) applied continuous hidden Markov models (CHMMs) to identify two-phase flow patterns using local void fraction signals from an optical fiber probe. They compared different feature extraction methods, including Walsh-Hadamard transform, autoregressive modeling, and a novel passage length-based approach. The passage length method with maximum total likelihood achieved the best results, showing good agreement with theoretical flow maps and experimental photographs. Their system achieved over 96% success rate in flow pattern identification. However, the study was limited to a single vertical pipe geometry, and the generalization capability to different pipe orientations or diameters was not explored.

Yan *et al.* (2018) provided a comprehensive review of soft computing techniques applied to multiphase flow measurement. They discussed various approaches including artificial neural networks (ANNs), support vector machines (SVMs), genetic algorithms (GAs), and adaptive neuro-fuzzy inference systems (ANFIS). The review highlighted that multi-sensor systems combined with soft computing techniques can achieve higher accuracies in estimating phase flowrates and fractions. For example, some studies reported errors less than ±2% for liquid mass flowrate measurements using Coriolis flowmeters with ANNs. However, the review also noted that most studies were limited to specific experimental setups, and there is a need for more generalization tests and in-field training to adapt models to real-world conditions.

Al-Naser *et al.* (2016) developed an artificial neural network (ANN) model for flow pattern identification using dimensionless inputs and a preprocessing stage with natural logarithmic normalization. They used three dimensionless parameters: Liquid Reynolds Number, Gas Reynolds Number, and Pressure Drop Multiplier. The model achieved over 97% accuracy in classifying flow patterns for a wide range of flow conditions. The authors validated their approach using both simulated and experimental data. The use of dimensionless inputs allowed the model to be applied to pipes with various diameters and fluids with different properties. However, while the study showed improved performance at transition regions, some misclassifications still occurred in these areas, indicating room for further improvement in handling complex flow transitions.

Liu *et al.* (2023) investigated two-phase flow pattern identification in horizontal gas-liquid swirling pipe flow using machine learning methods. They observed five swirling flow patterns through visualization experiments and extracted void fraction data using image processing. Statistical analysis of void fraction probability density functions (PDFs) was performed, and four parameters describing PDF characteristics were proposed as inputs for machine learning algorithms. Five machine learning algorithms were tested, with RUSBoost tree performing best at 97.4% accuracy. The study used a robust experimental setup with high-speed imaging and image processing, allowing for detailed flow visualization and quantitative data extraction. However, generalization tests to other flow conditions or geometries were not reported.

Sestito *et al.* (2023) compared artificial neural networks (ANNs) and support vector machines (SVMs) for classifying flow patterns in vertical tube bundles based on vibration data. They used a minimally invasive instrumented tube to collect frequency response data. Multiple ANN and SVM architectures were tested, with the best performers being ANN with log-sigmoid output (90% accuracy) for bubble flow, SVM with RBF kernel for churn flow, and SVM with linear kernel for intermittent flow. The study used a well-designed experimental setup mimicking industrial heat exchangers. However, limited data for churn flow patterns affected the reliability of those results. Generalization tests to other geometries or flow conditions were not reported.

Benito and Mureithi (2017) developed a support vector machine (SVM) classifier for two-phase flow pattern identification in tube arrays. They used optical probe measurements to obtain void fraction data and extracted statistical moments and probability density functions

(PDFs) as flow pattern indicators. The SVM classifier was trained on these features to identify flow patterns. The study used a simplified test section with good visual access. Performance metrics were not explicitly reported, but the authors stated the SVM provided "quantitative and accurate classifiers". The main limitation was the use of a simplified geometry that may not fully represent industrial tube arrays.

Rosa *et al.* (2010) compared various artificial neural networks (ANNs) and expert systems for flow pattern classification in vertical gas-liquid flows. They used a resistivity probe to measure void fraction and extracted statistical moments and PDFs as inputs. Multiple ANN architectures were tested, with multi-layer perceptrons (MLPs) with multiple outputs performing best (96-100% accuracy). Expert systems showed comparable performance to single-output ANNs (70-91% accuracy). The study used a well-designed experimental setup with a large number of flow conditions. However, the imbalanced dataset with more samples for certain flow patterns may have affected the results. Generalization tests to other pipe sizes or orientations were not reported.

Our previous research in Ambrosio (2019), presented an initial method for flow pattern classification, employing a Gaussian mixture model and a support vector machine classifier, which yields high accuracy and low classification errors compared to conventional techniques. The development used void fraction data from a wire-mesh sensor, leveraging data from the TOPFLOW test facility at HZDR, Germany. However, the study acknowledges limitations of the data source as a limitation for generalization analysis and a more detailed analysis of the features and the obtained results, which has motivated the proposal of the present research.

2.3.3 Deep Learning Techniques

OuYang *et al.* (2022) developed a deep neural network framework for two-phase flow pattern identification using multivariate time series data from a four-channel conductance sensor. Their model combined BiLSTM and CNN with attention mechanisms and residual connections. They achieved a high recognition accuracy of 99.32% for five flow patterns. The model showed good noise immunity, maintaining over 93% accuracy with added Gaussian noise. However, the study was limited to vertical gas-water flows and did not test generalization to other flow types or pipe orientations.

Zhu *et al.* (2022) proposed a machine learning approach for flow regime detection using a gamma-ray-based multiphase flowmeter. They experimented with CNN, LSTM, and

CNN-LSTM models, achieving the highest accuracy of 99.6$\%$ using LSTM with regularization and data preprocessing. Their approach could predict flow regimes at a horizontal pipe entrance using measurements from a vertical Venturi throat section. While innovative, the study was limited to a single experimental setup and did not explore generalization to other flowmeter types or flow conditions.

Zhang *et al.* (2020a) developed a flow regime identification method using liquid phase velocity information from ultrasound Doppler velocimetry and various machine learning algorithms. They achieved 93.1% accuracy for real-time identification using SVM and proposed novel parameters for identifying plug and decaying slug flows. For transient flow regime identification, they used LSTM and CNN, achieving 94$\%$ accuracy with CNN. The study used high-quality instrumentation, including synchronized high-speed camera visualization. However, generalization was limited, as the experiments focused on horizontal air-water flows in a single pipe diameter.

Hobold and da Silva (2019) developed neural network models to predict heat flux in nucleate boiling using only visual information from boiling process imaging. The models were able to encode bubble morphology and correlate it with heat flux, achieving errors as low as 7$\%$ compared to experimental measurements. This represents a significant improvement over existing prediction methods for boiling heat transfer. The authors demonstrated that these models could be implemented on inexpensive computers like Raspberry Pi to infer heat flux in real-time from visualization. A key achievement was that neural networks can learn to quantify heat transfer using only visual data without prior knowledge of governing equations. The study used high-quality experimental data from shadowgraph imaging but did not extensively test generalization to different boiling conditions or geometries. Overall, this work demonstrated the potential of machine learning to provide non-intrusive measurement of complex multiphase flows.

Nunno *et al.* (2020) developed machine learning models to predict the equivalent diameter and aspect ratio of air bubbles near a plunging jet. The authors used a shadowgraph technique to obtain experimental data and trained various algorithms including Random Forest, K-Star, and Support Vector Regression. For predicting equivalent diameter, the best models achieved R^2 values > 0.97 and mean absolute errors $< 0.1mm$. Aspect ratio predictions were less accurate, with R^2 values around 0.5. A key contribution was using unsupervised clustering to objectively define flow regimes before classification. The experimental setup provided high-resolution void fraction measurements, but was limited to a single geometry and adiabatic conditions. The authors did not extensively test generalization to other flow conditions or geometries. Overall, this work demonstrated the potential of machine learning to analyze complex multiphase flows, but highlighted challenges in predicting certain bubble properties.

Breitenmoser *et al.* (2023) developed a novel two-step machine learning approach for automatic flow regime recognition in helical two-phase flows. The authors used unsupervised clustering to objectively identify flow regimes, followed by supervised classification. The classifiers achieved excellent accuracy >98%, with K-Nearest Neighbors reaching 100% accuracy. This represents a significant improvement over previous studies which had errors $> 14\%$ for some regimes. The work used high-resolution X-ray radiography data, providing detailed void fraction measurements. However, experiments were limited to a single helical coil geometry and adiabatic conditions. The authors compared results to existing flow regime models, revealing significant discrepancies and highlighting limitations in current predictive capabilities for helical flows. While generalization was not extensively tested, this work demonstrated the potential of machine learning to objectively analyze flow regimes and improve understanding of complex multiphase flow dynamics.

2.3.4 Limitations of recent works and contributions of this thesis

The studies detailed so far have illustrated the breadth of methods being explored for two-phase flow analysis, from classical mechanistic models and statistical methods to advanced machine learning and deep learning frameworks. In general, the combination of traditional physical modeling and emerging AI-based approaches shows promise in addressing the limitations of each individual technique.

However, one can formulate different limitations for different approaches. In machine learning methods, depending on the method chosen for feature extraction, the result may be negatively impacted (LIU *et al.*, 2023; SESTITO *et al.*, 2023), especially for flow patterns in transitions between classes (BENITO; MUREITHI, 2017). In other cases, the interpretability of the model, especially in terms of features (SHANTHI; PAPPA, 2017; YAN *et al.*, 2018), may impact a more detailed analysis of error cases or even make it difficult to use the proposed approaches in practical cases, given the lack of relationship between the calculated features and physical aspects of the flow.

In this sense, our first proposal, *MoG+SVM*, which extends the work initially done in Ambrosio (2019), in addition to promoting quantitative results comparable to those in the literature, also promotes the use of more interpretable features with a physical relationship to the phenomenon. For example, the classical approaches of Jones and Zuber (1975) and Costigan and Whalley (1997), in which the time series and its respective histogram are used to classify the flow pattern, have a direct relationship with the features calculated by the MoG model, which, when combined with an SVM, result in classification performances above 90%, training and testing with the same dataset.

Regarding our second approach, *SOTA-DL*, we observed that not all works compare recent state-of-the-art architectures for the problem in question. Furthermore, some works use input data that is different from the present proposal. Additionally, and more relevant in the context proposed here, the models were not evaluated in generalization situations, with cross-dataset experiments, which makes it difficult to evaluate the proposals in situations closer to real operating situations (training and testing in different data collection situations). Hence, the main contributions of the *SOTA-DL* approach w.r.t. the literature can be summarized as follows:

- Propose using end-to-end SOTA time-series classification methods (ResNet, LSTM-FCN, and TSTPlus) for two-phase flow patterns, promoting robustness provided by those approaches in our generalization analysis.

- Present the generalization analysis of the models with cross-dataset experiments, training the model with one dataset and testing it with another dataset collected in another system.

- A detailed comparison of both approaches (*MoG+SVM* and *SOTA-DL*), demonstrating limitations and situations that each approach is more promising, particularly in generalization analysis.

Overall, this thesis builds upon these advancements by integrating deep learning models with feature extraction techniques, specifically addressing the need for improved generalization across different flow conditions. By enhancing flow regime identification accuracy and exploring novel void fraction measurement methodologies using state-of-the-art time-series analysis, this work aims to contribute significantly to the field of two-phase flow analysis. Hence, we believe that the second approach, *SOTA-DL*, may be the most relevant contribution of this work.

3 PROPOSED METHODS

This chapter outlines methods for classifying two-phase flow patterns using sensor data from the HZDR and TUD datasets. It begins by describing the datasets in detail, focusing on the void fraction time series and associated flow patterns. For a classical machine learning approach (*MoG+SVM*), with handcrafted features, the Gaussian Mixture Model (GMM), along with Maximum Likelihood Estimation and the Expectation-Maximization (EM) algorithm, are introduced for modeling the data. Following this, Multiclass Support Vector Machines are used for classification, with Nested Cross-Validation employed to evaluate model performance.

The second part of the chapter discusses the time series approach (*SOTA-DL*), including the preprocessing steps required for these data. State-of-the-art models for time series classification are presented, followed by details on the training process and cross-dataset experiments to assess model generalizability. The chapter concludes by explaining the performance metrics used for evaluation and providing information on the implementation of the entire pipeline.

3.1 OVERVIEW OF THE PROPOSED METHODS

To better understand the process as a whole, Figure 5 presents the general overview of the proposed method. Initially, the data from the HZDR and TUD datasets were organized, considering they were composed of time series from wire-mesh sensors. Then, each time series from each dataset undergoes a preprocessing step involving standardization of sampling frequency, resampling when necessary, and windowing with normalization. In this step, the division into folds is also performed to evaluate each dataset and perform cross-validation. Subsequently, the *MoG+SVM* model is trained for each dataset, as well as the *SOTA-DL* classification methods. Finally, the final classification is evaluated (a three-class classification problem, i.e., bubbly, slug, and churn), generating the comparison metrics. Each step of this method will be detailed below.

3.2 HZDR DATASET

The first dataset we used is wire-mesh air-water vertical flow data generated in the TOPFLOW test facility (Transient TwO-Phase Flow) at HZDR, Germany. In short, A vertical pipe of 52.3 mm internal diameter (DN50 pipe) is fed with air and water to form a two-phase flow,

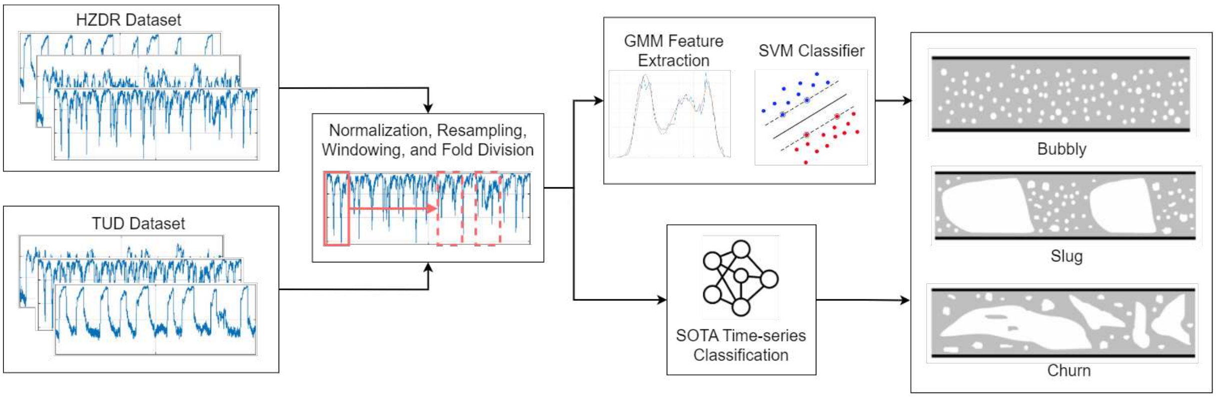

HZDR Dataset
TUD Dataset
Normalization, Resampling, Windowing, and Fold Division
GMM Feature Extraction
SVM Classifier
SOTA Time-series Classification
Bubbly
Slug
Churn

Figure 5 – General overview of the proposed method.

which is monitored by a WMS at the end of the test section. Figure 6 shows a simplified scheme of the vertical test section, in which the accessory equipment, such as the pump, tanks, and the gas-liquid separator, was omitted. More details on the plant, methodology, and experiments can be found in the technical report (SCHÜTZ *et al.*, 2007). The facility was conceived with the initial aim of studying nuclear reactor safety. Several experiments on vertical two-phase flows were carried out to obtain reliable data for developing and validating Computational Fluid Dynamics (CFD) codes. Due to the quality of the measurements and the amount of information available in the project mentioned above, several studies were developed using the high-quality data, for instance (BANOWSKI *et al.*, 2017) and (LUCAS *et al.*, 2010a; LUCAS *et al.*, 2010b).

The experiments carried out by the HZDR cover a wide combination of experimental

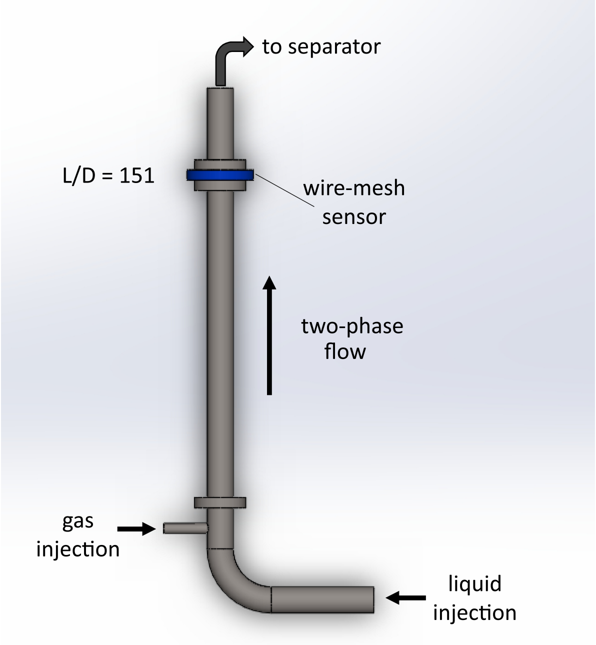


Figure 6 – Simplified schema of the vertical 52.3 mm i.d. (DN50) test section of the HZDR's TOPFLOW. WMS is placed at 7.9 m from gas injection point, i.e. length to diameter ratio $L/D = 151$

Table 2 – Main Parameters

Superficial Velocity Ranges	Flow Pattern Experimental Points	
j_g = 0.0025 to 12.14 m/s	Bubbly	35
	Slug	19
j_l = 0.0405 to 4.047 m/s	Churn	13
	Total	67

parameters, conditions, and applications. The complete data set includes void fraction data for WMS placed in different measurement positions. Among all of them, we selected the experimental points with the most significant distance from the mixing point (i.e., length to diameter ratio $L/D = 151$) so that the flow is considered fully developed. Furthermore, from the available 70 operating points, we also selected a subset containing 67, thus excluding 3 points for a flow type called annular flow (see more details on the flow pattern types below). As this work proposes to use a supervised learning technique, only 3 points for training and testing the annular pattern are unfeasible. A summary of the operational points used in this work is shown in Table 2. Figure 7 shows the data used according to the classification map from mechanistic flow modeling defined in (TAITEL *et al.*, 1980). Each operating point is determined by a pair of superficial liquid velocity j_l and gas superficial velocity j_g, obtained by reference sensors. Measurements were performed using 16×16 conductive wire-mesh sensors with an internal diameter of 52.3 mm. The sampling frequency used is 2.5 kHz with superficial gas velocity varying between 0.0025 and 12.14 m/s, the liquid superficial velocity varying between 0.0405 and 4.047 m/s, and a sampling period of 10 seconds.

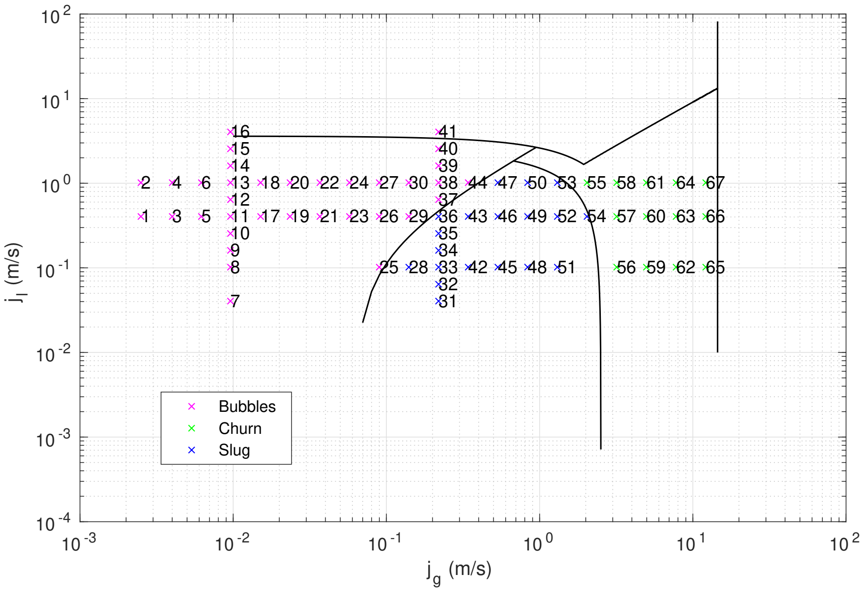


Figure 7 – HZDR data classified according to the map defined in (TAITEL *et al.*, 1980)

Table 3 – Parameters of the TUD Dataset.

Volume Flow Ranges	**Flow Pattern Experimental Points**	
j_g = 0.02 to 8.49 m/s	Bubbly	38
	Slug	35
j_l = 0.03 to 1.86 m/s	Churn	30
	Total	103

3.3 TUD DATASET

The second dataset utilizes data from a specialized test facility designed to evaluate the performance of the algorithm proposed in (KIPPING, 2014). The experimental setup consists of a vertical pipeline with an inner diameter of 50 mm, where fluids are injected at varying velocities to create distinct flow regimes. Flow behavior is closely monitored using a programmable logic controller (PLC) and a differential pressure transducer strategically placed along the test section. The facility, originally constructed to replicate all typical flow patterns in vertical upward flows, has been the foundation for numerous experiments aimed at providing high-fidelity data for validating computational fluid dynamics models.

This dataset offers an extensive range of experimental scenarios covering various parameters, operating conditions, and practical applications. Data were primarily gathered using magnetic-inductive flowmeters (MID) located at different axial points along the pipe. For this study, our analysis concentrated on the data corresponding to an L/D ratio of around 40, ensuring the flow was fully developed. The operational conditions used in this analysis are detailed in Table 3. Each experimental condition is defined by the superficial velocities of the gas j_g and liquid j_l, both measured using reference sensors. Data were collected using a 16×16 WMS, which provided high-resolution measurements. The experiments were conducted with a sampling frequency of 10 kHz, with gas superficial velocities ranging from 0.02 to 8.49 m/s and liquid superficial velocities ranging from 0.03 to 1.86 m/s.

The configuration of the test rig, including the modifications to the gas injection system and the L/D ratio, ensures that the flow regimes are fully developed before measurement. The core of the test apparatus is a 3-meter vertical test section composed of multiple acrylic glass segments, each outfitted with various sensor connections for comprehensive data collection. Further details about the measurement instruments, their operating ranges, and associated uncertainties are provided in (KIPPING, 2014).

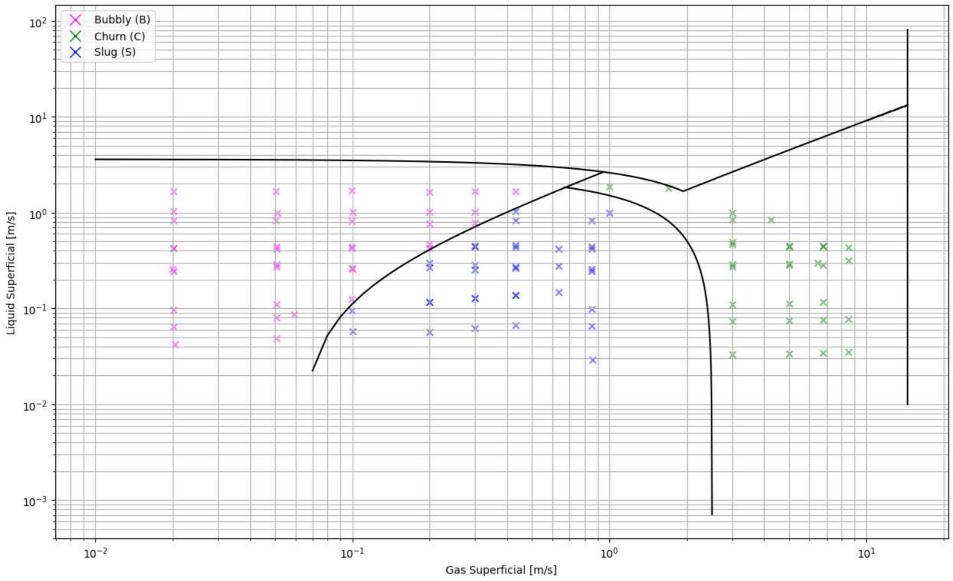


Figure 8 – TUD data classified according to the map defined by Taitel *et al.* (1980).

3.4 VOID FRACTION HISTOGRAM AND FLOW PATTERNS (*MOG+SVM* APPROACH)

The work by Jones and Zuber (1975) is a pioneer in the use of histograms of void fraction time series as a discriminant of flow patterns. In later works, for example, (COSTIGAN; WHALLEY, 1997), such histograms are applied to identify the flow pattern and transition regions.

In the development of this work, void fraction data obtained by the wire-mesh sensor, such as exemplary shown in Figure 9, is used. In this selected example, it is possible to observe the difference between each flow pattern, both by looking directly at the void fraction time series or by observing the histograms generated by each series.

First, we have the bubbly flow pattern. Figure 9(a) shows the time series of the void fraction, while Figure 9(b) shows the histogram. Figure 9(c) presents an illustration of the phase distribution for the bubbly pattern. With this visualization, it is easier to understand why the time series presents minor variations in the void fraction. Figures from 9(d) to 9(f) are relative to the slug pattern. It is possible to visualize two main structures; the first is the liquid slug, which may contain dispersed bubbles, as illustrated in Figure 9(f). The second structure, commonly called the Taylor bubble, is a large gas pocket. Figures 9(g)-(i) present information about the churn pattern. As the gas flow increases, the Taylor bubbles end up collapsing into more misshapen structures, thus generating a slightly more chaotic pattern, as can be seen in Figure 9(i). Finally, the figures from 9(j) to 9(l) show the time series, histogram, and representation of the annular pattern, respectively. This flow pattern is characterized by the highest gas phase velocity among

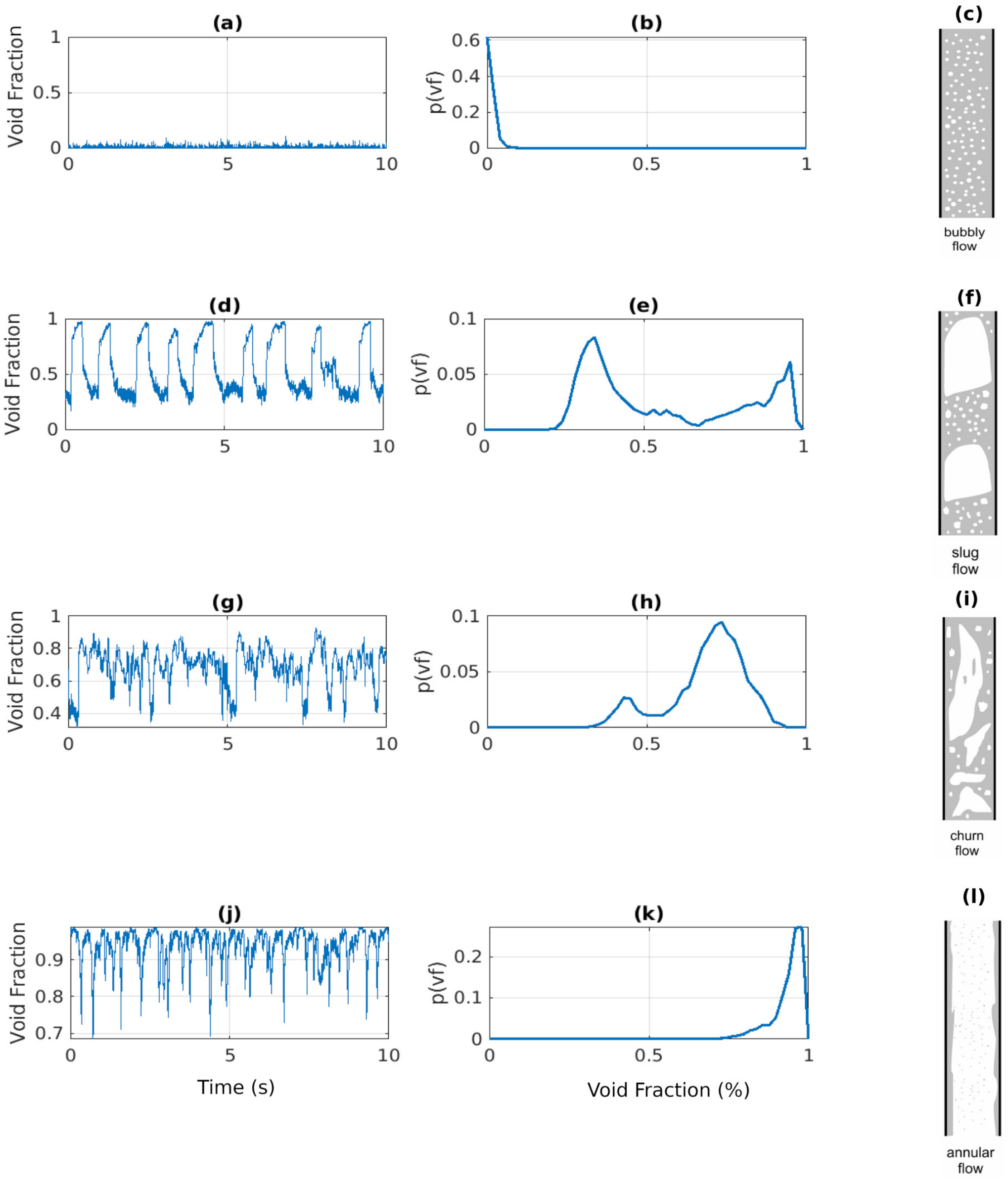


Figure 9 – Time series, histograms, and theoretical phase distribution in pipes for bubbly, slug, churn, and annular flow patterns.

the pattern, thus producing a central column of gas, which may contain dispersed liquid droplets, as seen in Figure 9(l).

In analyzing these distributions, it can be inferred that the slug flow pattern represents a composite behavior that incorporates characteristics of both the bubbly and annular flow patterns. In this case, the liquid slug would correspond to the bubbly pattern, and the Taylor bubble would correspond to the annular pattern. Hence, in a possible simple heuristic, an observer could identify the flow pattern by deciding upon the histogram format. Such behavior of the histogram depends on the flow pattern that has been discussed, for instance, by (COSTIGAN; WHALLEY, 1997). Different flow patterns will describe a different histogram. In this way, transitioning from one flow pattern to another follows a movement of the peaks of the histograms. A single-phase flow of liquid would have a void fraction distribution similar to a Dirac delta centered at zero since, ideally, there is no gas in the system. In the bubble flow pattern, there is a relatively narrow uni-modal distribution with the average at a low void fraction value. On the other end, similarly, a pipe filled with air would have a distribution similar to a Dirac delta centered on 100%, and the annular flow pattern would have a uni-modal distribution with an average in high void fraction values. The slug flow pattern, on the other hand, presents a pseudo-periodic alternation between large gas bubbles (so-called Taylor bubbles) and a slug body (which could be seen as bubbly flow), generating a bi-modal histogram, as shown in Figure 9(d). In this case, the peak in low void fractions represents liquid slugs, which usually contain some dispersed bubbles, and the high void fraction represents Taylor bubbles. As the gas flow increases, the slug pattern transitions to the churn pattern (which is characterized by more chaotic behavior), generating a more prominent peak in the higher void fraction values. Due to this varying multimodal behavior ranging from a uni-modal distribution with low void fractions, passing through bi-modal distributions with low and high void fractions, and then reaching a uni-modal distribution at high void fraction values, a Gaussian Mixture Model (GMM) seems to be the natural choice to cope with all flow pattern and their transitions of gas-liquid flows, hence allowing to classify the pattern based on the extracted and reduced parameter set.

It is noteworthy that the proposed approach does not directly consider temporal aspects, particularly the frame-to-frame correlation. In the first part of this work, we decided to concentrate fully on the automatic analysis of the void fraction signal histogram. This type of analysis has been widespread and used by specialists since its initial proposal by (COSTIGAN; WHALLEY, 1997). Here, the visual analysis generally performed by the experts was replaced by an analytical model of the histogram (Mixture of Gaussians) combined with a pattern classification method.

3.4.1 Gaussian mixture model and void fraction data

The void fraction signal is modeled as a stochastic process of independent and identically distributed (i.i.d) samples with probability density function described by a Gaussian mixture model. Although the independence assumption does not accurately represent the physics involved, it reduces the number of parameters and facilitates their estimation for classification (BOLóN-CANEDO *et al.*, 2015). Furthermore, it resembles the experimental observations as discussed in earlier works, e.g., (COSTIGAN; WHALLEY, 1997).

Let $p(x|\mu, \sigma)$ be a Gaussian probability density function (PDF):

$$p(x|\mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{ -\frac{(x-\mu)^2}{2\sigma^2} \right\}, \tag{4}$$

where μ is the mean of the distribution and σ the standard deviation. We define the mixture model as follows:

$$g(x|\Theta) = \sum_{j=1}^{K} \phi_j p(x|\mu_j, \sigma_j), \tag{5}$$

where $\theta_j = (\phi_j, \mu_j, \sigma_j)$, $\Theta = (\theta_1, \ldots, \theta_K)$, K is the number of components in the model and ϕ a proportion factor for each component, such that$\sum_{j=1}^{K} \phi_j = 1$. Hence, each void fraction time series will generate a vector of parameters $\Theta = (\theta_1, \ldots, \theta_K)$ that are calculated using the concept of Maximum Likelihood and the Expectation-Maximization (EM) algorithm (BILMES *et al.*, 1998).

3.4.1.1 Maximum likelihood function

The likelihood is analyzed according to the parameters and with a set of fixed data, and one of the ways of interpreting it is that the likelihood function indicates how well the parameters explain the observed data in a given distribution. With that, the objective is to find Θ^*, such that:

$$\Theta^* = \arg\max_{\Theta} \; L(\Theta|\mathbf{x}), \tag{6}$$

Which is the set of parameters that best explains the observed data. Often, the natural logarithmic of the likelihood function, i.e., $\ln(L(\Theta|\mathbf{x}))$ is maximized, as it is analytically simpler, especially for densities of the exponential family. Equation 6 defines what is commonly called the maximum likelihood estimator (MLE), which consists of estimating the parameters that best represent a set of data by maximizing the likelihood function. The next section 3.4.1.2 will deal with one of the possible MLE algorithms that will be used to estimate parameters for the Gaussian mixture model.

3.4.1.2 Expectaion maximization algorithm

The Expectation-Maximization algorithm is a general method of estimating parameters by the maximum likelihood that alternates between two stages until convergence. The idea of the algorithm is to maximize the function $\ln\left(L(\Theta|\mathbf{x})\right)$ to estimate the parameter vector Θ.

There are two main applications for the EM algorithm. The first occurs when the data have unobserved values due to problems or limitations in the observation process. The second occurs when the optimization of the likelihood function is analytically intractable. Still, it can be simplified by assuming the existence of additional values or parameters but not observed (BILMES *et al.*, 1998). Therefore, a variable $\mathbf{z}$ is introduced such that,

$$\mathbf{z} = (\mathbf{x},\mathbf{y}), \tag{7}$$

$$p(\mathbf{z}|\Theta) = p(\mathbf{x},\mathbf{y}|\Theta) = p(\mathbf{y}|\mathbf{x},\Theta)p(\mathbf{x}|\Theta), \tag{8}$$

with p being a distribution function, $\mathbf{x}$ the incomplete data set, $\mathbf{y}$ the unobserved data set and $\mathbf{z}$ the complete data set.

Expectation step: first, it is necessary to define an initial value for the parameter vector ($\Theta^{(0)}$). With each iteration, this value will be updated. This defines a function $Q(\Theta|\Theta^{(k)})$ that takes into account the initial value, or the previous iteration, as seen in Equation 9. This function is obtained by calculating the expected value of the likelihood function $L(\Theta|\mathbf{x},\mathbf{y})$ once the observed data set $\mathbf{x}$ and the previous value of the parameter vector $\Theta^{(k)}$.

$$Q(\Theta|\Theta^{(k)}) = E\left\{\ln L(\Theta|\mathbf{x},\mathbf{y})|\mathbf{x},\Theta^{(k)}\right\}. \tag{9}$$

Maximization Step: after defined, it is necessary to find the vector $\Theta^{(k+1)}$ that maximizes the function $Q(\Theta|\Theta^{(k)})$, as shown in Equation 10:

$$\Theta^{(k+1)} = \arg\max_{\Theta}\; Q(\Theta|\Theta^{(k)}). \tag{10}$$

These two procedures are performed alternately until convergence.

In short: An initial value for the parameter vector is defined, $\Theta^{(0)}$. Thereafter, in the expectation step, a function dependent on Θ is defined, which considers the observed data and the initial value defined for the parameter vector $Q(\Theta|\Theta^{(0)})$. Then, the maximization step is performed, where a value $\Theta^{(1)}$ is found that maximizes the function $Q(\Theta|\Theta^{(0)})$. This process is performed k times until convergence when the likelihood function stops increasing, and the value of the estimated parameters tends to be asymptotic. When this happens, the vector $\Theta^{(k)}$ is the vector of the final estimate of the parameters, as can be seen in the flowchart of Figure 10.

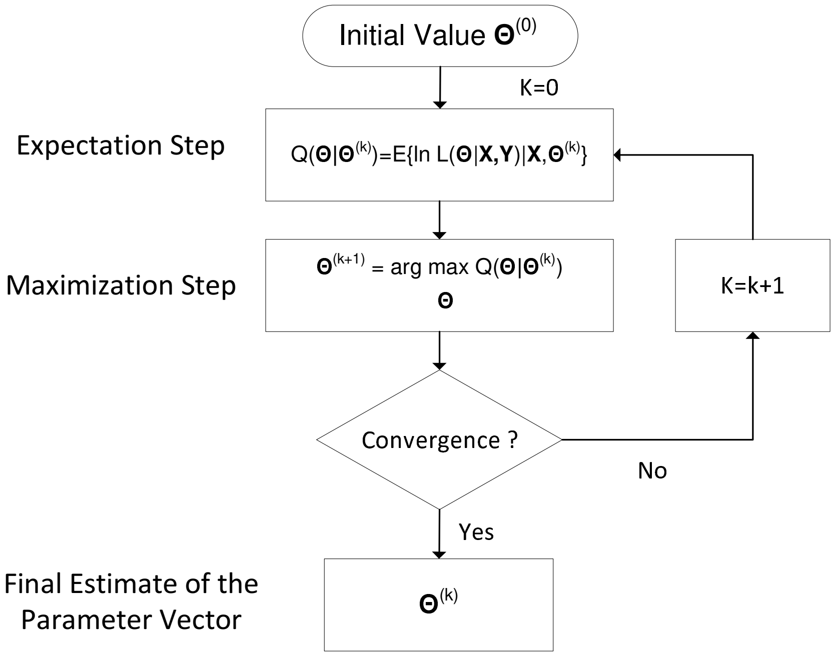

Initial Value $\mathbf{\Theta}^{(0)}$
K=0
Expectation Step
$Q(\mathbf{\Theta}|\mathbf{\Theta}^{(k)})=E\{\ln L(\mathbf{\Theta}|\mathbf{X},\mathbf{Y})|\mathbf{X},\mathbf{\Theta}^{(k)}\}$
Maximization Step
$\mathbf{\Theta}^{(k+1)} = \arg\max_{\mathbf{\Theta}} Q(\mathbf{\Theta}|\mathbf{\Theta}^{(k)})$
K=k+1
Convergence ?
No
Yes
Final Estimate of the Parameter Vector
$\mathbf{\Theta}^{(k)}$

Figure 10 – Flowchart of the Expectation-Maximization algorithm.

3.4.2 Machine learning and flow pattern classification

Once the features of the time series data have been extracted, it is necessary to define an approach that will use the information collected to realize pattern identification. In the first part of this work, the approach chosen will be supervised learning, specifically, using a support vector machine.

3.4.2.1 Binary support vector machine

The support vector machine consists of defining a hyperplane that maximizes the distance between the samples of each class. This technique can be used in cases where classes overlap or not (HAYKIN, 2008). This hyperplane is defined through the supervised training process, where the data are made available in N pairs $(x_1,y_1),\ldots,(x_n,y_n)$ such that $x_i \in \mathbb{R}^p$ are the sample values and $y_i \in \{-1,1\}$ the two possible classes. Thus, the support vector machine approach aims to build a classifier in the form (VAPNIK, 1995):

$$y(x) = \text{sign}\left(\sum_{k=1}^{N} \alpha_k y_k K(\mathbf{x}, \mathbf{x}_k) + b\right), \tag{11}$$

where α_k are real and positive constants, b is a real constant, and $K(\cdot,\cdot)$ is the kernel function, which in this work, can be linear: $K(\mathbf{x},\mathbf{x}_k) = \mathbf{x}_k^T\mathbf{x}$; polynomial of order d: $K(\mathbf{x},\mathbf{x}_k) = (\gamma\mathbf{x}_k^T\mathbf{x} + r)^d$ or Radial basis function (rbf): $K(\mathbf{x},\mathbf{x}_k) = \exp\left(-\gamma \left\|\mathbf{x} - \mathbf{x}_k\right\|^2\right)$, where γ, r and d are hyperparameters.

3.4.2.2 Multiclass Support Vector Machine and Nested Cross-Validation

In this work, the classification problem involves three classes, which are the flow patterns present in the data set. In the training phase, $\frac{N_C(N_C-1)}{2}$ binary SVM classifiers were trained, with N_C being the number of classes. This approach is often referred to as “one-versus-one” or “one-against-one” (BISHOP, 2006).

The diagram in Figure 11 illustrates the flow pattern classification approach proposed in this work. First, in the Preprocessing stage, the void fraction data is measured by the wire-mesh sensor, and the feature extraction is conducted using the Gaussian Mixture Model (section 3.4.1). Then, we have the model selection and generalization stages.

The generalization process consists of an outer loop implemented through a 10-fold cross-validation process. In this process, data are split between the outer training set and the test set. The outer training set goes through the model selection process (inner loop), where it is divided again between the inner training set and validation set. It goes through a 5-fold cross-validation process, where the best hyperparameters are selected from a grid search (which is explained in detail in section 3.2) technique. After model selection in the inner loop, the best model (defined by the hyperparameter from the inner loop) is tested with the test set defined at the beginning of the outer loop. That is, as the outer loop uses a 10-fold cross-validation process, the model evaluation runs 10 times, and on each run, the test dataset is different. It does not contain any overlap with the other test datasets in this step.

It is noteworthy that the Model Selection and Generalization stages can be viewed as a nested cross-validation procedure, with the parameters adjusted in the inner loop (model selection) and the generalization analysis in the outer loop. As discussed by Cawley and Talbot (2010), nested cross-validation estimates the generalization error of the underlying model and its parameter search. To better illustrate this approach, Figure 11 provides a diagram of the nested cross-validation process, and Figure 12 provides a diagram that specifies the split of data.

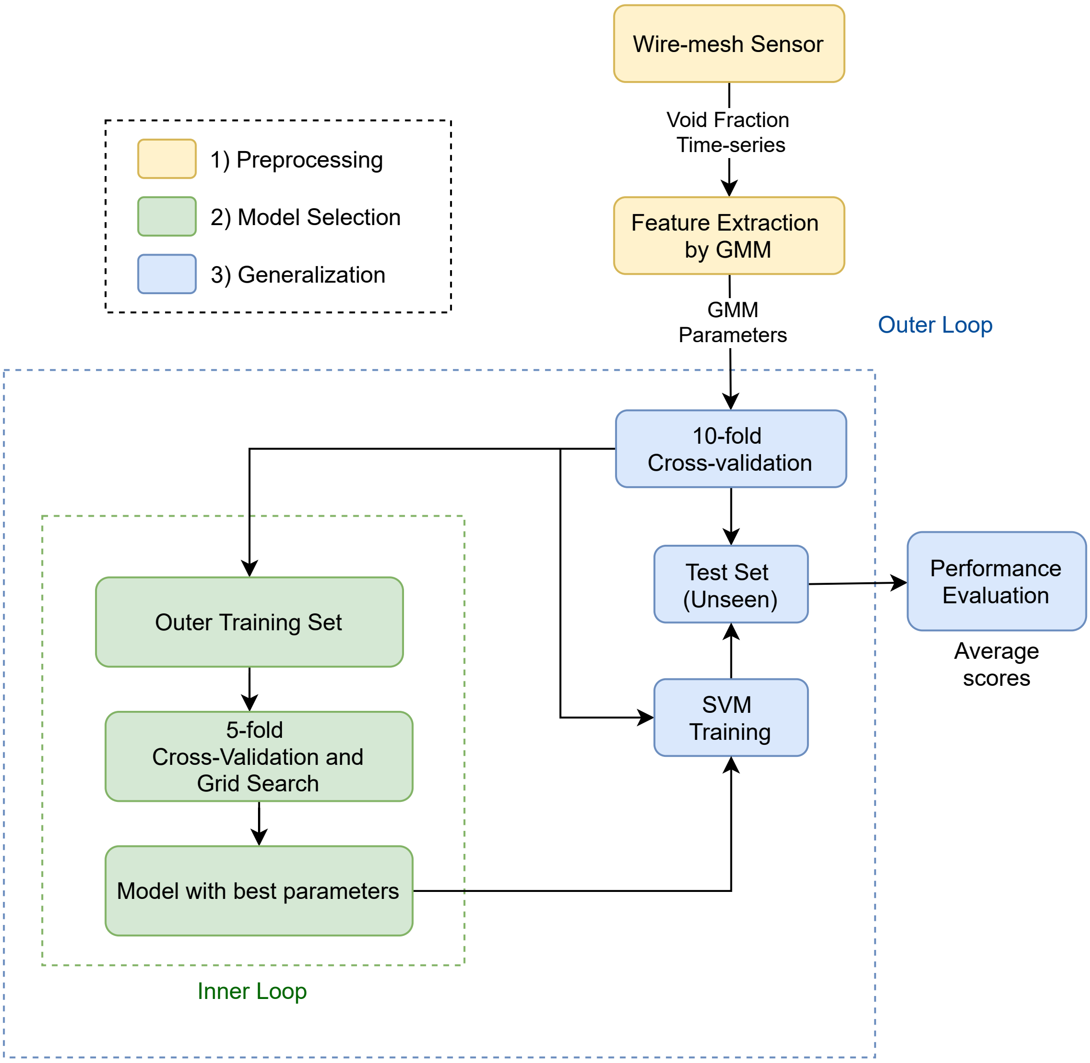


Figure 11 – Block diagram of the approach used in this work. Three subsequent steps are considered. The first (Preprocessing) is to acquire the data, which, in this case, were obtained from HZDR experiments. Then, feature extraction is performed using the Gaussian mixture model. In the second stage (Model Selection), the best SVM hyperparameters are selected with a 5-fold cross-validation procedure. Finally, in the Generalization stage, the performance is conducted by a 10-fold cross-validation outer loop, providing the generalization analysis.

3.5 *SOTA-DL* APPROACH

3.5.1 Time Series Preprocessing

The databases utilized in this study for model training and testing were the HZDR and TUD datasets, which contain comprehensive measurements of two-phase flow patterns essential for developing a robust classification model. However, the sampling frequencies of the two databases were different, necessitating a preprocessing step to ensure consistency. Specifically,

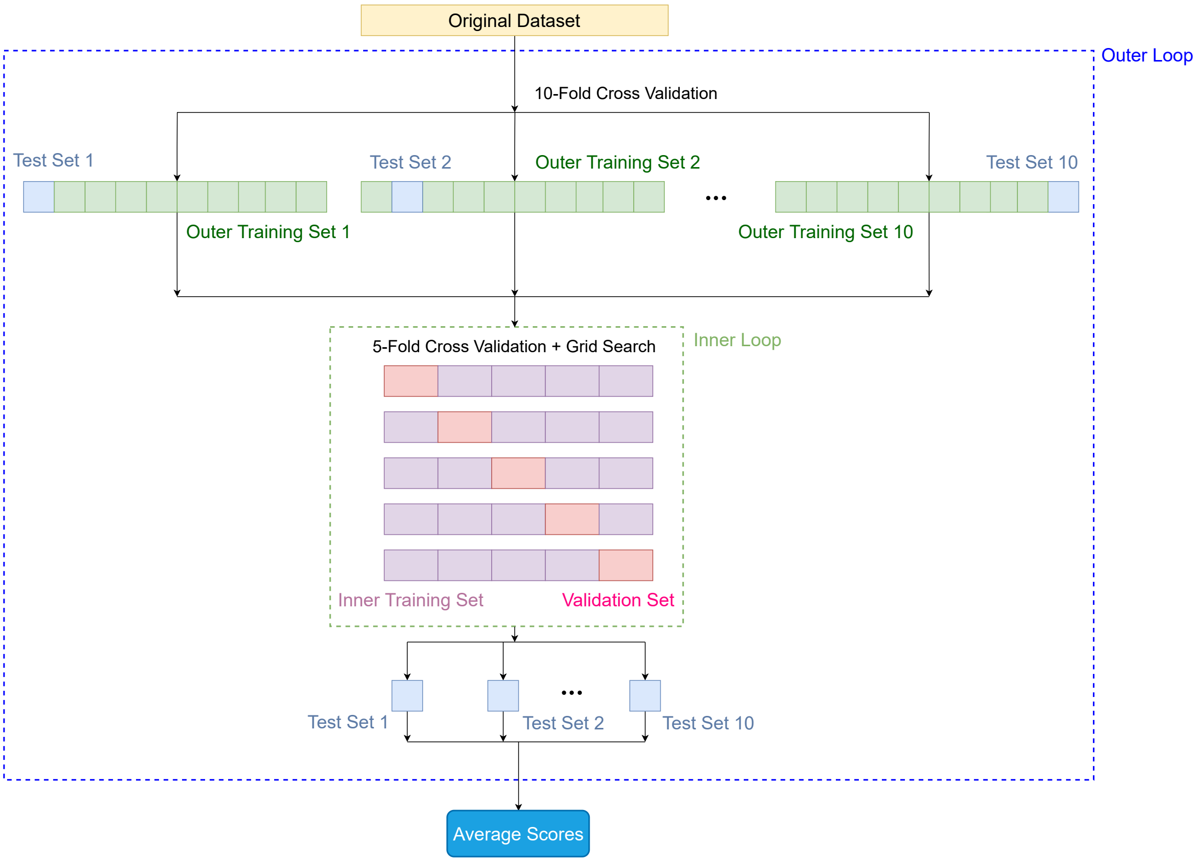


Figure 12 – Diagram of data division during the entire nested cross-validation process. First, the original data is separated between the outer training set and test set in the outer loop, which corresponds to the generalization step performed from the 10-fold cross-validation. Then, in the model selection step, an inner loop is executed, and the outer training data undergoes a second split, where the outer training data is separated into an inner training set and validation set through a 5-fold cross-validation process.

the TUD dataset, sampled initially at 10kHz, was downsampled to 2.5kHz to match the sampling frequency of the HZDR dataset. This adjustment is crucial to avoid any bias from varying data resolutions and to ensure that the model learns from uniformly sampled data.

Normalization is a critical preprocessing step in time series processing, ensuring that the features have a consistent scale. For this study, a min-max scaler normalization was applied to both databases, transforming the data to a range of $[0,1]$. This technique stabilizes the learning process of the neural network and accelerates convergence by preventing features with more extensive numerical ranges from dominating those with smaller ranges.

To enhance the size and variability of the dataset, a windowing technique with overlapping was employed. This approach involves segmenting the time series data into overlapping windows, which are then treated as individual samples. Empirical tests determined a window size of 4000 samples with an overlap of 1000 samples, effectively augmenting the dataset and providing more training instances for the model. This technique captures local patterns and temporal

Table 4 – Parameters of the HZDR Dataset after Preprocessing.

Superficial Velocity Ranges	Flow Pattern Experimental Points	
j_g = 0.0025 to 12.14 m/s	Bubbly	770
	Slug	418
j_l = 0.0405 to 4.047 m/s	Churn	286
	Total	1474

Table 5 – Parameters of the TUD Dataset after Preprocessing.

Volume Flow Ranges	Flow Pattern Experimental Points	
j_g = 0.02 to 8.49 m/s	Bubbly	5586
	Slug	5145
j_l = 0.03 to 1.86 m/s	Churn	4410
	Total	15141

dependencies within the data, leading to improved model performance and generalization.

3.5.2 Time-Series Models

The choice of SOTA Time-Series Models models was based on three distinct types of architectures, all with comparable results in the most recent state-of-the-art time series classification (FOUMANI *et al.*, 2024; TORRES *et al.*, 2021; GUPTA *et al.*, 2020), the first being a convolutional approach, the second with recurrent models, and the third with attention mechanism, given its relevance in several areas of machine and deep learning, in recent literature. Therefore, the following models were selected: ResNet Model (WANG *et al.*, 2017), LSTM-FCN Model (GERS *et al.*, 2000), and TST Model (ZERVEAS *et al.*, 2021), detailed as follows.

3.5.2.1 Resnet Model

As previously stated, our approach is based on SOTA time-series classification methods. The first one, Resnet, is an end-to-end model based on the ResNet idea. As claimed by the original authors (WANG *et al.*, 2017), the model provides a superior generalization capability compared to other approaches, mainly designed for real-world applications. ResNet enhances neural networks by incorporating shortcut connections in each residual block, allowing the gradient to flow directly through the lower layers. In the proposed architecture, each residual block consists of a convolutional layer, a batch normalization layer, and a ReLU activation layer. The convolution operation ($\otimes$) uses three 1-D kernels (**W**) with sizes of 8, 5, and 3. Each block receives an input (the first one is the input signal **x**) and generates an output $\widehat{h}$ with the following subsequent operations:

$$
\begin{aligned}
y_1 &= \mathbf{W}_1 \otimes \mathbf{x} + \mathbf{b}_1,\\
s_1 &= BN(y_1),\\
h_1 &= ReLU(s_1),\\
y_2 &= \mathbf{W}_2 \otimes \mathbf{h}_1 + \mathbf{b}_2,\\
s_2 &= BN(y_2),\\
h_2 &= ReLU(s_2),\\
y_3 &= \mathbf{W}_2 \otimes \mathbf{h}_2 + \mathbf{b}_2,\\
s_3 &= BN(y_3),\\
h_3 &= ReLU(s_3),\\
y &= x + h_3,\\
\widehat{h} &= ReLU(y). \qquad (12)
\end{aligned}
$$

Following guidance from the original ResNet model (HE *et al.*, 2016), pooling operations were omitted to reduce overfitting risk. Batch normalization (BN) is implemented to accelerate convergence and enhance generalization (IOFFE; SZEGEDY, 2015). The architecture consists of three residual blocks containing 64, 128, and 128 filters. Instead of a fully connected layer, a global average pooling layer is used after the convolutional blocks, significantly reducing the model's parameter count. A softmax layer produces the final classification. The general architecture is illustrated in Figure 13.

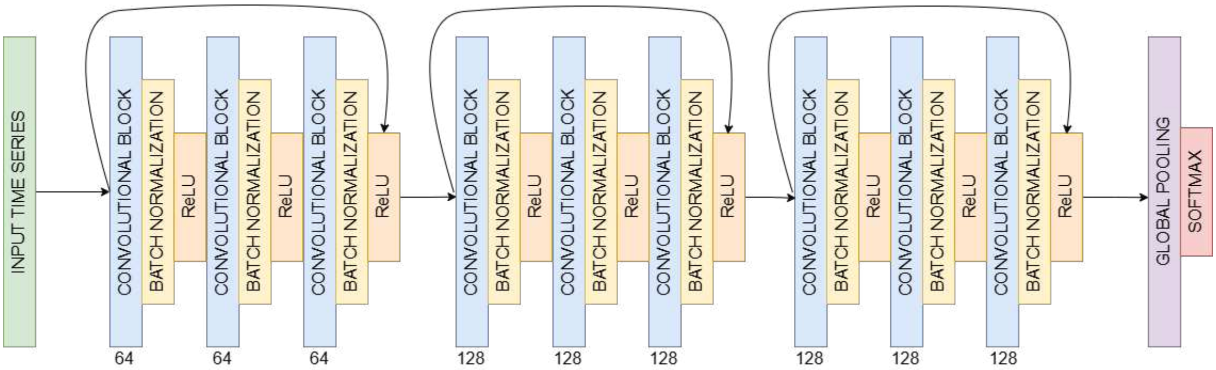


Figure 13 – Architecture of the Resnet Model.

The ResNet is trained with the Adam optimizer, with the learning rate of 0.001, $\beta_1 = 0.9$, $\beta_1 = 0.999$, and $\epsilon = 10^{-8}$, as suggested in Wang *et al.* (2017). The loss function is the categorical cross-entropy, defined as (TIAN *et al.*, 2022):

$$-\sum_{c=1}^{M} y_{o,c}\log(p_{o,c}), \tag{13}$$

in which M is the number of classes, y is the binary indicator (0 or 1) if class label c is the correct classification for observation o, and p the predicted probability observation o is of class c.

3.5.2.2 LSTM-FCN Model

The second approach used in this work, LSTM-FCN, integrates fully convolutional networks with long short-term memory recurrent neural network sub-modules, as described in Karim *et al.* (2018). The architecture comprises two distinct feature extraction segments detailed in Figure 14.

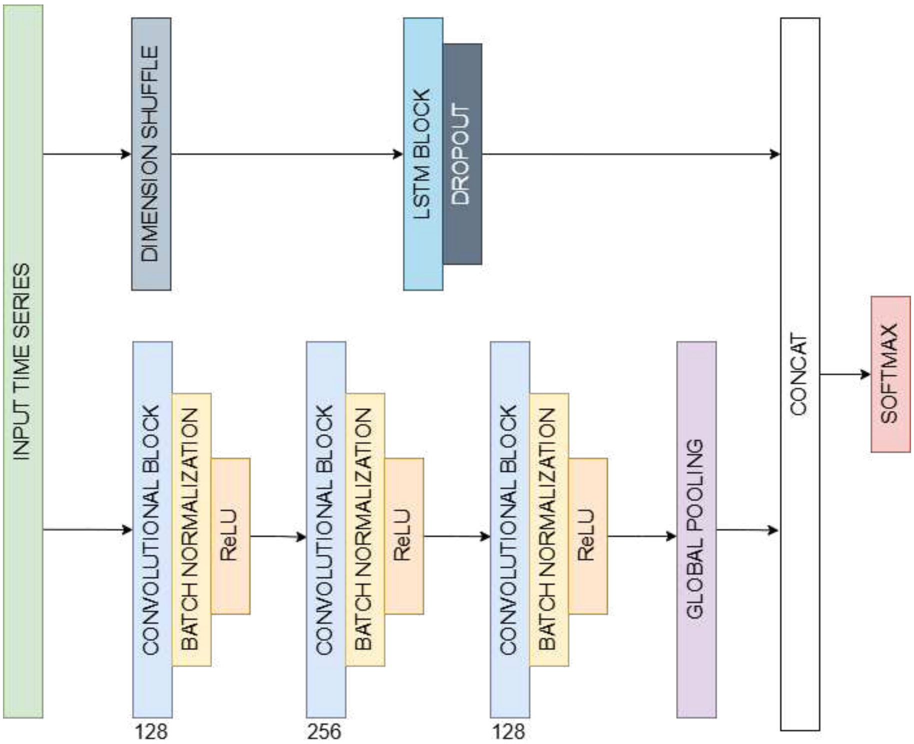


Figure 14 – Architecture of the LSTM-FCN Model.

The first segment is a fully convolutional block. In the previous ResNet model (WANG *et al.*, 2017), the authors reinforced that temporal convolutions have strong performance for time series classification tasks. Hence, Fully Convolutional Networks (FCNs) using temporal convolutions are commonly employed as feature extractors, with global average pooling applied to minimize model parameters before the classification stage. The convolutional structure includes three sequential temporal convolutional blocks with filter sizes of 128, 256, and 128. Each block mirrors the convolution block used in the CNN design suggested by Wang *et al.* (2017). These blocks contain a temporal convolutional layer, paired with batch normalization (IOFFE; SZEGEDY, 2015) (using a momentum of 0.99 and ϵ of 0.001), and conclude with a ReLU activation (ZEILER, 2012). Global average pooling is applied after the last convolutional layer to complete the architecture.

In the proposed methods, this fully convolutional component is enhanced by adding an LSTM block, followed by dropout (SRIVASTAVA *et al.*, 2014). LSTMs offer a solution to the vanishing gradient issue that typically hampers standard recurrent neural networks (RNNs). LSTMs mitigate this challenge by incorporating gating mechanisms within their state dynamics, enabling them to maintain and regulate information across time steps more effectively (HOCHREITER; SCHMIDHUBER, 1997). In each time step, an LSTM unit manages two vectors: a hidden state vector, $\mathbf{h}$, and a memory cell vector, $\mathbf{m}$, which together control how information is updated and output. The specific computations at each time step t as follows (GRAVES, 2012):

$$\begin{aligned}
\mathbf{g}^u &= \sigma(\mathbf{W^u}\mathbf{h}_{t-1} + \mathbf{I}^u\mathbf{x}_t),\\
\mathbf{g}^f &= \sigma(\mathbf{W^f}\mathbf{h}_{t-1} + \mathbf{I}^f\mathbf{x}_t),\\
\mathbf{g}^o &= \sigma(\mathbf{W^o}\mathbf{h}_{t-1} + \mathbf{I}^o\mathbf{x}_t),\\
\mathbf{g}^c &= tanh(\mathbf{W^c}\mathbf{h}_{t-1} + \mathbf{I}^c\mathbf{x}_t),\\
\mathbf{m}_t &= \mathbf{g}^f \odot \mathbf{m}_{t-1} + \mathbf{g}^u \odot \mathbf{g}^c,\\
\mathbf{h}_t &= tanh(\mathbf{g}^o \odot \mathbf{m}_t),
\end{aligned} \tag{14}$$

in which σ is the logistic sigmoid function, $\odot$ represents element-wise multiplication, $\mathbf{W^u}$; $\mathbf{W^f}$; $\mathbf{W^o}$; $\mathbf{W^c}$ are recurrent weight matrices and $\mathbf{I}^u$; $\mathbf{I}^f$; $\mathbf{I}^o$; $\mathbf{I}^c$ are projection matrices.

The fully convolutional block and LSTM block process the same time series data from distinct perspectives (KARIM *et al.*, 2019). The fully convolutional block interprets the series as a sequence of single-variable observations across multiple time steps. For a time series of length N, this block processes the data across N sequential time steps.

Conversely, the LSTM block in this architecture treats the input time series as a multivariate series with just a one-time step. This is achieved through a dimension shuffle layer, which rearranges the temporal dimension. As a result, a univariate series of length N is transformed into a multivariate series with N features at a single time step. Without this dimension shuffle, the LSTM block's performance degrades significantly, leading to rapid overfitting on smaller datasets and an inability to capture long-term dependencies in larger datasets.

Furthermore, using dimension shuffle significantly boosts the model's efficiency, reducing training time substantially. Without dimension shuffling, an LSTM working with a dataset of N time steps and M variables requires N time steps to process each batch of M variables. Dimension shuffling allows the LSTM to reverse this order, handling each batch of N variables in just M time steps instead. This results in a significant speed advantage when M is much smaller than N. Given that the datasets in this work are univariate, the LSTM component only needs a single time step to manage a batch of N variables (YOSINSKI *et al.*, 2014).

In the original approach (KARIM *et al.*, 2018), the authors introduced a refinement technique to iteratively enhance a pre-trained model's performance. As Huang et al. (HUANG *et al.*, 2017) note, the optimization process often encounters multiple local minima along its path. After a model reaches a local minimum in initial training, it can be re-trained with an increased learning rate to move beyond this point and potentially get a more optimal minimum. Gradually decreasing both the learning rate and batch size during this re-training phase allows for a more precise search toward an improved local optimum. Thus, refinement consists of adapting the learning rate and batch size to increase the final performance of the model.

Finally, as illustrated in Figure 14, the output of the global pooling layer and the LSTM block are concatenated and passed onto a softmax classification layer to generate the final output.

3.5.2.3 TST Model

The core of the TST model (ZERVEAS *et al.*, 2021) is an adaptation of the transformer encoder from (VASWANI *et al.*, 2017), modified to handle univariate and multivariate time series data instead of sequences of discrete word indices. The general overview of the TST is presented in Figure 15, detailed as follows.

Firstly, given a training set $\mathbf{X} \in \mathbb{R}^{w \times m}$, i.e., a multivariate time series of length w and m variables in which each feature vector is defined as $\mathbf{x}_t \in \mathbb{R}^m$, the first layer of the model is a linear projection onto a d-dimensional vector space, where d is the dimension of the transformer model sequence element representations (defined as model dimension):

$$\mathbf{u}_t = \mathbf{W}_p \mathbf{x}_t + \mathbf{b}_p, \tag{15}$$

in which $\mathbf{W}_p \in \mathbb{R}^{d \times m}$ and $\mathbf{b}_p \in \mathbb{R}^d$ are learnable parameters. This equation presents the operation for a single time step; however, all input vectors are embedded concurrently by a single matrix-matrix multiplication in the full formulation.

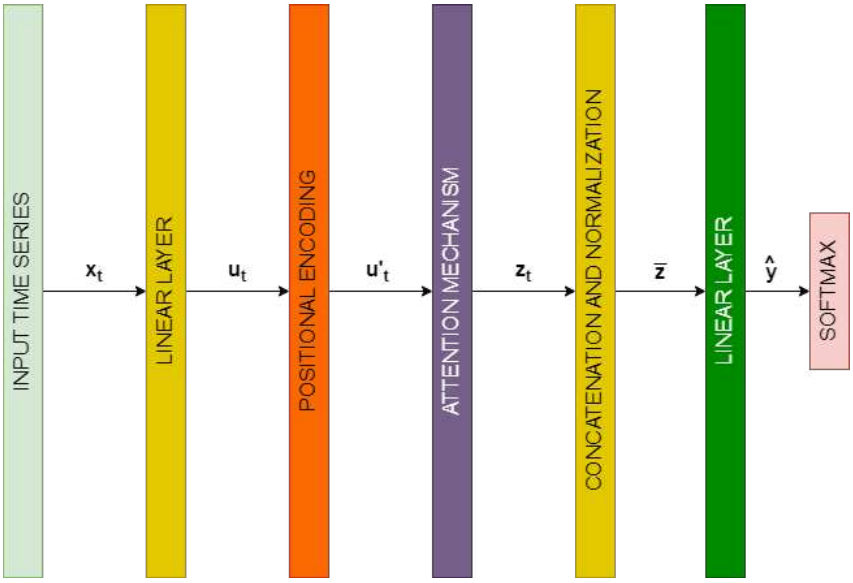


Figure 15 – Architecture of the TST Model.

Because the transformer is a feed-forward architecture and does not inherently account for input order, positional encodings $\mathbf{W}_{pos} \in \mathbb{R}^{w\times d}$ are added to the input vectors $\mathbf{U} \in \mathbb{R}^{w\times d}$ to make them aware of the sequential nature of the time series: $\mathbf{U}' = \mathbf{U} + \mathbf{W}_{pos}$. In the original transformer formulation (VASWANI *et al.*, 2017), positional encodings inject information about the position of samples in a sequence, helping the model understand order, which it otherwise lacks due to its feed-forward architecture. Typically, sinusoidal functions are used to generate these values because they allow the model to generalize across different sequence lengths. However, for TSTs, rather than using deterministic sinusoidal encodings, entirely learnable positional encodings are employed, as the authors found these to perform better on most datasets in their study, as also detailed by Zheng *et al.* (2022) and Zhao *et al.* (2023).

Subsequently, the representation $\mathbf{U}'$, which captures the meaning of each value of the time series together with the positional encoding, is fed into the transformer encoder, which utilizes the self-attention mechanism to enable the model to relate different values within the features. Self-attention works by creating three matrices: a query $\mathbf{Q}$, a key $\mathbf{K}$, and a value $\mathbf{V}$. The model calculates attention scores between each pair of samples by computing the dot product between the query and the key, scaling and normalizing these scores to represent their relevance to each other. The result is used to weight the value vectors, with each token ultimately described as a weighted sum of all samples in the sequence, capturing contextual relationships effectively. This process is further extended in multi-head attention, where multiple self-attention heads capture various aspects of the data, and their outputs are concatenated and linearly transformed to enrich the representation. This mechanism allows transformers to model complex dependencies without relying on sequence order.

More formally, the attention matrix, $\mathbf{A}$, is calculated using Eq. 16, with the softmax function ensuring that the sum of the values in each row equals one:

$$\mathbf{A}(\mathbf{K},\mathbf{Q},\mathbf{V}) = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}}\right)\mathbf{V}. \tag{16}$$

Each row in matrix $\mathbf{A}$ captures the contextual meaning derived from the input embedding, indicating both the position in the time series through positional encoding and each time-stamp's interaction across the sequence. This operation is referred to as single-head attention. The TST model generally leverages a multi-head attention mechanism to enhance performance. In multi-head attention, as with the single-head setup, the input sequence is transformed into three key matrices: $\mathbf{Q}$, a key $\mathbf{K}$, and a value $\mathbf{V}$. These matrices are then multiplied by learnable weight matrices $\mathbf{W}^K$, $\mathbf{W}^Q$, and $\mathbf{W}^V$, all $\in \mathbb{R}^{d\times d}$, to adjust the embedding representation. Following this, the output is split into smaller sub-matrices of dimension d/n_h, where n_h represents the number of parallel attention heads. Each attention head thus processes the entire input sequence but focuses on a different aspect of each embedding by working with these smaller representations. This approach enables each head to contribute uniquely, capturing diverse features and nuances in the data. The attention score for each head is calculated from these segmented matrices, allowing for richer representations across heads, such that (CIESLAK *et al.*, 2024):

$$\mathbf{A}_{head_i}\left(\mathbf{K}\mathbf{W}_i^K, \mathbf{Q}\mathbf{W}_i^Q, \mathbf{V}\mathbf{W}_i^V\right) = \mathbf{z}_t. \tag{17}$$

The final representation vectors $\mathbf{z}_t$ for all time steps are concatenated into a single vector $\overline{\mathbf{z}}_t$, which is the input to a linear output layer with parameters $\mathbf{W}_0$ and $\mathbf{b}_0$. The prediction vector $\hat{\mathbf{y}}$ is passed through a softmax function for classification tasks to produce a probability distribution over the classes:

$$\hat{\mathbf{y}} = \mathbf{W}_0\bar{\mathbf{z}} + \mathbf{b}_0. \tag{18}$$

The classification loss is the cross-entropy between this predicted distribution and the categorical ground truth labels.

3.6 MODEL TRAINING AND CROSS-DATASET EXPERIMENTS

The model training and testing were conducted using various scenarios to evaluate performance comprehensively. First, the datasets were split into training and testing sets within each database, ensuring that the model was trained and tested on the same dataset. This scenario helps in understanding the model's performance when exposed to data with s. Still, its statistical properties. In the second scenario, cross-dataset evaluation was performed, where the model was trained on one dataset (e.g., HZDR) and tested on the other (e.g., TUD), as illustrated in Fig. 16. This scenario is critical for assessing the model's generalization capability and robustness when applied to data from different sources, which may have inherent variances due to other measurement conditions and sampling methodologies.

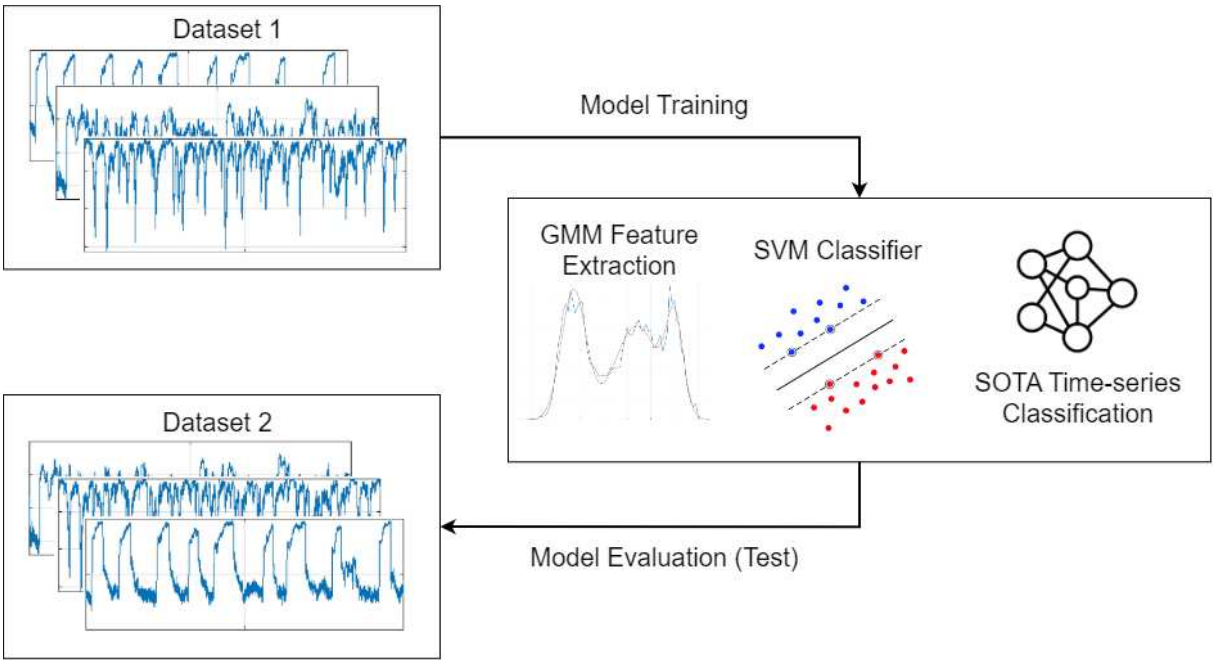


Figure 16 – General overview of the cross-dataset experiments.

3.7 PERFORMANCE METRICS

Evaluating the performance of a classification model is crucial to understanding its effectiveness and reliability. In this study, several performance metrics were used to assess the model, each providing unique insights into various aspects of classification performance. The metrics used include Accuracy, Precision, Recall, F1-Score, and Balanced Accuracy (HASTIE *et al.*, 2009; RASCHKA; MIRJALILI, 2019).

Accuracy is the most straightforward performance metric, defined as the ratio of correctly predicted instances to the total number of cases. It provides a general sense of the model's performance but can be misleading in the presence of class imbalance, where the model might predict the majority class correctly most of the time while performing poorly on the minority class:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}.$$

Precision measures the accuracy of the positive predictions made by the model. It is defined as the ratio of true positive predictions to the total number of positive predictions (true positives plus false positives). Precision is instrumental in scenarios where the cost of false positives is high:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}}.$$

Recall, also known as Sensitivity or True Positive Rate, measures the model's ability to correctly identify all relevant instances. It is defined as the ratio of true positive predictions to the total number of actual positives (true positives plus false negatives). The recall is crucial when the cost of false negatives is high:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}.$$

F1-Score is the harmonic mean of Precision and Recall, providing a single metric that balances both concerns. It is beneficial when there is an uneven class distribution and when both Precision and Recall are important:

$$\text{F1-Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}.$$

Balanced Accuracy addresses the issue of class imbalance by averaging the recall obtained on each class. It provides a more representative performance measure when dealing with imbalanced datasets, ensuring that the performance of the minority class is considered equally with the majority class:

$$\text{Balanced Acc.} = \frac{1}{2}\left(\frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} + \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}\right).$$

These performance metrics collectively provide a comprehensive evaluation of the classification model, ensuring that it is not only accurate in its predictions but also effective across all classes and sensitive to the specific requirements of the application.

3.8 IMPLEMENTATIONS

Implementing the proposed methods in this thesis involved several tools and frameworks, which were carefully selected to ensure efficient processing and analysis of the data. The development was primarily carried out in the Google Colab Pro environment to leverage better computational resources, particularly for executing tasks that required higher processing power. Below are the details of the key implementation aspects:

- **Development Environment**: The codebase was developed using Google Colab Pro, which provided access to enhanced runtimes, including more powerful GPUs and CPUs with additional threads. This setup was crucial for efficiently handling both time-series classification and SVM model training tasks that demanded significant computational resources.

- **SVM and Grid Search**: The support vector machine was implemented using the sklearn library (PEDREGOSA *et al.*, 2011). The GridSearchCV method was employed to optimize hyperparameters. Given the high dimensionality of the parameter space, nested cross-validation was also implemented using sklearn. This process took advantage of multithreading and the maximum number of CPU cores available in the Colab runtime, as sklearn does not support GPU acceleration. This optimization was necessary to reduce the computational time of the cross-validation process, which could otherwise be prohibitive.

- **Time-Series Classification Models**: For time-series classification, the TSAI library (OGUIZA, 2023) was utilized. TSAI, which builds on fastai (HOWARD *et al.*, 2018) and TensorFlow (ABADI *et al.*, 2015), offers state-of-the-art methods for time-series analysis. However, implementing models using TSAI required some adaptation due to differences in syntax and usage compared to more standard machine learning libraries. Despite the availability of documentation, some aspects were not entirely clear, which introduced additional complexity in this part of the implementation.

- **Data Analysis and Preprocessing**: Data analysis was conducted using the pandas library (MCKINNEY, 2010), one of Python's most widely adopted libraries for data manipulation. Time-series preprocessing, an essential step before model training, was performed using a combination of NumPy (HARRIS *et al.*, 2020) and Pandas libraries. These steps ensured that the time-series data was formatted correctly and cleaned, making it suitable for both the machine-learning models and time-series classifiers.

- **Saving Data and Models**: Throughout the testing and validation phases, all relevant data, including training and testing sets, as well as the trained models, were saved. This practice ensured that the experiments could be replicated or reviewed later. The models were stored in formats compatible with the respective libraries (e.g., pickle for sklearn models and native formats for TSAI models), and the data was saved in standard formats such as CSV and numpy arrays for easy access.

4 RESULTS

This chapter presents the results for the *MoG+SVM* and *SOTA-DL* models. For the *MoG+SVM* approach, we perform a preliminary study to validate a classical machine learning approach utilizing handcrafted features, detailing feature selection and hyperparameter tuning. In this initial study, the preprocessing steps detailed in Section **??** were not applied; instead, void fraction data from the HZDR were used without windowing, overlapping, or normalization. Beyond validating the proposed approach, the preliminary study also helped establish the optimal number of components for the Gaussian mixture model (GMM). Following this preliminary phase, comprehensive preprocessing steps and both single-dataset and cross-dataset tests were conducted, as outlined in the Proposed Method chapter, Section 3.6, for both approaches (*MoG+SVM* and *SOTA-DL*). Throughout the Sections, qualitative results and comparisons are also presented.

4.1 PRELIMINARY STUDY FOR THE *MOG+SVM* MODEL

4.1.1 Feature Selection

First, it is necessary to define how many components will be used in the mixture model. By a preliminary analysis of the experimental data, it was found that the histograms of the time series showed uni-, bi-, or tri-modal behavior, depending on the point of operation and the flow pattern. Thus, we defined $K = 3$ in Equation 5. In this case, and for all others where $K < 3$, the GMM will adapt, resetting the values of the parameters that do not contribute to the mixture, that is, that are unnecessary to represent a given PDF. Therefore, the model is prepared for all circumstances in which $K \leq 3$. We also did not observe patterns that require $K > 3$. Hence, the Gaussian mixture model (GMM) generates a vector:

$$\Theta = (\phi_1,\phi_2,\phi_3,\mu_1,\mu_2,\mu_3,\sigma_1,\sigma_2,\sigma_3), \tag{19}$$

for each operation point. Figure 17 shows an exemplary operational point histogram and approximated PDF with the GMM approach and EM algorithm.

Then, once the parameters were obtained, an analysis was performed to verify how these parameters are related to the given flow patterns. Hence, Figures 18, 19 and 20 show how the GMM parameters, namely ϕ_i, μ_i and σ_i (where i = 1,2,3) behave by the different flow

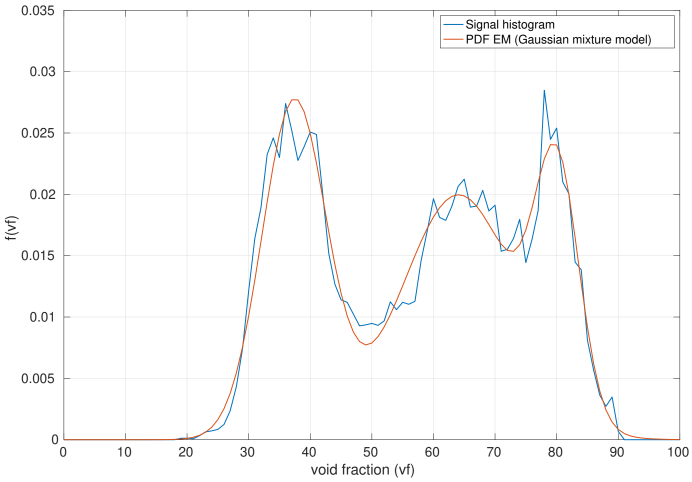


Figure 17 – PDF approximation by the Gaussian mixture model.

patterns. In order to present in a compact manner, the time-averaged void fraction is used in the x-axis. It is possible to observe that all parameters present some degree of dependence on the flow pattern. However, the best one, with the lowest overlap, is the mean values μ. The second best parameter with moderate overlap between flow patterns is the proportion factors ϕ. Finally, the feature σ shows the most accentuated overlap, which can degrade the performance of the latter classification algorithm.

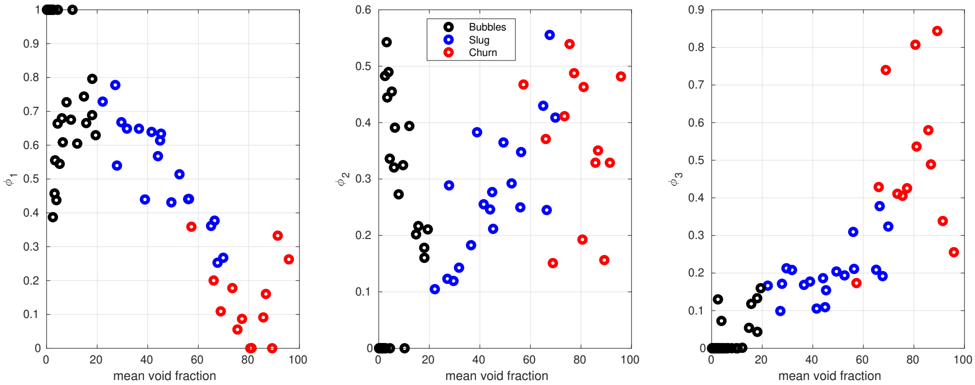


Figure 18 – Relationship between the proportion factors ϕ of the Gaussian mixture model and the flow patterns.

4.1.2 Flow pattern classification

To realize the classification, the “one-against-one” approach is used for the multiclass SVM. In this step, tests were performed with different features, kernels, and parameters. This type of approach is widely used, mainly to avoid overfitting and underfitting, and has been

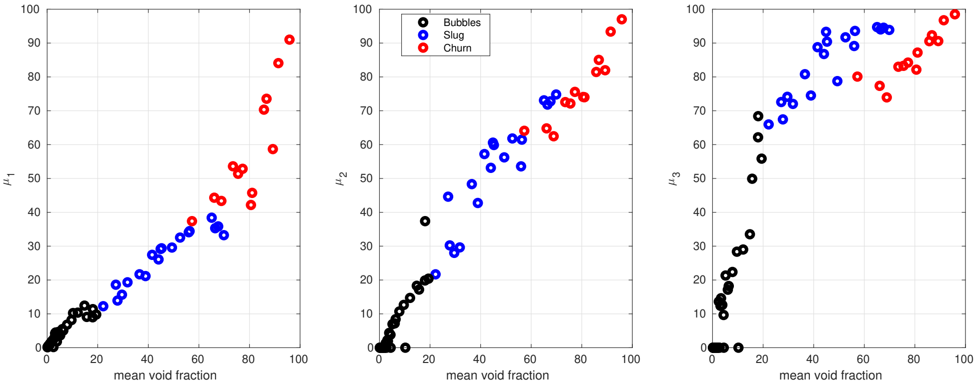


Figure 19 – Relationship between the means μ of the Gaussian mixture model and the flow patterns.

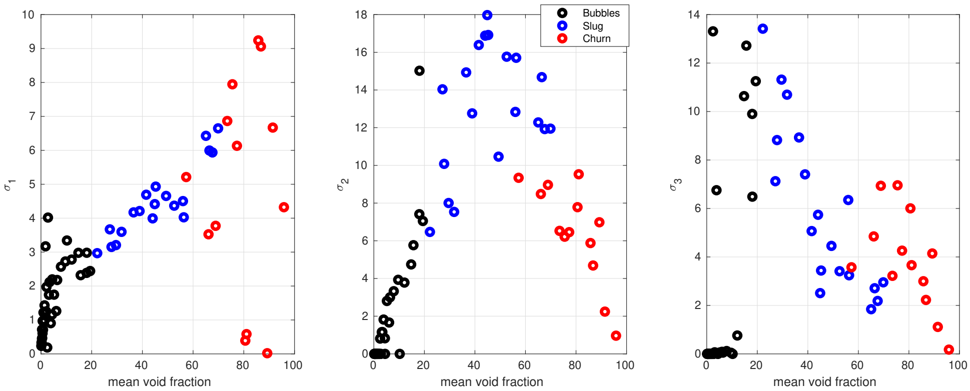


Figure 20 – Relationship between standard deviations σ of the Gaussian mixture model and the flow patterns.

present in diverse papers in the field of multiphase flow pattern classification with machine learning (EYO *et al.*, 2019; WANG; ZHANG, 2009; ZHANG *et al.*, 2020b).

To test the performance of the classifiers, the nested cross-validation methodology was used, as mentioned in section 3.4.2.2. As explained above, to determine the performance of the classifiers, an internal 5-fold cross-validation step was used to determine the best set of hyperparameters. After that, an outer loop performs the assessment of the best models using 10-fold cross-validation.

In the inner loop, the parameters in the grid search are:

- Optimization parameter of SVM: $C \in \{1, 10, 1 \times 10^2, 1 \times 10^3\}$;
- Rbf kernel: $\gamma \in \{1 \times 10^{-4}, 1 \times 10^{-3}, \frac{1}{7}, \frac{1}{5}, \frac{1}{3}\}$;
- Polynomial kernel: $\gamma \in \{1 \times 10^{-1}, 1, 10\}$, $d \in \{1, 2, 3, 4\}$, and $r \in \{0, 1 \times 10^{-1}, 1, 10\}$.

It is noteworthy that for the linear kernel, the only hyperparameter used is the optimization of the SVM itself. To perform the feature selection step, the Figure 11 process was performed for each set of test features, where $N = 3$ represents the vector with all the features

ϕ, μ and σ, $N = 2$ corresponds to the vector with features ϕ and μ, and $N = 1$ is a vector with feature μ.

Table 6 summarizes the results obtained by the proposed method. As explained earlier, the outer loop is executed 10 times (10-fold cross-validation), and in each of these runs, the inner loop output is the best hyperparameter set found in a grid search. As the process is repeated for three feature configurations, we have a total of 30 realizations. In the three cases where the polynomial kernel appears as the best alternative, the hyperparameters d and r are equal to 3.0 and 0.0, respectively, as also shown at the bottom of the table. The "N" column indicates which feature configuration was used, and the "Frequency" column indicates how many times that set of hyperparameters appeared in the process. When the value of column "Frequency" is higher than 1, it is implied that the accuracy and F-score values are the mean values between all occurrences. The results were ordered taking into account frequency and greateAccuracycy, as done in other works that use this nested cross-validation process, e.g., Krstajic *et al.* (2014).

Table 6 – SVM results of testing phase from outer cross-validation.

kernel	C	γ	N	Frequency	Accuracy	F-Score
rbf	1	1×10^{-3}	3	3	0.9563 ± 0.0754	0.9427 ± 0.1007
rbf	1	1×10^{-3}	1	3	0.9476 ± 0.0850	0.9369 ± 0.1077
rbf	1	1×10^{-3}	2	3	0.9476 ± 0.0850	0.9369 ± 0.1077
rbf	100	1×10^{-3}	1	3	0.9357 ± 0.0897	0.9076 ± 0.1462
rbf	100	1×10^{-3}	2	3	0.9357 ± 0.0897	0.9076 ± 0.1462
rbf	10	1×10^{-4}	1	2	0.9548 ± 0.0796	0.9445 ± 0.1062
rbf	10	1×10^{-4}	2	2	0.9548 ± 0.0796	0.9445 ± 0.1062
rbf	10	1×10^{-4}	3	2	0.9488 ± 0.0821	0.9371 ± 0.1026
linear	1	$-$	3	2	0.9488 ± 0.0576	0.9302 ± 0.0880
poly	1	1×10^{-1}	1	1	0.9571 ± 0.0655	0.9322 ± 0.1165
poly	1	1×10^{-1}	2	1	0.9571 ± 0.0655	0.9322 ± 0.1165
poly	1	1×10^{-1}	3	1	0.9571 ± 0.0655	0.9241 ± 0.1269
rbf	100	1×10^{-3}	3	1	0.9548 ± 0.0694	0.9305 ± 0.1209
rbf	10	1×10^{-3}	3	1	0.9381 ± 0.0762	0.9185 ± 0.1018
linear	1	$-$	1	1	0.9286 ± 0.0714	0.8905 ± 0.1376
linear	1	$-$	2	1	0.9286 ± 0.0714	0.8905 ± 0.1376

For the polynomial kernel, the values of hyperparameters d and r found in the inner loop were 3.0 and 0.0, respectively.

It is interesting to verify that the three best classifiers have the same set of hyperparameters (kernel rbf with $C = 1$ and $\gamma = 1 \times 10^{-3}$), but with different feature configurations. The classifier that considers the three features (i.e., ϕ, μ and σ) presents the best performance witAccuracycy equals to 0.9563 ± 0.0754, and F-Score equals to 0.9427 ± 0.1007. This classifier also has the highest accuracy and F-Score values along with the lowest standard deviations.

4.2 HYPERPARAMETER TUNING FOR *SOTA-DL* MODELS

The full nested or cross-validation is unfeasible for the *SOTA-DL* models due to the high computational complexity in model training. In this case, we chose to follow the recommendations of the original authors of the papers (KARIM *et al.*, 2018; WANG *et al.*, 2017; ZERVEAS *et al.*, 2021), in addition to a specific refinement of some parameters, using only a hold-out validation, with the dataset being split with 80% of the HZDR data used for training and 20% for the validation. Table 7, 8, and 9 present the final hyperparameters used in the ResNet, LSTM-FCN, and TST classification models, respectively.

Table 7 – Hyperparameters used in ResNet classification model.

Parameter	Description	Value
$block$	number of conv. blocks	3
n	number of classes	3
–	number of conv. layers per block	3
k_1	number of filters layer 1	64
k_2	number of filters layer 2	128
k_3	number of filters layer 3	128
activation	activation function	ReLU

Table 8 – Hyperparameters used in LSTM-FCN classification model.

Parameter	Description	Value
$block$	number of conv. blocks	3
n	number of classes	3
–	number of conv. layers per block	1
k_1	number of filters layer 1	128
k_2	number of filters layer 2	256
k_3	number of filters layer 3	128
$\mathbf{h}$	number of LSTM hidden units	100
activation	activation function	ReLU

Table 9 – Hyperparameters used in TST classification model.

Parameter	Description	Value
m	number of features	8
n	number of classes	3
w	time steps	50
d	dimension of the model	128
n_h	number of attention heads	16
d_k	size of queries and keys	32
d_v	size of values	32
d_{ff}	dimension of the FC layer	256
dropout	dropout in the encoder	0.1
activation	activation function	GELU
n_l	number of sub-encoder-layers	3
dropout$_{fc}$	dropout of FC layer	0
pe	positional encoder	Uniform

4.3 COMPARISONS

In this section, after the preliminary study, which aimed to validate the proposed approach for SVM with feature extraction using MoG and hyperparameter tuning through nested cross-validation, we present the results of our experiments using two datasets: HZDR and TUD. We evaluate the performance of various models using standard metrics, including Accuracy, Precision, Recall, F1-Score, and Balanced Accuracy. For single dataset evaluations, we employ 3-fold Cross-Validation. Additionally, we perform cross-dataset evaluations where models trained on one dataset are tested on the other to assess model generalization. For the *MoG-SVM*, in the single-dataset comparison, the nested cross-validation process was rerun to identify the best hyperparameters for the preprocessed datasets. For the cross-dataset studies, however, the same hyperparameters were retained to enable an analysis of the model's generalization capacity.

4.3.1 HZDR Dataset

Table 10 summarizes the performance metrics for different models. The results indicate that ResNet achieves the highest overall performance across all metrics, with an average Accuracy, Precision, Recall, and F1-Score of 0.986 ± 0.004 and a Balanced Accuracy of 0.983 ± 0.005. LSTM-FCN also performs well, showing robust results with an average Accuracy, Precision, and Recall of 0.957 ± 0.01 and an F1-Score of 0.956 ± 0.01. TSTPlus performs with slightly lower metrics, achieving an Accuracy of 0.91 ± 0.004 and a Balanced Accuracy of 0.898 ± 0.003. *MoG-SVM* demonstrates moderate performance with an Accuracy, Precision, Recall, and F1-Score of 0.949 ± 0.007 and a Balanced Accuracy of 0.941 ± 0.01.

Figure 21 illustrates the confusion matrices for each model, providing further insight into their performance on the HZDR dataset. ResNet (Figure 21(a)) demonstrates excellent classification capabilities, with only a single misclassification in the Bubbly and three in Slug categories, and no errors in identifying Churn, resulting in the highesAccuracycy of 98.64%. LSTM-FCN (Figure 21(b)) also shows strong performance, with minor misclassifications, particularly in the Slug category, leading to an overall accuracy of 96.95%. TSTPlus (Figure 21(c)) has more difficulty differentiating between the Slug class and other categories, resulting in the lowesAccuracycy of 90.17% among the models evaluated. Finally, *MoG-SVM* (Figure 21(d)) performs better than TSTPlus but still exhibits misclassification, especially in identifying Slug, with an overall accuracy of 94.58%. These results highlight the challenges in classifying the Slug category accurately, which is common across all models.

Table 10 – Performance metrics for different models on the HZDR dataset using 3-Fold Cross-Validation.

Model	Accuracy	Precision	Recall	F1-Score	Balanced Acc.
ResNet	0.986 ± 0.004	0.986 ± 0.004	0.986 ± 0.004	0.986 ± 0.004	0.983 ± 0.005
LSTM-FCN	0.957 ± 0.01	0.957 ± 0.01	0.957 ± 0.01	0.956 ± 0.01	0.951 ± 0.013
TSTPlus	0.91 ± 0.004	0.913 ± 0.006	0.91 ± 0.004	0.911 ± 0.005	0.898 ± 0.003
MoG-SVM	0.949 ± 0.007	0.949 ± 0.008	0.949 ± 0.007	0.949 ± 0.007	0.941 ± 0.01

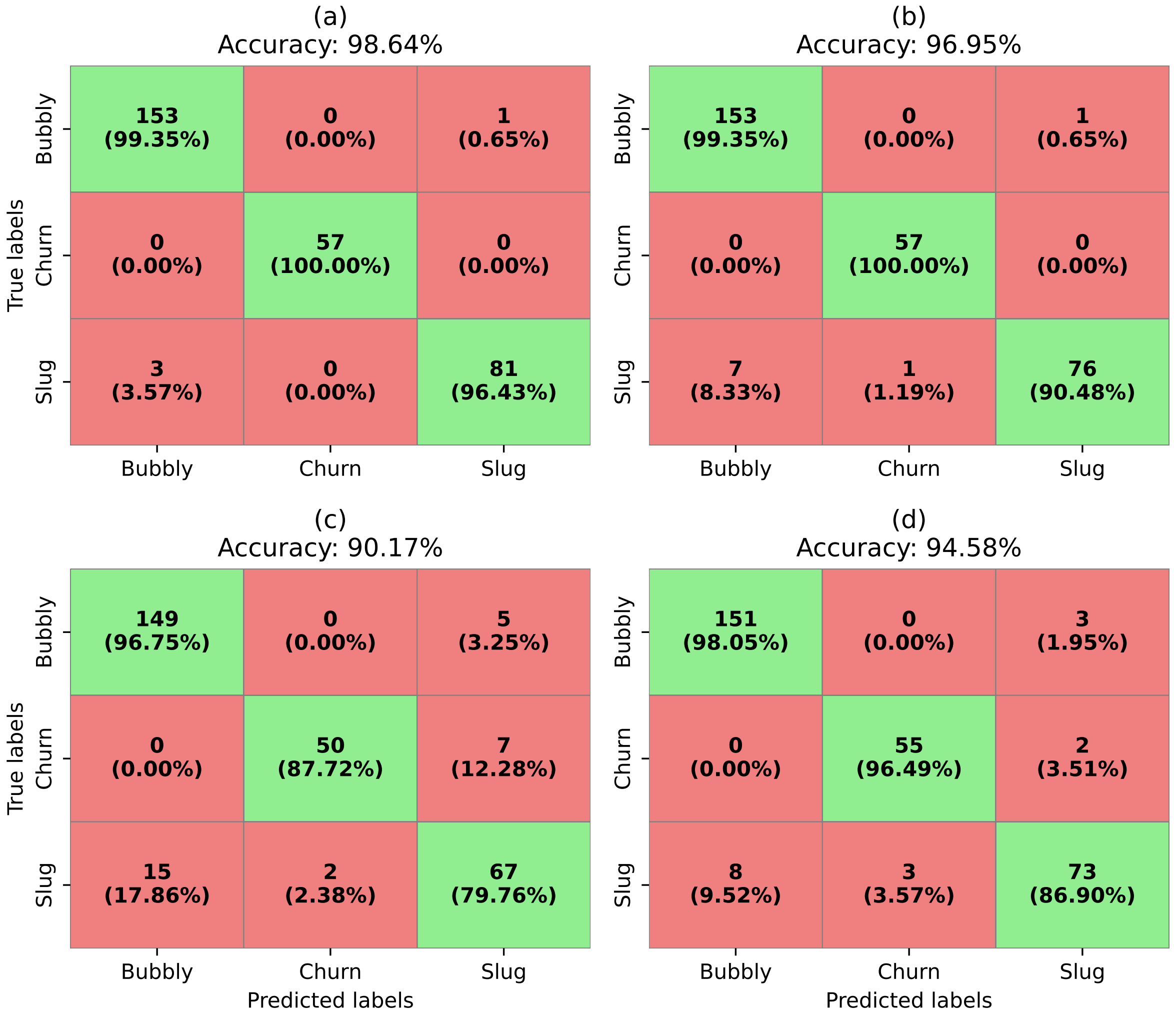


Figure 21 – Confusion matrices for models trained with the HZDR dataset. (a) ResNet. (b) LSTM-FCN. (c) TSTPlus. (d) *MoG-SVM*.

4.3.2 TUD Dataset

The TUD dataset is also evaluated using 3-fold Cross-Validation, as shown in Table 11. ResNet again exhibits superior performance, achieving an average Accuracy, Precision, Recall, and F1-Score of 0.992 ± 0.002, along with a Balanced Accuracy of 0.993 ± 0.001. LSTM-FCN follows closely, showing high consistency across all metrics, each at 0.986 ± 0.001, indicating its robustness. TSTPlus, while performing well, has slightly lower metrics, with an Accuracy of 0.961 ± 0.003 and a Balanced Accuracy of 0.962 ± 0.004. *MoG-SVM* shows the lowest performance among the models, with consistent metrics across Accuracy, Precision, Recall, and F1-Score, all at 0.954 ± 0.002, indicating some challenges in classification tasks on the TUD dataset.

Table 11 – Performance metrics for different models on the TUD dataset using 3-Fold Cross-Validation.

Model	Accuracy	Precision	Recall	F1-Score	Balanced Acc.
ResNet	0.992 ± 0.002	0.992 ± 0.002	0.992 ± 0.002	0.992 ± 0.002	0.993 ± 0.001
LSTM-FCN	0.986 ± 0.001	0.986 ± 0.001	0.986 ± 0.001	0.986 ± 0.001	0.986 ± 0.001
TSTPlus	0.961 ± 0.003	0.962 ± 0.003	0.961 ± 0.003	0.961 ± 0.003	0.962 ± 0.004
MoG-SVM	0.954 ± 0.002	0.954 ± 0.002	0.954 ± 0.002	0.954 ± 0.002	0.954 ± 0.002

Figure 22 illustrates the confusion matrices for the models trained on the TUD dataset, further aiding in understanding their generalization performance. ResNet (Figure 22(a)) exhibits almost perfect classification across all categories, a trait consistent across datasets, which supports its high generalization capacity. LSTM-FCN (Figure 22(b)), while nearly matching ResNet in performance, displays minor discrepancies in classification, hinting at potential areas for improving its generalization to different contexts. TSTPlus (Figure 22(c)) shows more confusion among categories compared to ResNet and LSTM-FCN, reflecting its more limited generalization. *MoG-SVM* (Figure 22(d)) struggles more across datasets, confirming its comparative difficulty in generalizing.

Comparing this performance with that on the HZDR dataset reveals notable insights into the models' generalization abilities. Both ResNet and LSTM-FCN consistently perform well across datasets, validating their robustness and adaptability. *MoG-SVM* and TSTPlus performed worse than the other two models. TSTPlus was the only model that showed relatively poorer performance in the case of the HZDR dataset, possibly indicating the need for more data to properly train the model since the TUD, with larger data availability, presented comparable performance among SOTA models. In general, SOTA models present superior average performance for each dataset evaluated, indicating that choosing this approach is more promising for time series generated for classifying flow patterns.

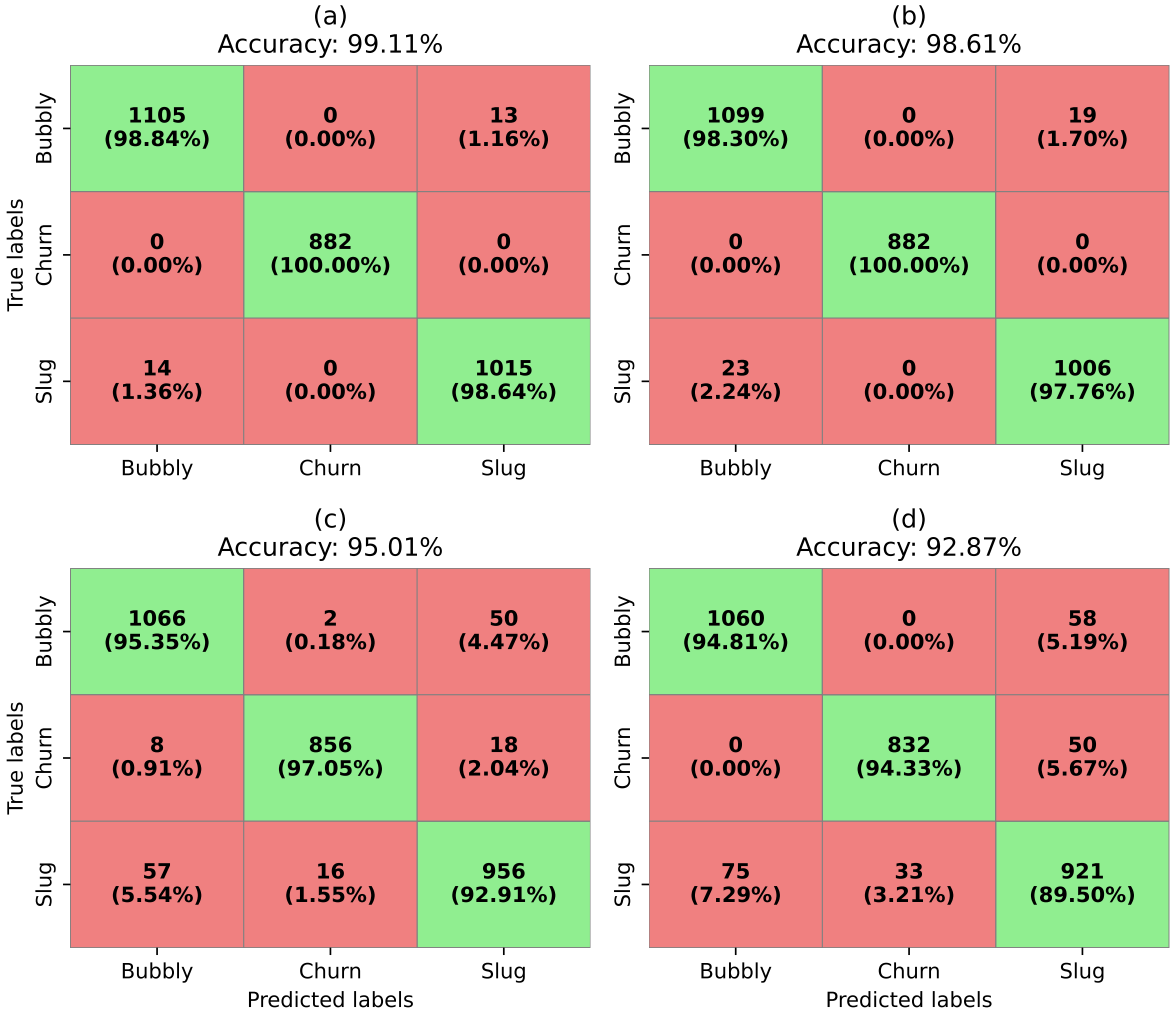


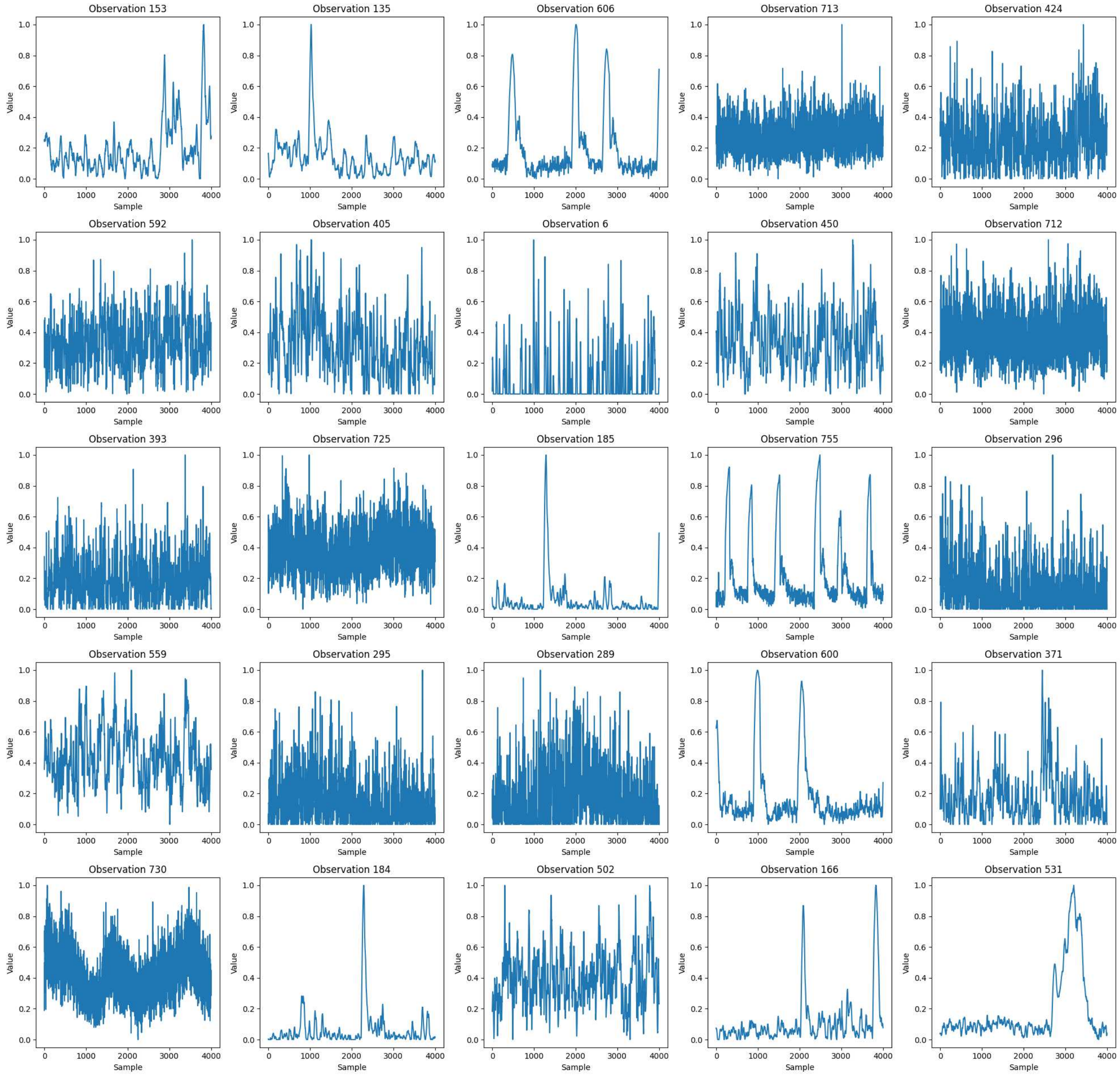
Figure 22 – Confusion matrices for models trained with the TUD dataset. (a) ResNet. (b) LSTM-FCN. (c) TSTPlus. (d) *MoG-SVM*

4.3.3 Qualitative Analysis

4.3.3.1 Time series

By analyzing the confusion matrices presented in Figures 21 and 22, a clear pattern of misclassification emerges. Most misclassifications are found in operational points associated with the Slug flow pattern that were incorrectly classified as Bubbly, as well as operational points of the Bubbly flow pattern misclassified as Slug. This trend is observed in both the *MoG+SVM* and *SOTA-DL* approaches.

To gain further insight into this behavior, we generated figures illustrating samples of the preprocessed time series selected at random. Figure 23 displays 25 time-series of the Bubbly pattern for the HZDR dataset, Figure 24 shows 25 time-series of the Churn pattern for the same dataset, and Figure 25 presents 25 time-series of the Slug pattern. Examining these samples allows us to identify signal characteristics that may contribute to the observed misclassifications between the Slug and Bubbly patterns.



Observation 153 Observation 135 Observation 606 Observation 713 Observation 424

Observation 592 Observation 405 Observation 6 Observation 450 Observation 712

Observation 393 Observation 725 Observation 185 Observation 755 Observation 296

Observation 559 Observation 295 Observation 289 Observation 600 Observation 371

Observation 730 Observation 184 Observation 502 Observation 166 Observation 531

Figure 23 – Plot of randomly selected observations from time series of the Bubbly flow pattern in the HZDR dataset.

Analyzing each figure in more detail allows us to draw further conclusions about the characteristics contributing to the misclassification between the Slug and Bubbly patterns. In Figure 23, which depicts time series classified as Bubbly, certain series exhibit characteristics resembling those of the Slug pattern. Notable examples include observations 606, 755, and 600, which display features that align more closely with the typical Slug series. This resemblance suggests a degree of overlap in the signal attributes of these two patterns that likely contributes to the observed misclassification in the confusion matrices.

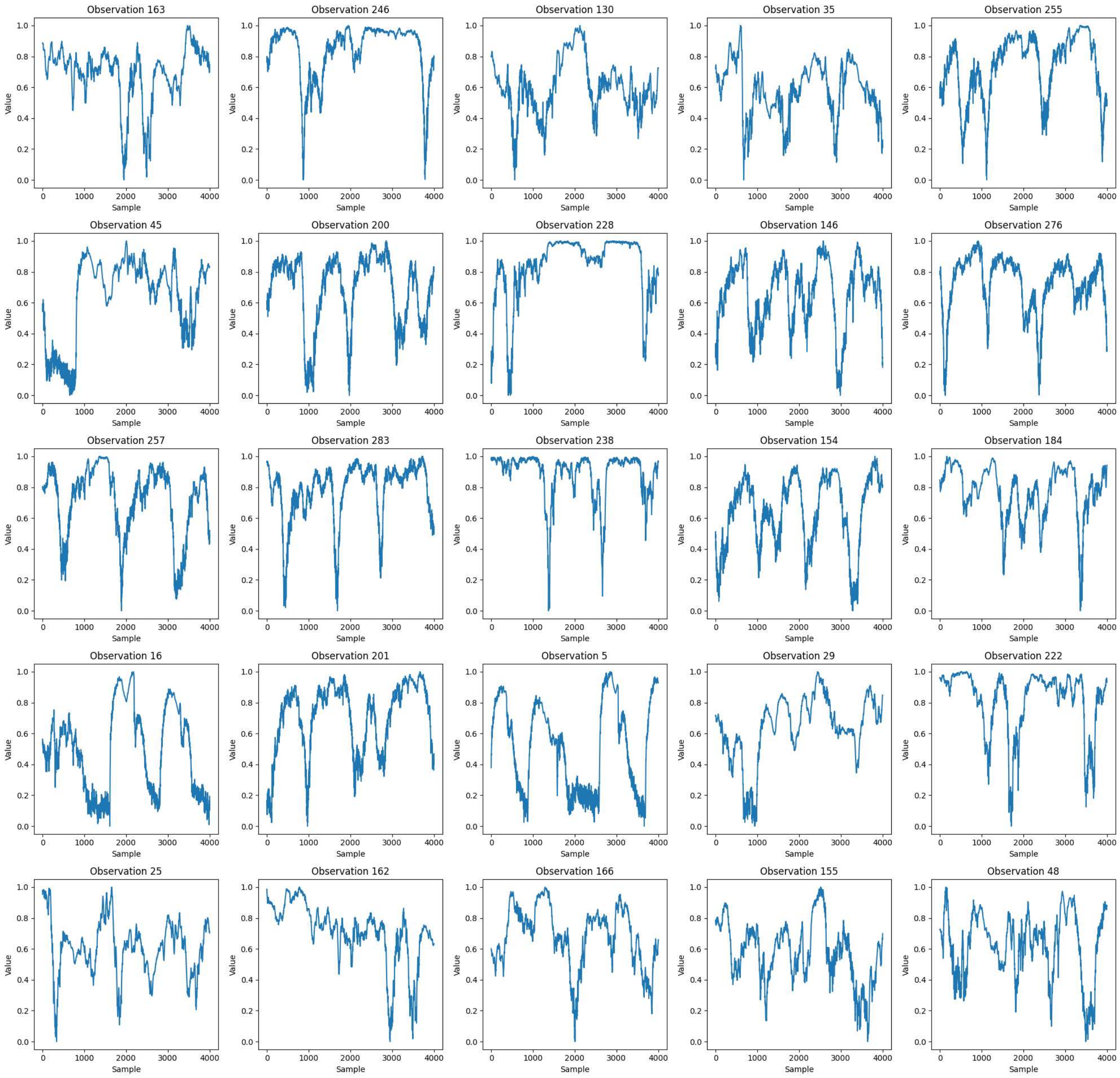


Figure 24 – Plot of randomly selected observations from time series of the Churn flow pattern in the HZDR dataset.

To better understand this overlap, we can compare the series from observations 606, 755, and 600 in Figure 23 with those in observations 117 and 188 in Figure 25, which represent samples of the Slug pattern. Both series sets share visual similarities, reinforcing the hypothesis that certain operational conditions or signal behaviors are common to both patterns, making them challenging to differentiate through automated classification.

In contrast, the series associated with the Churn pattern, as displayed in Figure 24, exhibit distinct characteristics that set them apart from both the Bubbly and Slug patterns shown in Figures 23 and 25. The Churn series tends to display a unique behavior that does not overlap with the Bubbly and Slug characteristics, which may explain why the Churn pattern has lower misclassification rates in most cases compared to the other two. This distinctiveness highlights the

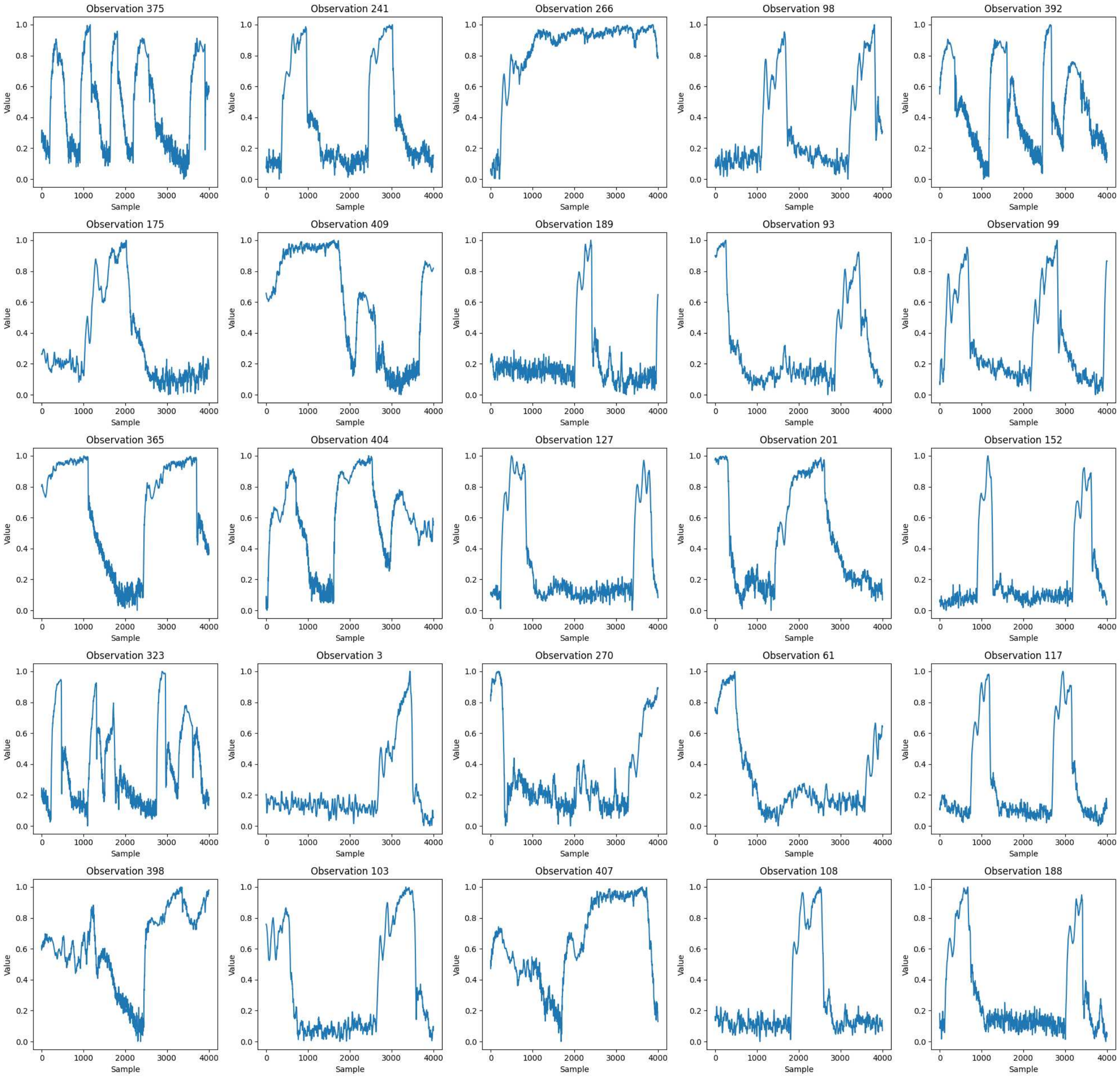


Figure 25 – Plot of randomly selected observations from time series of the Slug flow pattern in the HZDR dataset.

more separable nature of the Churn pattern within the dataset, suggesting that the classification methods are more effective in identifying it compared to the more visually similar Bubbly and Slug patterns.

This similarity in time series patterns for the Slug and Bubbly flows may occur at points close to the pattern transition area on the map that was used for the initial pre-qualification. It is worth noting that this map serves as an approximate model for classification and may not fully capture the nuances in boundary regions where flow characteristics can overlap. As a result, the proximity of operational points to these transition areas may contribute to the challenge of correctly classifying these flow patterns.

Similar figures were generated for the TUD dataset. Figure 26 presents 25 randomly

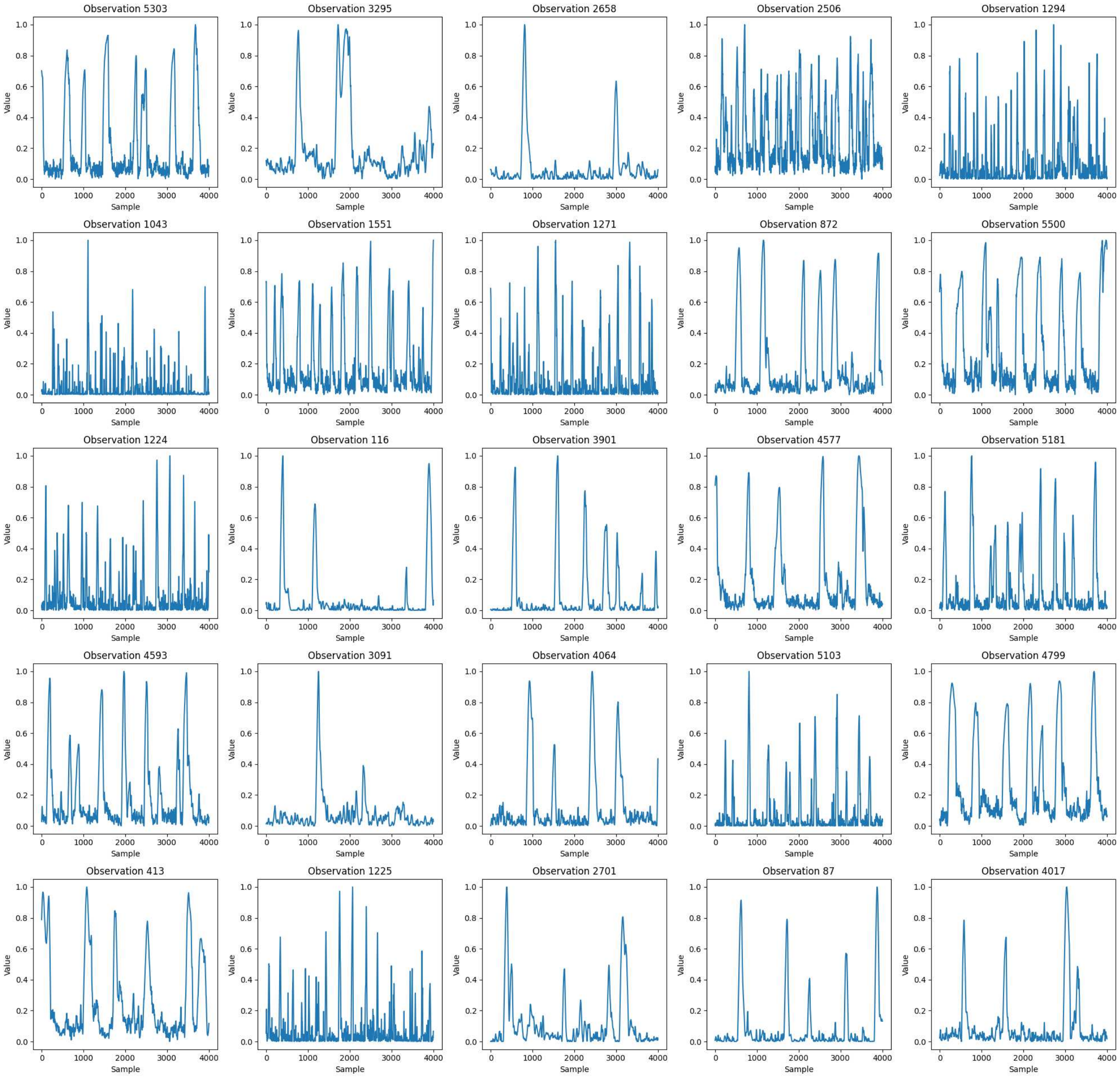


Figure 26 – Plot of randomly selected observations from time series of the Bubbly flow pattern in the TUD dataset.

chosen time series of the Bubbly pattern, Figure 27 shows 25-time series of the Churn pattern, and Figure 28 displays 25 time series of the Slug pattern for the TUD dataset. Observing Figures 26 and 28 reveals similarities between the time series of the Bubbly and Slug patterns, echoing the behavior observed in the HZDR data. This pattern of similarity likely contributes to misclassification in the same way, with certain operational points for Bubbly flow being classified as Slug and vice versa.

Since the TUD dataset contains a larger amount of data, this increased data volume positively impacted the performance of both classification approaches on this dataset. The greater data diversity allowed the models to better capture the nuances between patterns, potentially reducing the misclassification rate compared to the HZDR dataset.

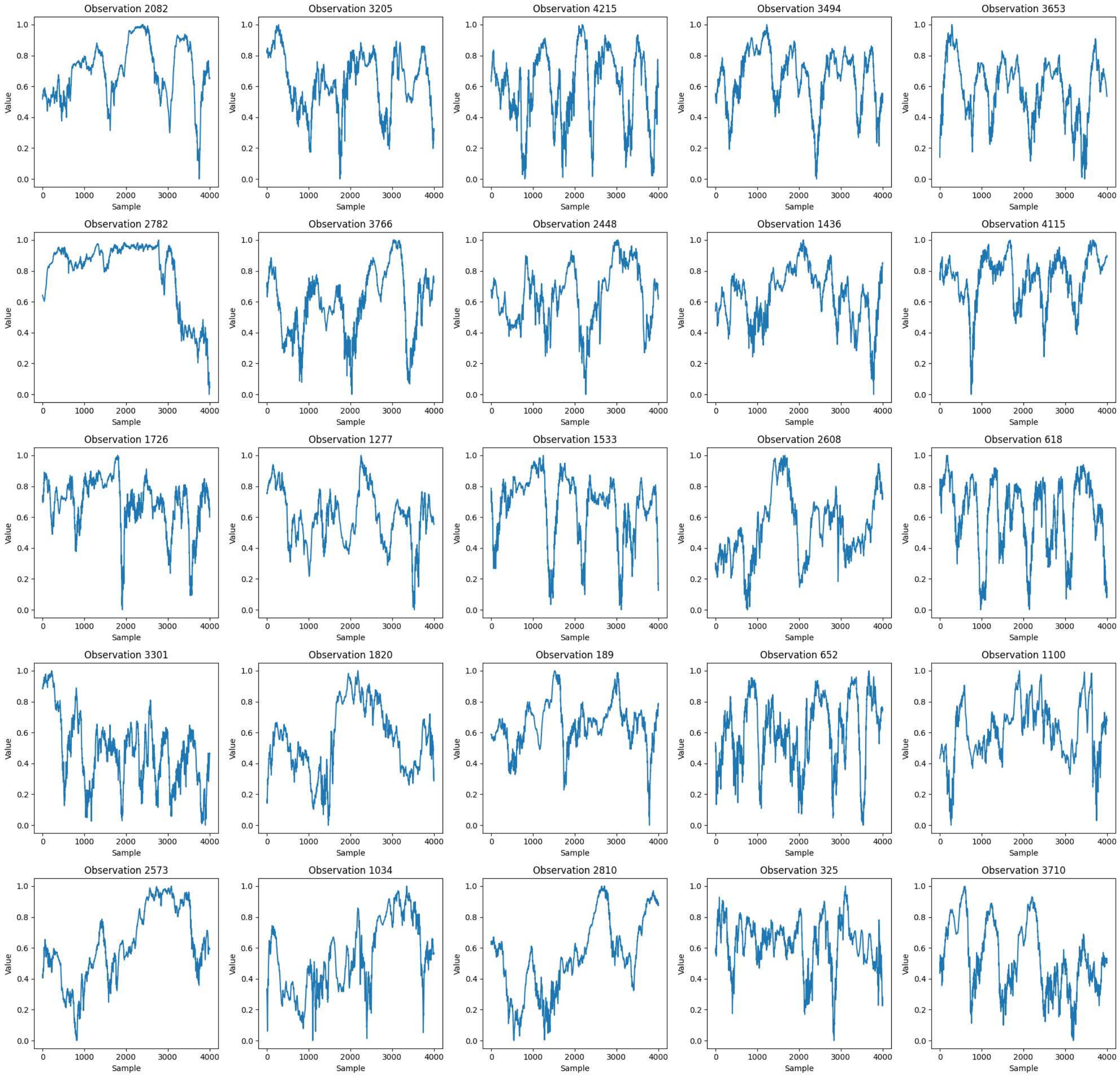


Figure 27 – Plot of randomly selected observations from time series of the Churn flow pattern in the TUD dataset.

Again, the time series for the Churn pattern, as shown in Figure 27, exhibit a distinct behavior that sets them apart from the Bubbly and Slug series in Figures 26 and 28. This distinct behavior of the Churn pattern likely contributes to its lower misclassification rates and reinforces the observation that Churn is more easily separable than the other two patterns.

4.3.3.2 Histogram and Gaussian Mixture Model

To gain deeper insights into the performance of the *MoG+SVM* approach, Figures 29 through 34 illustrate various aspects of the time series data for each flow pattern. In each figure, the first and fourth rows contain randomly selected time series samples, visually representing the

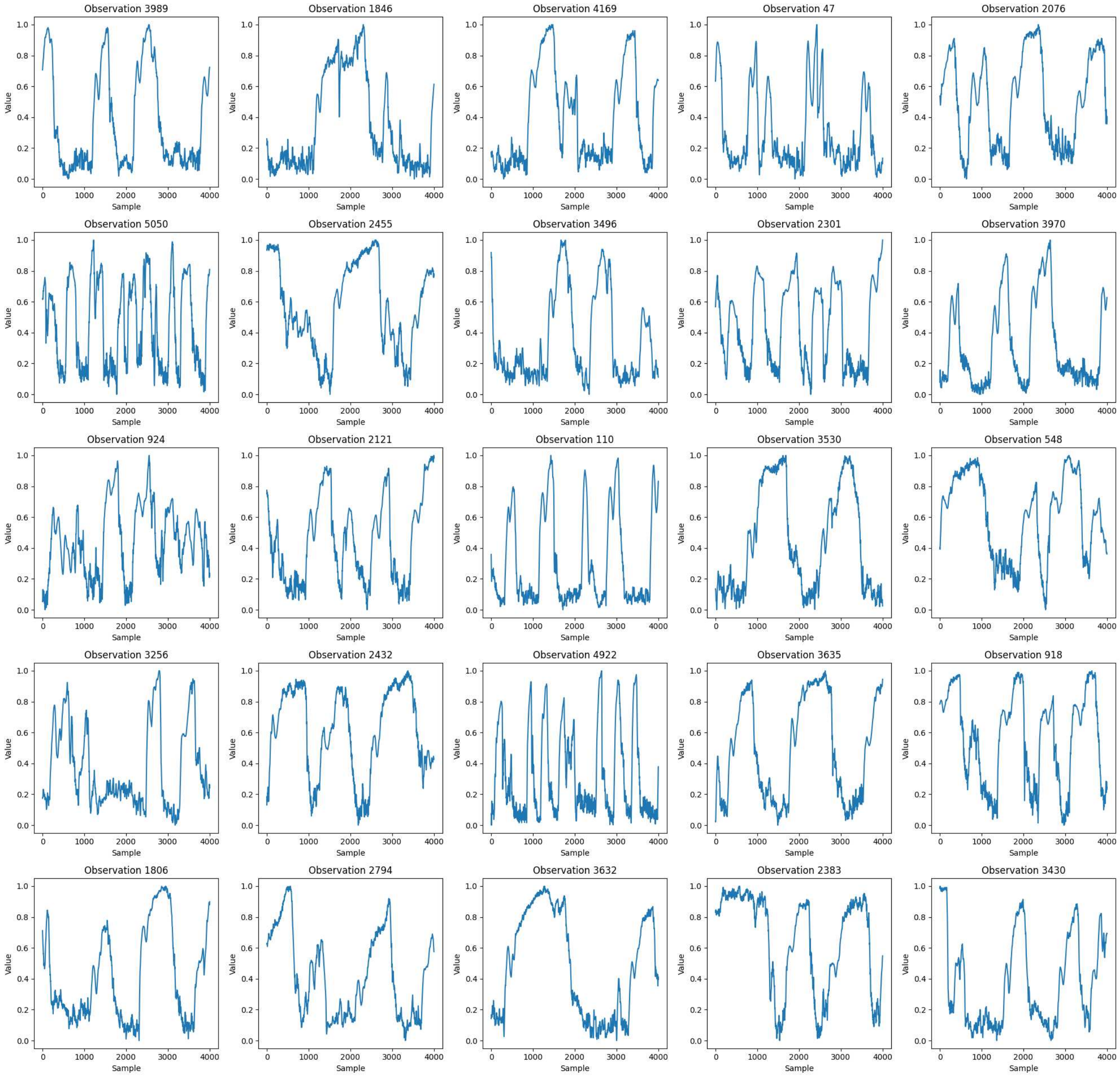
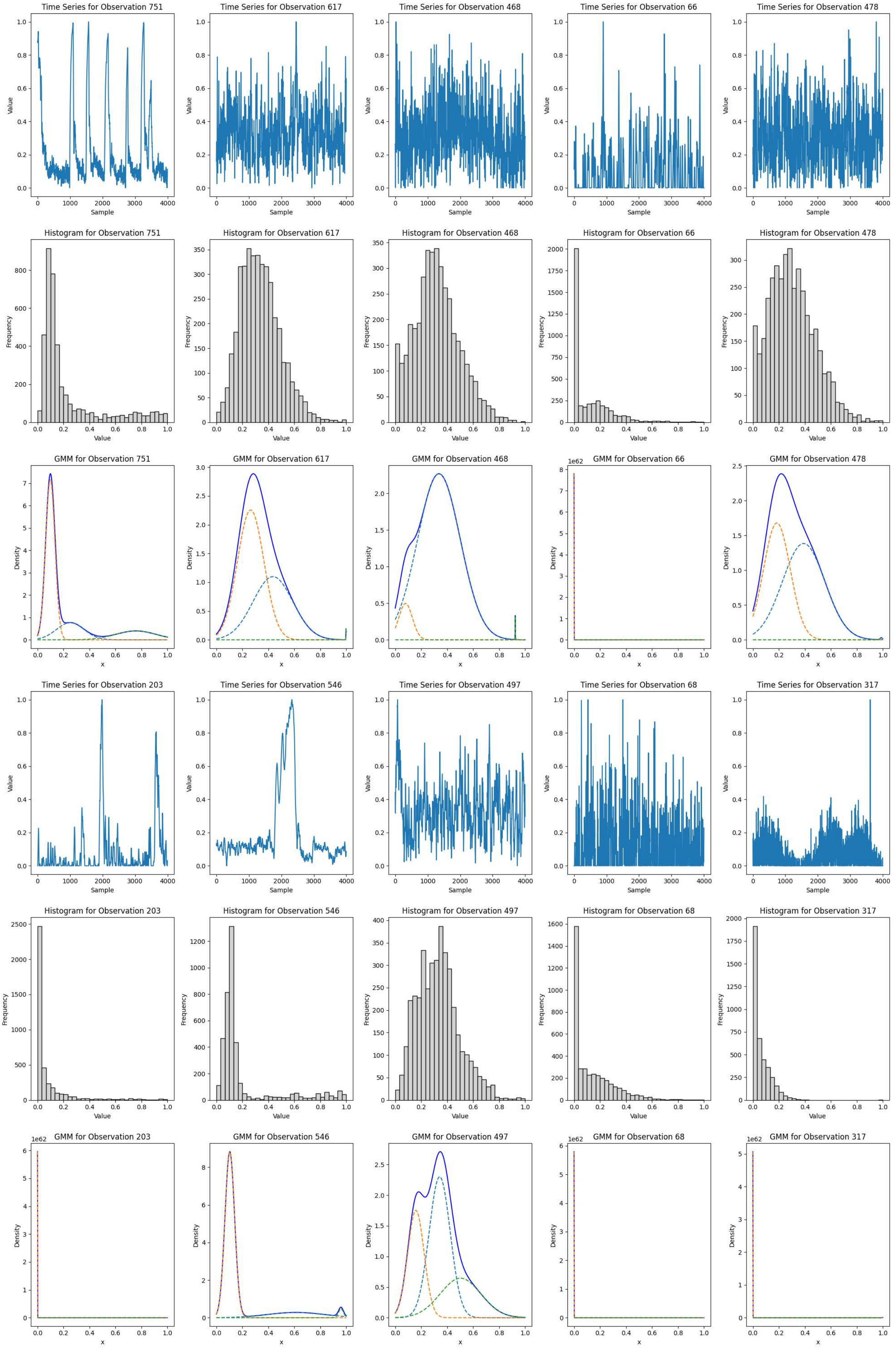


Figure 28 – Plot of randomly selected observations from time series of the Slug flow pattern in the TUD dataset.

raw data for each pattern. The second and fifth rows present the data's histograms, highlighting the distribution characteristics for each sample set. Finally, the third and sixth rows display the approximated PDF based on the parameters obtained from the Gaussian mixture model, offering a statistical approximation of the underlying data distribution. Figure 29 shows data from the Bubbly pattern of the HZDR dataset, Figure 30 presents data from the Churn pattern of the HZDR dataset, and Figure 31 displays data from the Slug pattern of the HZDR dataset. For the TUD dataset, Figure 32 contains data for the Bubbly pattern, Figure 33 for the Churn pattern, and Figure 34 for the Slug pattern. Together, these figures facilitate comparative analysis across different flow patterns and datasets, allowing for a closer examination of the *MoG+SVM* approach's ability to capture and differentiate the statistical characteristics of each flow regime.



Time Series for Observation 751

Time Series for Observation 617

Time Series for Observation 468

Time Series for Observation 66

Time Series for Observation 478

Histogram for Observation 751

Histogram for Observation 617

Histogram for Observation 468

Histogram for Observation 66

Histogram for Observation 478

GMM for Observation 751

GMM for Observation 617

GMM for Observation 468

GMM for Observation 66

GMM for Observation 478

Time Series for Observation 203

Time Series for Observation 546

Time Series for Observation 497

Time Series for Observation 68

Time Series for Observation 317

Histogram for Observation 203

Histogram for Observation 546

Histogram for Observation 497

Histogram for Observation 68

Histogram for Observation 317

GMM for Observation 203

GMM for Observation 546

GMM for Observation 497

GMM for Observation 68

GMM for Observation 317

Figure 29 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Bubbly flow pattern in the HZDR dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

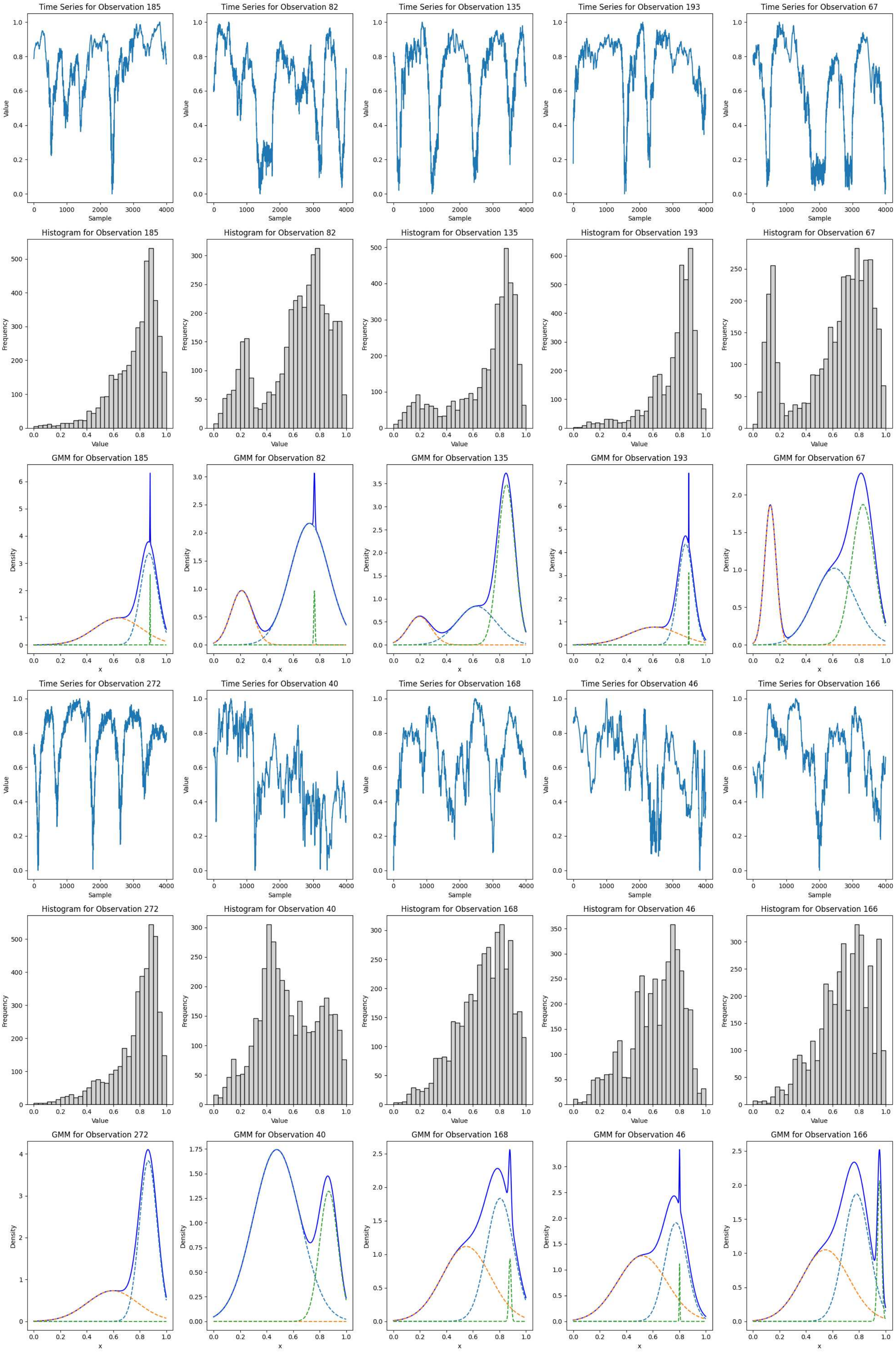


Figure 30 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Churn flow pattern in the HZDR dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

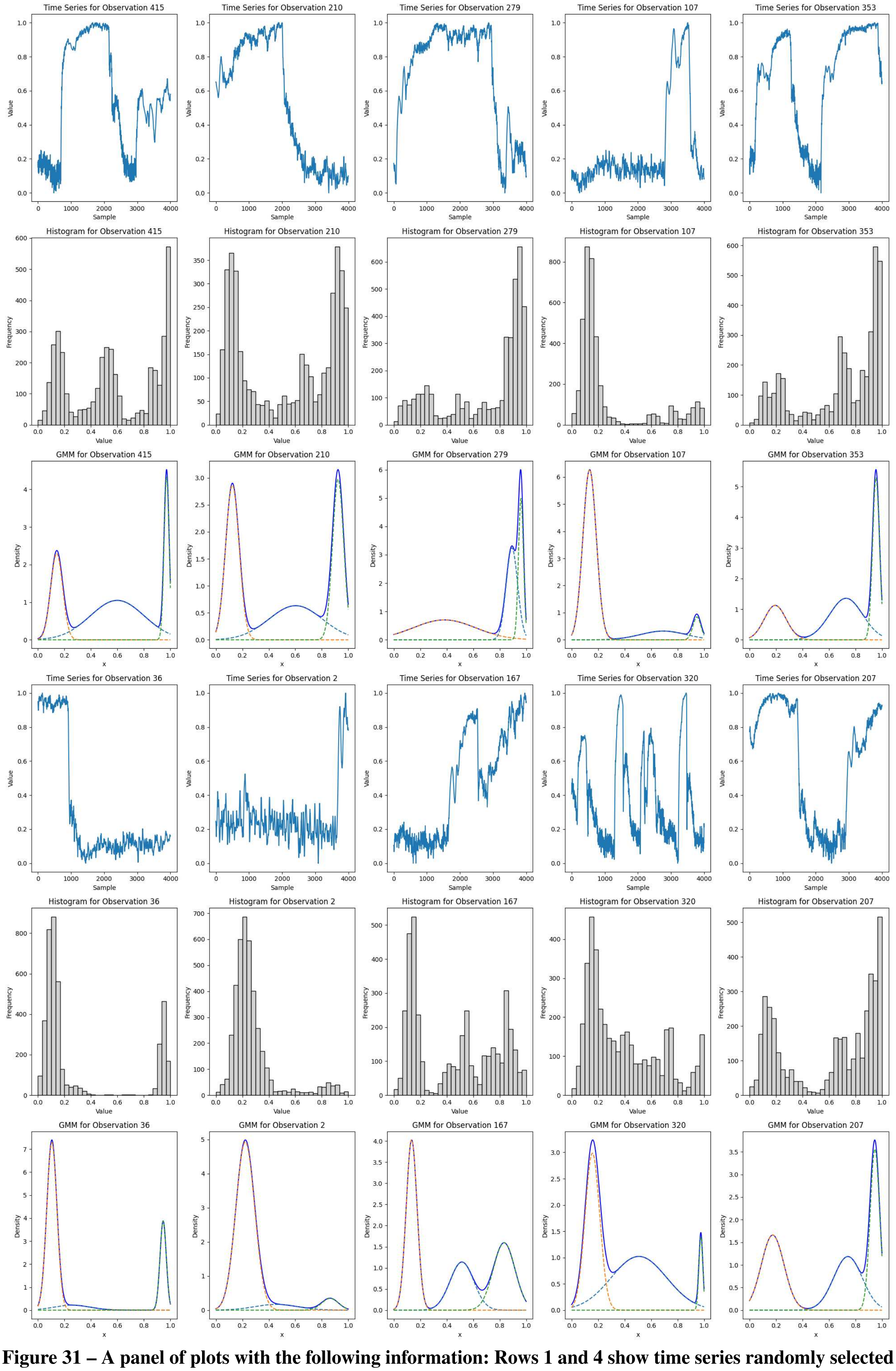


Figure 31 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Slug flow pattern in the HZDR dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

In Figure 29, representing the Bubbly pattern for the HZDR dataset, the Gaussian mixture model frequently generates right-skewed PDFs. Notably, for observations like 66 and 203, the model parameters produce identical PDFs where the estimated air presence is nearly null. This deviation from the actual data indicates limitations in capturing air-related dynamics in the time series. However, Figure 32, which shows data for the Bubbly pattern from the TUD dataset, does not exhibit this behavior, suggesting that the issue may be dataset-specific or influenced by data distribution within the HZDR dataset.

For the Churn pattern in the HZDR dataset, shown in Figure 30, the model primarily generates left-skewed multimodal PDFs, suggesting complex, overlapping flow characteristics within this pattern. This left-skewed, multimodal behavior appears less frequently in the Churn pattern data for the TUD dataset (Figure 33), indicating potential differences in flow characteristics or sensor responses between the two datasets.

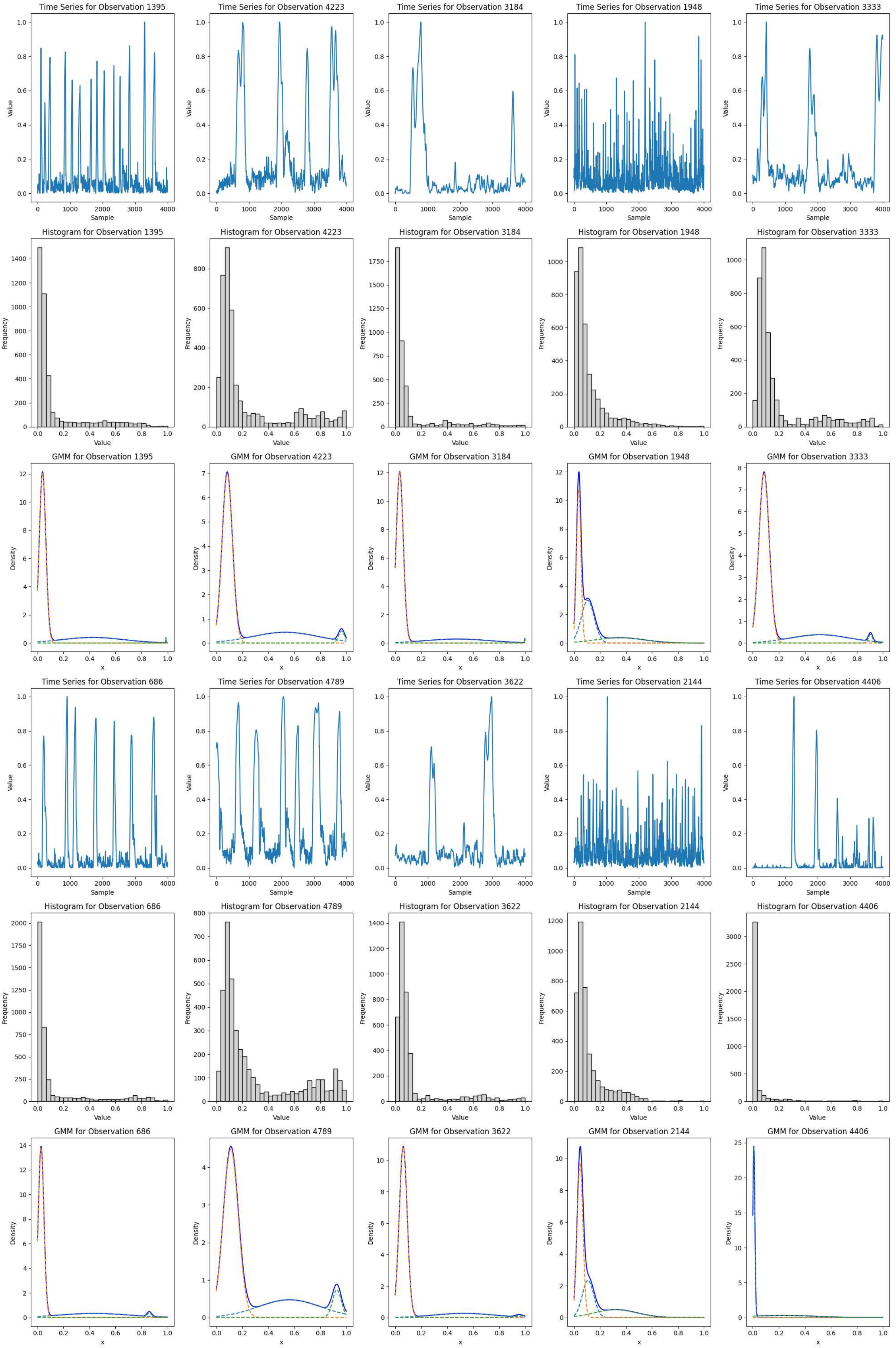
For the Slug pattern, both the HZDR dataset (Figure 31) and the TUD dataset (Figure 34) exhibit multimodal PDF distributions with characteristic modes at low and high void fractions. These modes likely represent distinct phases within the flow, where low void fractions correspond to liquid pistons and high void fractions indicate air slugs or Taylor bubbles. This consistent multimodal behavior across datasets underscores the model's capacity to capture the Slug pattern's defining characteristics effectively.

4.3.4 Cross-Datasets

To evaluate the model generalization, we conduct cross-dataset experiments. In these experiments, models trained on one dataset are tested on the other. The results are summarized in two parts: first, models trained on the HZDR dataset and tested on the TUD dataset, and second, models trained on the TUD dataset and tested on the HZDR dataset.

4.3.4.1 Train with HZDR dataset and test with TUD dataset

Table 12 shows the performance metrics for models trained on the HZDR dataset and tested on the TUD dataset. The results reveal a general decline in performance compared to single-dataset evaluations, highlighting the challenges of model generalization. ResNet exhibits a relatively high Accuracy of 92.17%, along with Precision, Recall, and F1-Score slightly above 92%, and a Balanced Accuracy of 91.97%. LSTM-FCN maintains solid performance as well,



Time Series for Observation 1395

Time Series for Observation 4223

Time Series for Observation 3184

Time Series for Observation 1948

Time Series for Observation 3333

Histogram for Observation 1395

Histogram for Observation 4223

Histogram for Observation 3184

Histogram for Observation 1948

Histogram for Observation 3333

GMM for Observation 1395

GMM for Observation 4223

GMM for Observation 3184

GMM for Observation 1948

GMM for Observation 3333

Time Series for Observation 686

Time Series for Observation 4789

Time Series for Observation 3622

Time Series for Observation 2144

Time Series for Observation 4406

Histogram for Observation 686

Histogram for Observation 4789

Histogram for Observation 3622

Histogram for Observation 2144

Histogram for Observation 4406

GMM for Observation 686

GMM for Observation 4789

GMM for Observation 3622

GMM for Observation 2144

GMM for Observation 4406

Figure 32 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Bubbly flow pattern in the TUD dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

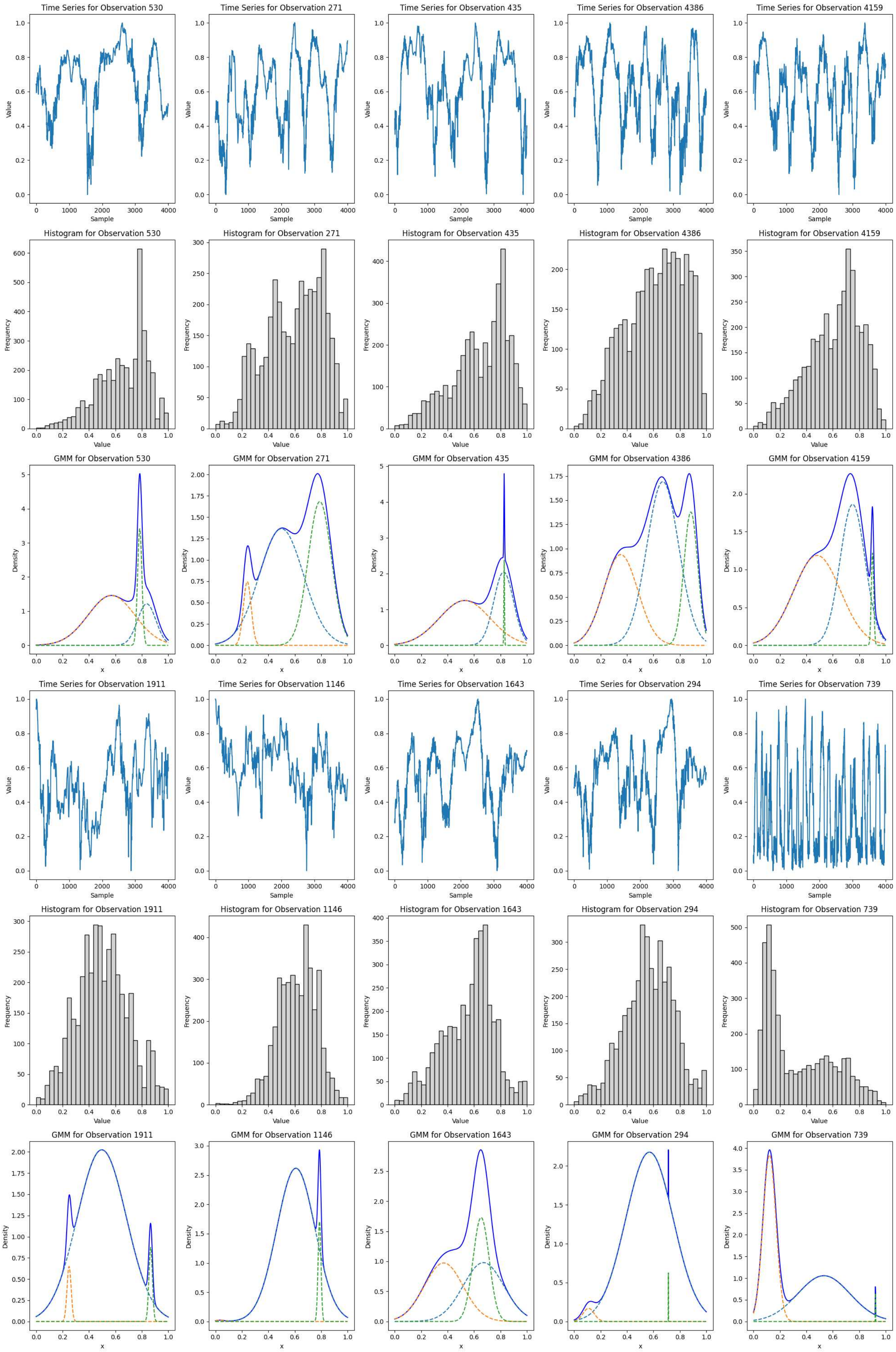


Figure 33 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Churn flow pattern in the TUD dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

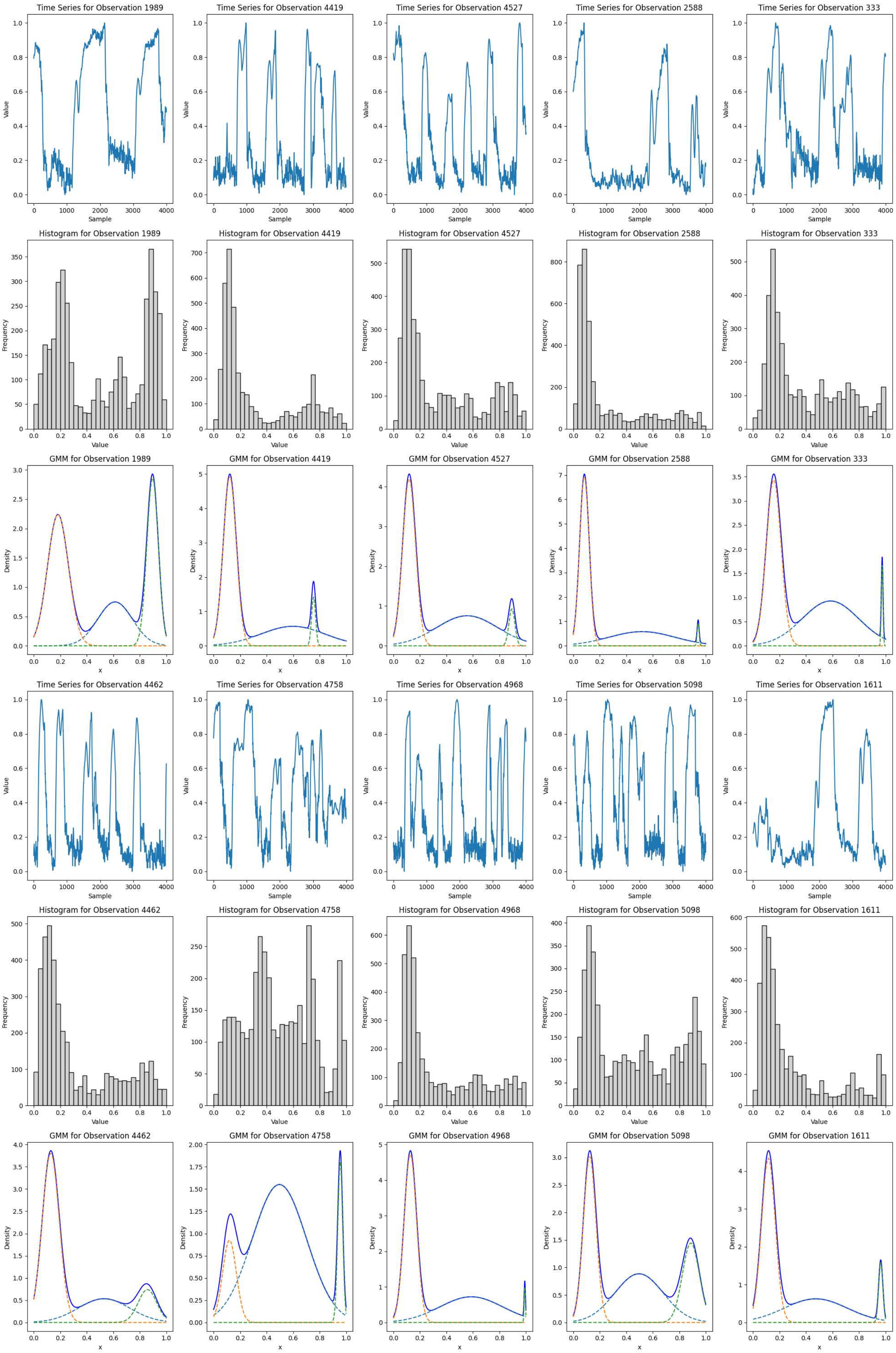


Figure 34 – A panel of plots with the following information: Rows 1 and 4 show time series randomly selected from the Slug flow pattern in the TUD dataset. Rows 2 and 5 display the corresponding histograms, while rows 3 and 6 present the approximated PDFs based on a Gaussian mixture model.

with an Accuracy of 91.42% and slightly higher Precision and F1-Score. TSTPlus shows a more significant degradation in performance with an Accuracy of 84.19%, indicating specific challenges in adapting to the TUD dataset. *MoG-SVM* performs better than TSTPlus, with an Accuracy of 87.50%, despite its generally lower performance in single-dataset evaluations.

Table 12 – Performance metrics for different models trained on the HZDR dataset and tested on the TUD dataset.

Model	Accuracy	Precision	Recall	F1-Score	Balanced Acc.
ResNet	0.9217	0.9268	0.9217	0.9223	0.9197
LSTM-FCN	0.9142	0.9234	0.9142	0.9152	0.9112
TSTPlus	0.8419	0.8619	0.8419	0.8434	0.8453
MoG-SVM	0.8750	0.8813	0.8750	0.8760	0.8741

Figure 35 provides the confusion matrices for models trained on the HZDR dataset and tested with the TUD dataset. ResNet (Figure 35(a)) shows impressive specificity with Slug but experiences some classification challenges with Churn. LSTM-FCN (Figure 35(b)) also performs well, particularly in distinguishing Slug, but like ResNet, it misclassifies a notable number of Churn instances. TSTPlus (Figure 35(c)) and *MoG-SVM* (Figure 35(d)) both display increased errors across all categories, especially in misclassifying the Bubbly and Churn patterns, which significantly impacts their performance metrics.

When compared to the single-dataset results where models showed higheAccuracycy and more stable metrics (e.g., ResNet and LSTM-FCN both achieving over 98% in HZDR and TUD datasets), the cross-dataset results underscore the challenges in generalization. Notably, while *MoG-SVM* generally underperforms in single-dataset scenarios, it shows comparative resilience in cross-dataset testing, particularly outperforming TSTPlus accuracy and other metrics.

4.3.4.2 Train with TUD dataset and test with HZDR dataset

Table 13 presents the performance metrics for models trained on the TUD dataset and tested on the HZDR dataset. Contrary to earlier evaluations, TSTPlus emerges as the top performer, with an Accuracy of 86.23% and nearly consistent scores across Precision, Recall, and F1-Score, showing strong generalization. ResNet, while maintaining reasonable metrics, sees a significant drop in performance compared to its single-dataset evaluations, with an Accuracy of 75.37%. LSTM-FCN displays slightly higher betteAccuracycy at 77.07%, but still below TSTPlus. *MoG-SVM* again shows the lowest performance, particularly struggling with generalization, as evidenced by its Accuracy of 70.42% and the lowest Balanced Accuracy.

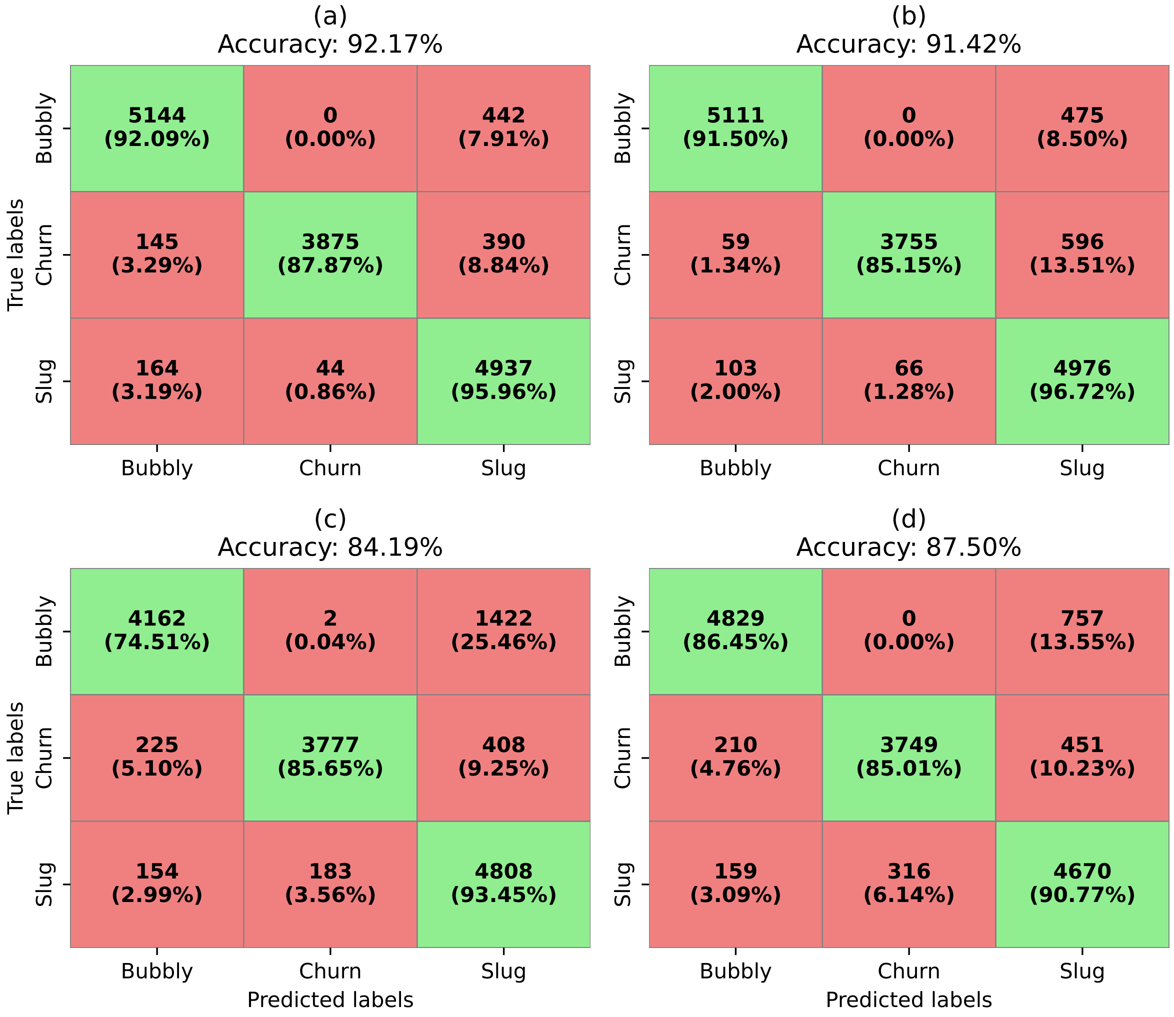


Figure 35 – Confusion matrices for models trained with the HZDR dataset and tested with the TUD dataset. (a) ResNet. (b) LSTM-FCN. (c) TSTPlus. (d) *MoG-SVM*

Table 13 – Performance metrics for different models trained on the TUD dataset and tested on the HZDR dataset.

Model	Accuracy	Precision	Recall	F1-Score	Balanced Acc.
ResNet	0.7537	0.8536	0.7537	0.7682	0.8144
LSTM-FCN	0.7707	0.8525	0.7707	0.7837	0.8077
TSTPlus	0.8623	0.8698	0.8623	0.8636	0.8584
MoG-SVM	0.7042	0.7502	0.7042	0.7060	0.7502

Figure 36 illustrates the confusion matrices for models trained with the TUD dataset and tested with the HZDR dataset. TSTPlus (Figure 36(c)) shows the best performance in correctly classifying the majority of instances across all classes with fewer misclassifications compared to other models. ResNet (Figure 36(a)) and LSTM-FCN (Figure 36(b)) exhibit some difficulty, reflecting their lower accuracy metrics. *MoG-SVM* (Figure 36(d)) struggles significantly across all categories, particularly with a high misclassification rate in the Bubbly category, which greatly affects its overall performance.

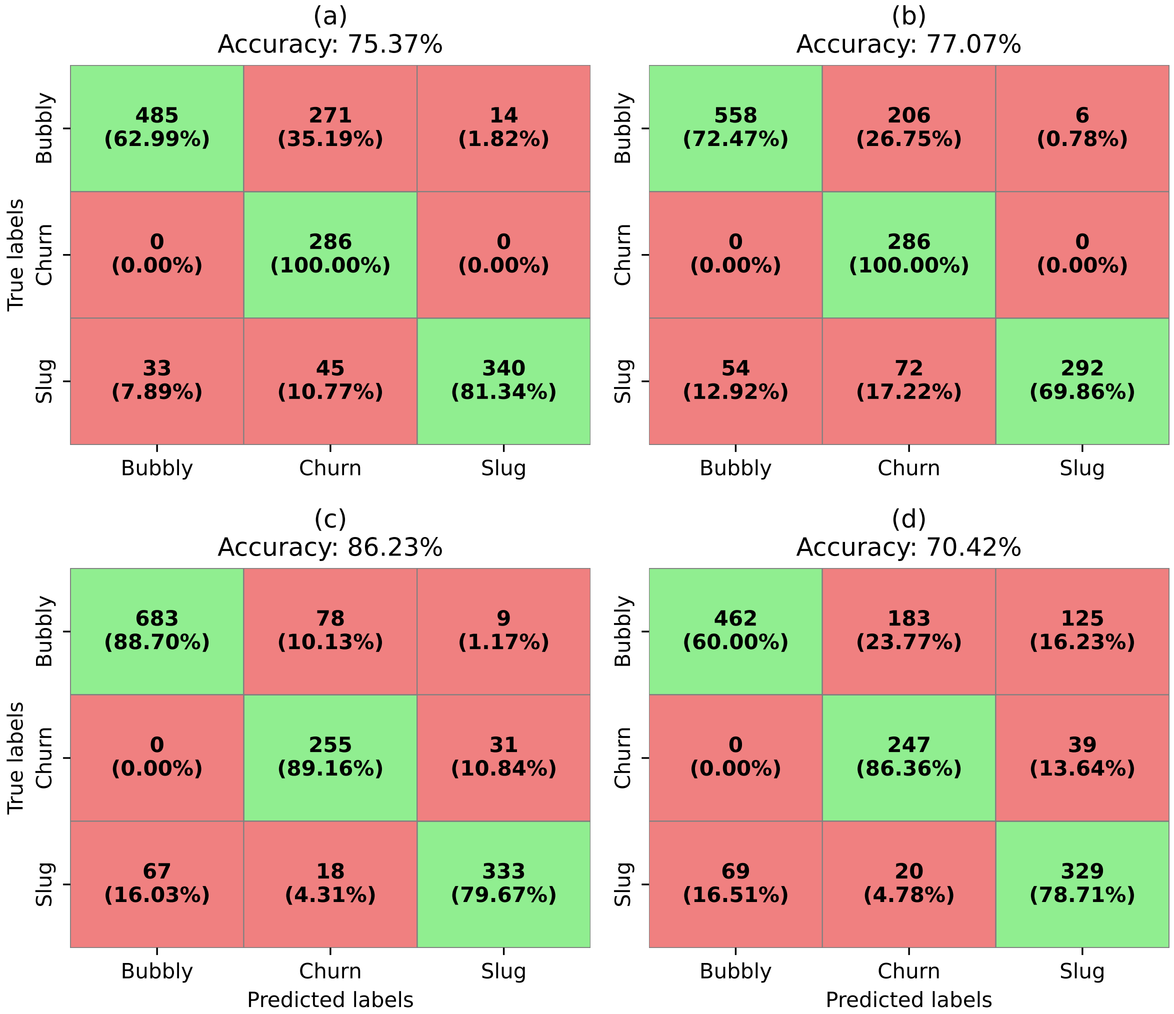


Figure 36 – Confusion matrices for models trained with the TUD dataset and tested with the HZDR dataset. (a) ResNet. (b) LSTM-FCN. (c) TSTPlus. (d) *MoG-SVM*

This section contrasts sharply with the results observed when models were trained on the HZDR dataset and tested on the TUD dataset. TSTPlus, which had performed worse in the previous setting, shows adaptability and generalization when trained on TUD and tested on HZDR. This inversion highlights the model's sensitivity to training conditions and dataset characteristics. ResNet and LSTM-FCN, while generally robust, exhibit a decline in performance, suggesting their potential overfitting to the TUD dataset characteristics or an inability to generalize as effectively to the HZDR dataset's conditions. *MoG-SVM*'s consistently lower performance in cross-dataset evaluations further underscores its challenges with generalization compared to other models.

This detailed analysis reveals the intricate balance between high performance in familiar settings versus robustness in diverse scenarios, emphasizing the importance of evaluating models across different datasets to fully understand their generalization capabilities and reinforcing the need for end-to-end models for this task. Using handcrafted features, the *MoG-SVM*-based model can become specialized for one dataset, reducing performance in others and indicating possible overfitting for this strategy. This characteristic may make practical applications.

4.3.5 Qualitative Analysis

The study of flow classification across different datasets revealed distinct challenges, especially in classifying the Slug and Bubbly flow patterns. A significant number of misclassifications were observed between these two classes across all models. When analyzing the time series for both classes from the HZDR and TUD datasets, a noticeable pattern emerged that could help explain the high misclassification rate, as shown in Fig. 37 for the HZDR dataset and Fig. 38 for the TUD dataset. The similarities between Bubbly and Slug patterns were evident in both datasets, with cases from the HZDR dataset (Fig. 37) and the TUD dataset (Fig. 38) displaying almost indistinguishable characteristics. The distinction between these two flow patterns becomes even less obvious when the analysis window is adjusted, especially for the TUD dataset, as observed in Fig. 39, where a cross-dataset comparison further highlights the overlap in flow characteristics.

A deeper analysis of the misclassifications suggests that the transition region between Bubbly and Slug patterns might be critical for real-world applications. Such regions may warrant the inclusion of transition classes for better evaluation. Despite the misclassifications, models demonstrated overall strong performance, with metrics exceeding 90% for most of the flow patterns. The misclassification patterns, particularly in the transition zones, are represented in the figures where the flow regions are categorized according to mechanistic modeling (TAITEL *et al.*, 1980). Figs. 40 and 41 show the misclassified instances for the HZDR and TUD datasets, with the misclassification regions highlighted and zoomed in for better visualization.

In the cross-dataset evaluation, models trained on the HZDR dataset and tested on the TUD dataset showed varying performance, particularly with the Slug class achieving the highest accuracy. The Churn class, however, exhibited substantial misclassification across three of the four models, suggesting that the Churn pattern contains features that do not generalize well across datasets. On the other hand, models trained on the TUD dataset and tested on the HZDR dataset experienced a sharp drop in performance across all models except for the TSTPlus model, which exhibited more stable results. Despite this, a large portion of Bubbly data was misclassified as Churn, highlighting the overlapping features that make class differentiation challenging.

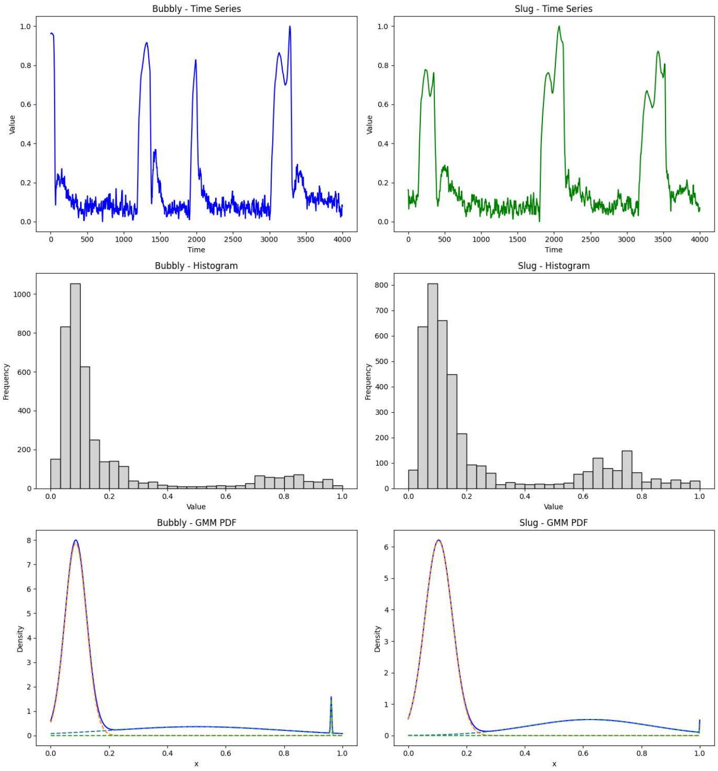


Figure 37 – Comparison between two time-series from the HZDR dataset with different flow patterns: The first column displays a time series with the Bubbly flow pattern, its histogram, and the approximated PDF from a Gaussian mixture model. The second column presents the same information for a time series with the Slug flow pattern.

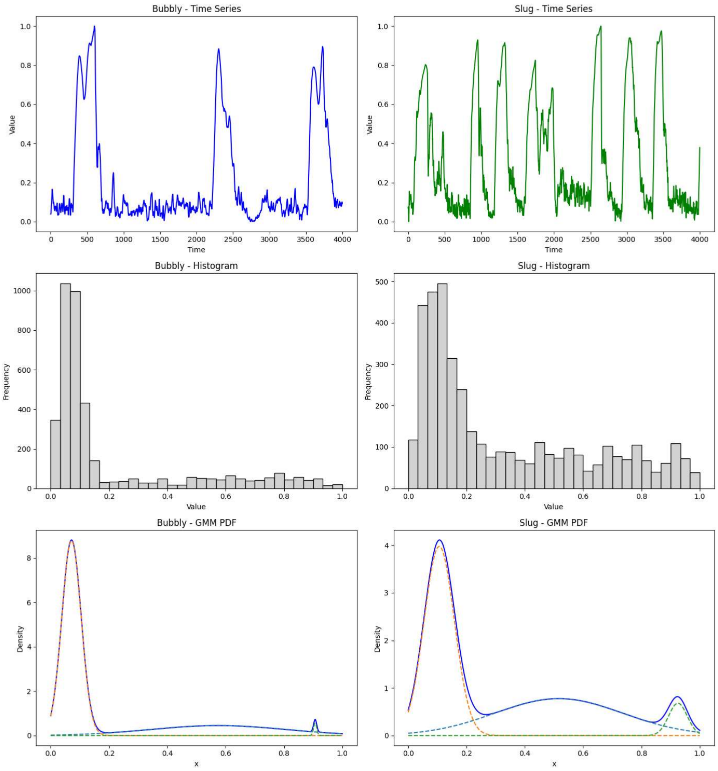


Figure 38 – Comparison between two time series from the TUD dataset with different flow patterns: The first column displays a time series with the Bubbly flow pattern, its histogram, and the approximated PDF from a Gaussian mixture model. The second column presents the same information for a time series with the Slug flow pattern.

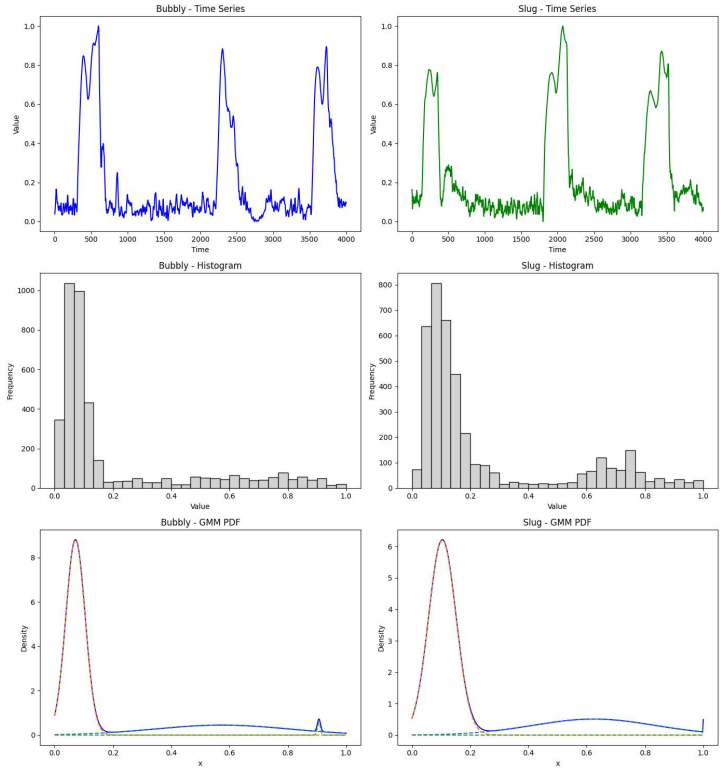


Figure 39 – Comparison between two time series from different datasets and flow patterns: The first column shows a time series with the Bubbly flow pattern from the TUD dataset, along with its histogram and the approximated PDF from a Gaussian mixture model. The second column presents the same information for a time series with the Slug flow pattern from the HZDR dataset.

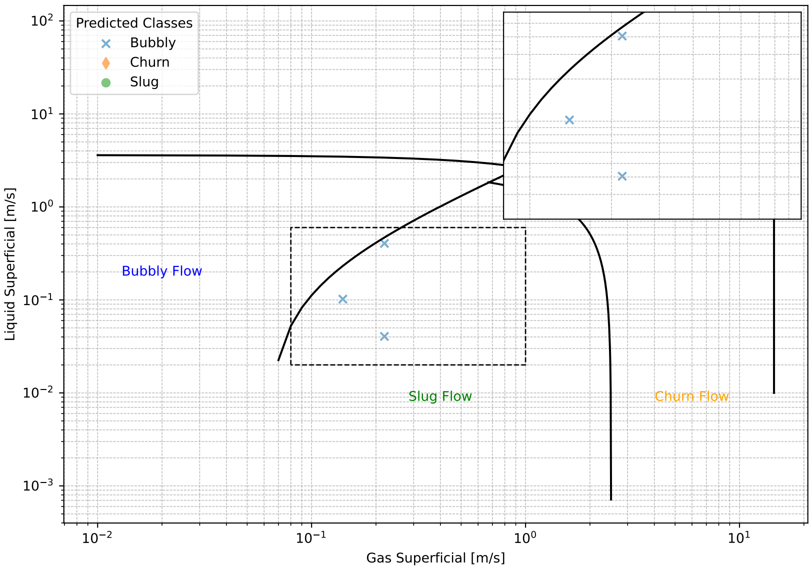


Figure 40 – HZDR data misclassified according to the map defined in (TAITEL *et al.*, 1980). In the upper right corner, a detail of the error region is shown.

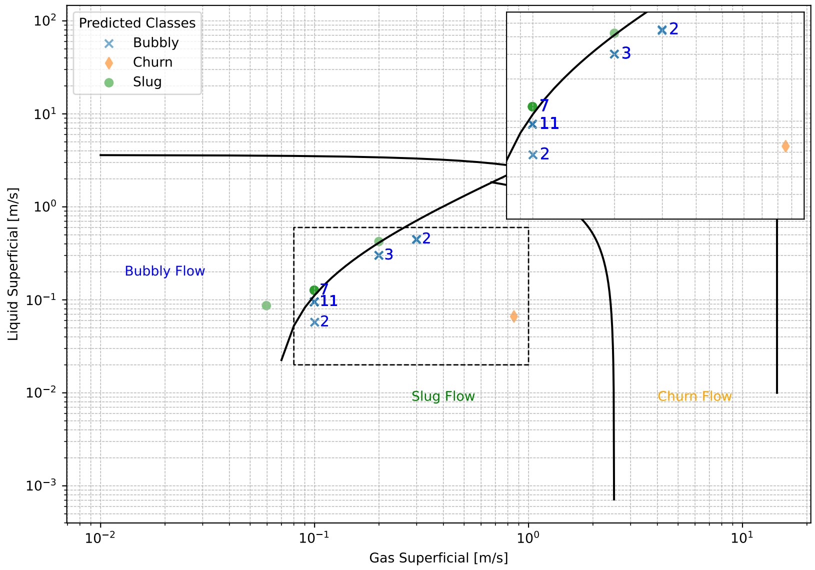


Figure 41 – TUD data misclassified according to the map defined in (TAITEL *et al.*, 1980). In the upper right corner, a detail of the error region is shown.

These findings are corroborated by the analysis of Figures 37 through 39, which show time series, histograms, and Gaussian Mixture Model (GMM) PDFs for representative examples from both datasets. Figure 37 compares Bubbly and Slug flow patterns in the HZDR dataset, revealing similarities in their statistical features. Similarly, Fig. 38 highlights similar characteristics between Bubbly and Slug patterns in the TUD dataset. Figure 39 illustrates a comparison between the TUD Bubbly and HZDR Slug data, emphasizing the overlap in flow characteristics that further complicates accurate classification across datasets.

5 CONCLUSIONS AND FUTURE WORKS

This work proposed and evaluated two distinct approaches for classifying two-phase flow patterns using void fraction data from wire-mesh sensors: the traditional *MoG+SVM* approach, which combines a Gaussian Mixture Model (MoG) for feature extraction with Support Vector Machines (SVM) for classification, and the more modern *SOTA-DL* (State-of-the-Art Deep Learning) approach, which leverages advanced time-series classification models including ResNet, LSTM-FCN, and TSTPlus. The main objectives of this thesis were to compare the performance of these methods, focusing mainly on their ability to generalize across different datasets in cross-dataset experiments.

5.1 SUMMARY OF KEY FINDINGS

In the first approach, *MoG+SVM*, we proposed a combination of feature extraction using the Gaussian Mixture Model with the supervised learning algorithm Support Vector Machine. Various kernel functions (linear, polynomial, and Radial Basis Function) were tested, and the optimal parameters were selected using nested cross-validation. The results showed that the *MoG+SVM* model achieved excellent performance with an average accuracy of 95% when evaluated on the HZDR dataset and 95,4% when evaluated on the TUD dataset. The method also proved to be effective in terms of interpretability, offering a direct relationship with the physical phenomenon, as described in Jones and Zuber (1975) and Costigan and Whalley (1997). This approach demonstrated that feature engineering using physically meaningful parameters can yield competitive results, particularly when dataset characteristics are well understood.

In contrast, the *SOTA-DL* approach, which utilized deep learning models like ResNet, LSTM-FCN, and TSTPlus, provided an end-to-end solution for flow regime classification, moving beyond the need for hand-crafted features. The results revealed that the deep learning models, particularly ResNet and LSTM-FCN, achieved higher accuracy and F1-scores than the *MoG+SVM* on single-dataset evaluations, with ResNet showing superior performance, surpassing 98% accuracy on both the HZDR and TUD datasets. However, a key limitation of these models was their performance degradation in cross-dataset scenarios. While ResNet and LSTM-FCN showed good performance on single-dataset evaluation but worse performance considering both cross-dataset tests, TSTPlus, despite its lower performance in single-dataset evaluations, showed better resilience across datasets, achieving a cross-dataset accuracy of 86.23% when trained on TUD and tested on HZDR, the best performance on this case.

These findings underline the critical importance of model generalization, particularly in real-world applications where data from different acquisition systems may vary. While the SOTA methods are robust within a specific dataset, their performance can be sensitive to shifts in data distribution, especially when models are exposed to data collected with different instrumentation or under varying operational conditions. This highlights the need for future research to explore ways to improve model robustness and generalization capabilities.

5.2 CONTRIBUTIONS AND ACHIEVEMENTS

This thesis has made several key contributions to the field of two-phase flow pattern classification:

1. **Development of a Hand-Crafted Approach:** The *MoG+SVM* method, as presented in Section 3.4, showed competitive results with an average accuracy exceeding 95% on the HZDR dataset. This approach emphasizes the value of using interpretable features derived from physical models, thus contributing a unique perspective in flow regime classification.

2. **Implementation of End-to-End Deep Learning Models:** The *SOTA-DL* approach, detailed in Section 3.5, introduced modern deep learning architectures like ResNet, LSTM-FCN, and TSTPlus, which allowed for direct classification from raw time-series data. This method demonstrated the potential of deep learning models in overcoming the limitations of hand-crafted features and setting new benchmarks in flow regime classification.

3. **Evaluation and Dataset Organization:** Two distinct datasets, HZDR and TUD, were utilized in this study. Sections 3.2 and 3.3 provided a comprehensive overview of the datasets and the necessary preprocessing steps for their use in flow regime classification. The careful organization of these datasets facilitated rigorous experiments and meaningful comparisons across methods.

4. **Single-Dataset and Cross-Dataset Performance Evaluation:** A comprehensive evaluation of the performance of both approaches on single datasets and in cross-dataset scenarios was conducted in Section 4.3. This comparison revealed the strengths and weaknesses

of each method, with the SOTA-DL models outperforming *MoG+SVM* in single-dataset scenarios but demonstrating a need for improvement in generalization across datasets.

5. **Comparative Analysis of Approaches:** Section 4.3 also presented a detailed quantitative and qualitative comparison of the proposed methods, shedding light on their relative advantages, such as the high interpretability of *MoG+SVM* and the scalability of the deep learning models.

5.3 CONCLUSIONS

In conclusion, this thesis advanced state of the art in two-phase flow classification by comparing two fundamentally different approaches: the *MoG+SVM* method, based on handcrafted features, and the *SOTA-DL* method, based on deep learning models. The results demonstrate that while deep learning models such as ResNet and LSTM-FCN show remarkable performance on single datasets, their generalization across datasets remains a challenge. On the other hand, the *MoG+SVM* method, although not as powerful in terms of accuracy, provides a more interpretable approach that leverages physical features related to the flow phenomenon, offering a unique advantage in terms of transparency and application in scenarios where explainability is crucial.

This study highlights the trade-offs between accuracy and interpretability in the context of two-phase flow pattern recognition. While SOTA methods offer significant improvements in classification performance, their deployment in real-world applications requires careful consideration of their generalization capabilities. Future work should focus on overcoming these limitations by enhancing the generalization capacity of deep learning models, improving the interpretability of complex models, and exploring hybrid approaches that combine the strengths of both deep learning and traditional machine learning techniques.

5.4 FUTURE WORKS

Several avenues for future research are identified based on the findings of this thesis:

1. **Improving Generalization:** Future work should focus on enhancing the generalization capabilities of deep learning models through more sophisticated training techniques, regularization methods, or hybrid model architectures that combine the strengths of traditional

and deep learning models.

2. **Computational Complexity and Embedded Systems:** Analyzing the computational complexity of the proposed models is crucial for their application in embedded systems. Future research could focus on optimizing these models for real-time deployment in industrial environments.

3. **Model Interpretability:** Further investigation into the interpretability of deep learning models is needed. Research could explore methods for improving the transparency of these models, such as by examining the importance of specific layers or operations in the decision-making process.

4. **Multimodal Models and Sensor Fusion:** The possibility of multimodal models or sensor fusion, incorporating additional data sources (e.g., visual or acoustic sensors), could significantly enhance the robustness and accuracy of flow regime classification systems.

5. **Expanding the Dataset and Classifying Additional Flow Regimes:** The models developed in this work should be tested in a broader range of datasets, including datasets from different operational conditions, and further refined to handle additional flow regimes, particularly transition states between flow patterns.

6. **Hybrid Approaches for Real-Time Applications:** Future studies could explore hybrid approaches that combine the strengths of both hand-crafted features and deep learning methods, optimizing for real-time classification and dynamic flow conditions.

By addressing these challenges, future work can significantly advance the application of machine learning and deep learning techniques in two-phase flow classification, improving the accuracy, robustness, and interpretability of models in industrial applications such as oil and gas, nuclear reactors, and other multi-phase flow systems.

REFERENCES

ABADI, Martín; AGARWAL, Ashish; BARHAM, Paul; BREVDO, Eugene; CHEN, Zhifeng; CITRO, Craig; CORRADO, Greg S.; DAVIS, Andy; DEAN, Jeffrey; DEVIN, Matthieu; GHEMAWAT, Sanjay; GOODFELLOW, Ian; HARP, Andrew; IRVING, Geoffrey; ISARD, Michael; JIA, Yangqing; JOZEFOWICZ, Rafal; KAISER, Lukasz; KUDLUR, Manjunath; LEVENBERG, Josh; MANé, Dandelion; MONGA, Rajat; MOORE, Sherry; MURRAY, Derek; OLAH, Chris; SCHUSTER, Mike; SHLENS, Jonathon; STEINER, Benoit; SUTSKEVER, Ilya; TALWAR, Kunal; TUCKER, Paul; VANHOUCKE, Vincent; VASUDEVAN, Vijay; VIéGAS, Fernanda; VINYALS, Oriol; WARDEN, Pete; WATTENBERG, Martin; WICKE, Martin; YU, Yuan; ZHENG, Xiaoqiang. **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. 2015. Software available from tensorflow.org. Available at: https://www.tensorflow.org/.

AL-NASER, Mustafa; ELSHAFEI, Moustafa; AL-SARKHI, Abdelsalam. Artificial neural network application for multiphase flow patterns detection: A new approach. **Journal of Petroleum Science and Engineering**, Elsevier, v. 145, p. 548–564, 2016.

AMBROSIO, Jefferson dos Santos. **Classificação de padrões de escoamento bifásico líquido-gás baseada em séries temporais de fração de vazio e técnicas de aprendizagem de máquina**. 2019. Master's Thesis (Master's Thesis) — Universidade Tecnológica Federal do Paraná, 2019.

Ambrosio, Jefferson dos Santos; LAZZARETTI, André Eugenio; PIPA, Daniel Rodrigues; da Silva, Marco Jose. Two-phase flow pattern classification based on void fraction time series and machine learning. **Flow Measurement and Instrumentation**, v. 83, p. 102084, 2022. ISSN 0955-5986. Available at: https://www.sciencedirect.com/science/article/pii/S0955598621001849.

AZZOPARDI, Barry; HILLS, John. Flow patterns, transitions and models for specific flow patterns. *In*: BERTOLA, Volfango (Ed.). **Modelling and experimentation in Two-phase flow**. Vienna: Springer, 2003. p. 1–77. ISBN 978-3-7091-2538-0.

BAHRAMI, Siamak; ALAMDARI, Saeid; FARAJMASHAEI, Mohammadreza; BEHBAHANI, Mohammad; JAMSHIDI, Saeid; BAHRAMI, Babak. Application of artificial neural network to multiphase flow metering: A review. **Flow Measurement and Instrumentation**, v. 97, p. 102601, 2024. ISSN 0955-5986. Available at: https://www.sciencedirect.com/science/article/pii/S0955598624000815.

BANOWSKI, Manuel; BEYER, Matthias; SZALINSKI, Lutz; LUCAS, Dirk; HAMPEL, Uwe. Comparative study of ultrafast x-ray tomography and wire-mesh sensors for vertical gas–liquid pipe flows. **Flow Measurement and Instrumentation**, Elsevier, v. 53, p. 95–106, 2017.

BAO, Minghan; WANG, Mi; LI, Kang; JIA, Xiaodong. Integrating machine learning with sensor technology for multiphase flow measurement: A review. **IEEE Sensors Journal**, v. 24, n. 19, p. 29603–29618, 2024.

BENITO, Ines; MUREITHI, Njuki W. Identification of two-phase flow patterns using support vector classification. *In*: AMERICAN SOCIETY OF MECHANICAL ENGINEERS. **Pressure Vessels and Piping Conference**. *[S.l.]*, 2017. v. 57977, p. V004T04A027.

BERTOLA, Volfango. **Modelling and experimentation in two-phase flow**. *[S.l.]*: Springer, 2014. v. 450.

BILMES, Jeff A *et al.* A gentle tutorial of the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models. **International Computer Science Institute**, v. 4, n. 510, p. 126, 1998.

BISHOP, Christopher M. **Pattern Recognition and Machine Learning**. 1st ed. 2006. corr. 2nd printing. ed. *[S.l.]*: Springer, 2006. (Information science and statistics). ISBN 9780387310732,0387310738.

BOLóN-CANEDO, V.; ATAER-CANSIZOGLU, E.; ERDOGMUS, D.; KALPATHY-CRAMER, J.; CHIANG, M.F. A gmm-based feature extraction technique for the automated diagnosis of retinopathy of prematurity. *In*: **2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)**. *[S.l.: s.n.]*, 2015. p. 1498–1501.

BREITENMOSER, David; PRASSER, Horst-Michael; MANERA, Annalisa; PETROV, Victor. Machine learning based flow regime recognition in helically coiled tubes using x-ray radiography. **International Journal of Multiphase Flow**, v. 161, p. 104382, 2023. ISSN 0301-9322. Available at: https://www.sciencedirect.com/science/article/pii/S0301932223000058.

CAWLEY, Gavin C.; TALBOT, Nicola L.C. On over-fitting in model selection and subsequent selection bias in performance evaluation. **J. Mach. Learn. Res.**, JMLR.org, v. 11, p. 2079–2107, Aug. 2010. ISSN 1532-4435.

CIESLAK, Dionatan A. G.; MORETO, Miguel; LAZZARETTI, André E.; MACEDO-JúNIOR, José R. High impedance fault classification in microgrids using a transformer-based model with time series harmonic synchrophasors under data quality issues. **Neural Comput. Appl.**, Springer-Verlag, Berlin, Heidelberg, v. 36, n. 23, p. 14017–14034, May 2024. ISSN 0941-0643. Available at: https://doi.org/10.1007/s00521-024-09802-y.

COSTIGAN, G; WHALLEY, PB. Slug flow regime identification from dynamic void fraction measurements in vertical air-water flows. **International Journal of Multiphase Flow**, Elsevier, v. 23, n. 2, p. 263–282, 1997.

EYO, Edem Nsefik; PILARIO, Karl Ezra Salgado; LAO, Liyun; FALCONE, Gioia. Development of a real-time objective gas-liquid flow regime identifier using kernel methods. **IEEE Transactions on Cybernetics**, IEEE, 2019.

FIGUEIREDO, Maurício de Melo Freire; CARVALHO, Felipe de Castro Teixeira; FILETI, Ana Maria Frattini; SERPA, Alberto Luiz. Flow pattern classification in water-air vertical flows using a single ultrasonic transducer. **Experimental Thermal and Fluid Science**, v. 119, p. 110189, 2020. ISSN 0894-1777. Available at: https://www.sciencedirect.com/science/article/pii/S0894177720306932.

FOUMANI, Navid Mohammadi; MILLER, Lynn; TAN, Chang Wei; WEBB, Geoffrey I.; FORESTIER, Germain; SALEHI, Mahsa. Deep learning for time series classification and extrinsic regression: A current survey. **ACM Comput. Surv.**, Association for Computing Machinery, New York, NY, USA, v. 56, n. 9, Apr. 2024. ISSN 0360-0300. Available at: https://doi.org/10.1145/3649448.

GERS, Felix A; SCHMIDHUBER, Jürgen; CUMMINS, Fred. Learning to forget: Continual prediction with lstm. **Neural computation**, MIT Press, v. 12, n. 10, p. 2451–2471, 2000.

GIOIA, Falcone; HEWITT, GF; ALIMONTI, C. Multiphase flow metering: Principles and applications. **Developments in Petroleum Science series**, v. 54, 2009.

GRAVES, Alex. **Supervised Sequence Labelling with Recurrent Neural Networks**. *[S.l.: s.n.]*, 2012. v. 385. ISBN 978-3-642-24796-5.

GUPTA, Ashish; GUPTA, Hari Prabhat; BISWAS, Bhaskar; DUTTA, Tanima. Approaches and applications of early classification of time series: A review. **IEEE Transactions on Artificial Intelligence**, v. 1, n. 1, p. 47–61, 2020.

HARRIS, Charles R.; MILLMAN, K. Jarrod; WALT, Stéfan J. van der; GOMMERS, Ralf; VIRTANEN, Pauli; COURNAPEAU, David; WIESER, Eric; TAYLOR, Julian; BERG, Sebastian; SMITH, Nathaniel J.; KERN, Robert; PICUS, Matti; HOYER, Stephan; KERKWIJK, Marten H. van; BRETT, Matthew; HALDANE, Allan; RÍO, Jaime Fernández del; WIEBE, Mark; PETERSON, Pearu; GÉRARD-MARCHANT, Pierre; SHEPPARD, Kevin; REDDY, Tyler; WECKESSER, Warren; ABBASI, Hameer; GOHLKE, Christoph; OLIPHANT, Travis E. Array programming with NumPy. **Nature**, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357–362, Sep. 2020. Available at: https://doi.org/10.1038/s41586-020-2649-2.

HASTIE, Trevor; TIBSHIRANI, Robert; FRIEDMAN, Jerome. **The elements of statistical learning: data mining, inference, and prediction**. *[S.l.]*: Springer Science & Business Media, 2009.

HAYKIN, Simon O. **Neural Networks and Learning Machines**. 3rd edition. ed. *[S.l.]*: Prentice Hall, 2008. ISBN 0131471392,9780131471399.

HE, Kaiming; ZHANG, Xiangyu; REN, Shaoqing; SUN, Jian. Deep residual learning for image recognition. *In*: **2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. *[S.l.: s.n.]*, 2016. p. 770–778.

HEWITT, G. F. **Measurement of two phase flow parameters**. London; New York: Academic Press, 1978. ISBN 0123462606.

HEWITT, Geoffrey F. Gas-liquid flow. *In*: **Thermopedia$^{\text{TM}}$**. Begell House, 2010. (accessed: 08.10.2021). Available at: http://dx.doi.org/10.1615/AtoZ.g.gas-liquid_flow.

HOBOLD, Gustavo M.; da Silva, Alexandre K. Visualization-based nucleate boiling heat flux quantification using machine learning. **International Journal of Heat and Mass Transfer**, v. 134, p. 511 – 520, 2019. ISSN 0017-9310.

HOCHREITER, Sepp; SCHMIDHUBER, Jürgen. Long Short-Term Memory. **Neural Computation**, v. 9, n. 8, p. 1735–1780, 11 1997. ISSN 0899-7667. Available at: https://doi.org/10.1162/neco.1997.9.8.1735.

HOWARD, Jeremy *et al.* **fastai**. *[S.l.]*: GitHub, 2018. https://github.com/fastai/fastai.

HUANG, Gao; LI, Yixuan; PLEISS, Geoff; LIU, Zhuang; HOPCROFT, John; WEINBERGER, Kilian. Snapshot ensembles: Train 1, get m for free. 03 2017.

IOFFE, Sergey; SZEGEDY, Christian. Batch normalization: Accelerating deep network training by reducing internal covariate shift. 02 2015.

JONES, Owen C.; ZUBER, Novak. The interrelation between void fraction fluctuations and flow patterns in two-phase flow. **International Journal of Multiphase Flow**, Elsevier, v. 2, n. 3, p. 273–306, 1975.

KARIM, Fazle; MAJUMDAR, Somshubra; DARABI, Houshang; CHEN, Shun. Lstm fully convolutional networks for time series classification. **IEEE Access**, v. 6, p. 1662–1669, 2018.

KARIM, Fazle; MAJUMDAR, Somshubra; DARABI, Houshang; HARFORD, Samuel. Multivariate lstm-fcns for time series classification. **Neural Networks**, v. 116, p. 237–245, 2019. ISSN 0893-6080. Available at: https://www.sciencedirect.com/science/article/pii/S0893608019301200.

KHAN, Umair; PAO, William; PILARIO, Karl Ezra; SALLIH, Nabihah. Flow regime classification using various dimensionality reduction methods and automl. **Engineering Analysis with Boundary Elements**, v. 163, p. 161–174, 2024. ISSN 0955-7997. Available at: https://www.sciencedirect.com/science/article/pii/S0955799724000936.

KIPPING, Ragna. **Entwicklung und experimentelle Validierung von Algorithmen zur Datenreduktion von WMS-Daten**. 2014. Phd Thesis (PhD Thesis) — Technische Universität Dresden, 2014.

KIPPING, Ragna; BRITO, Rosmer; SCHEICHER, Eckhard; HAMPEL, Uwe. Developments for the application of the wire-mesh sensor in industries. **International Journal of Multiphase Flow**, Elsevier, v. 85, p. 86–95, 2016.

KRSTAJIC, Damjan; BUTUROVIC, Ljubomir J; LEAHY, David E; THOMAS, Simon. Cross-validation pitfalls when selecting and assessing regression and classification models. **Journal of cheminformatics**, BioMed Central, v. 6, n. 1, p. 1–15, 2014.

Kumar Vandrangi, Seshu; Alemu Lemma, Tamiru; Muhammad Mujtaba, Syed; OFEI, Titus N. Developments of leak detection, diagnostics, and prediction algorithms in multiphase flows. **Chemical Engineering Science**, v. 248, p. 117205, 2022. ISSN 0009-2509. Available at: https://www.sciencedirect.com/science/article/pii/S0009250921007703.

LI, Huajun; JI, Haifeng; HUANG, Zhiyao; WANG, Baoliang; LI, Haiqing; WU, Guohua. A new void fraction measurement method for gas-liquid two-phase flow in small channels. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 16, n. 2, p. 159, 2016.

LI, Huajun; JI, Haifeng; HUANG, Zhiyao; WANG, Baoliang; LI, Haiqing; WU, Guohua. A new void fraction measurement method for gas-liquid two-phase flow in small channels. **Sensors**, v. 16, n. 2, 2016. ISSN 1424-8220. Available at: https://www.mdpi.com/1424-8220/16/2/159.

LITAK, Grzegorz; GóRSKI, Grzegorz; MOSDORF, Romuald; RYSAK, Andrzej. Study of dynamics of two-phase flow through a minichannel by means of recurrences. **Mechanical Systems and Signal Processing**, v. 89, p. 48 – 57, 2017. ISSN 0888-3270.

LIU, Wen; LV, Xiaofei; JIANG, Sheng; LI, Huazheng; ZHOU, Hao; DOU, Xiangji. Two-phase flow pattern identification in horizontal gas–liquid swirling pipe flow by machine learning method. **Annals of Nuclear Energy**, v. 183, p. 109644, 2023. ISSN 0306-4549. Available at: https://www.sciencedirect.com/science/article/pii/S0306454922006740.

LUCAS, D; BEYER, M; KUSSIN, J; SCHÜTZ, P. Benchmark database on the evolution of two-phase flows in a vertical pipe. **Nuclear engineering and design**, Elsevier, v. 240, n. 9, p. 2338–2346, 2010.

LUCAS, D; BEYER, M; SZALINSKI, L; SCHÜTZ, P. A new database on the evolution of air–water flows along a large vertical pipe. **International journal of thermal sciences**, Elsevier, v. 49, n. 4, p. 664–674, 2010.

MAHVASH, Ali; ROSS, Annie. Application of CHMMs to two-phase flow pattern identification. **Engineering Applications of Artificial Intelligence**, v. 21, n. 8, p. 1144 – 1152, 2008. ISSN 0952-1976.

MCKINNEY, Wes. Data structures for statistical computing in python. *In*: WALT, Stéfan van der; MILLMAN, Jarrod (Ed.). **Proceedings of the 9th Python in Science Conference**. *[S.l.: s.n.]*, 2010. p. 51 – 56.

NUNNO, Fabio Di; PEREIRA, Francisco Alves; MARINIS, Giovanni de; FELICE, Fabio Di; GARGANO, Rudy; MIOZZI, Massimo; GRANATA, Francesco. Deformation of air bubbles near a plunging jet using a machine learning approach. **Applied Sciences**, Multidisciplinary Digital Publishing Institute, v. 10, n. 11, p. 3879, 2020.

OGUIZA, Ignacio. **tsai - A state-of-the-art deep learning library for time series and sequential data**. 2023. Github. Available at: https://github.com/timeseriesAI/tsai.

OUYANG, Lei; JIN, Ningde; REN, Weikai. A new deep neural network framework with multivariate time series for two-phase flow pattern identification. **Expert Systems with Applications**, v. 205, p. 117704, 2022. ISSN 0957-4174. Available at: https://www.sciencedirect.com/science/article/pii/S0957417422009939.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

POELMA, Christian. Measurement in opaque flows: a review of measurement techniques for dispersed multiphase flows. **Acta Mechanica**, v. 231, n. 6, p. 2089 – 2111, 2020. ISSN 1619-6937.

PRASSER, H.-M; BÖTTGER, A; ZSCHAU, J. A new electrode-mesh tomograph for gas-liquid flows. **Flow Measurement and Instrumentation**, v. 9, n. 2, p. 111 – 119, 1998. ISSN 0955-5986. Available at: http://www.sciencedirect.com/science/article/pii/S0955598698000156.

RASCHKA, Sebastian; MIRJALILI, Vahid. **Python Machine Learning, 3rd Ed.** Birmingham, UK: Packt Publishing, 2019. 748 p. ISBN 978-1789955750.

ROSA, ES; SALGADO, RM; OHISHI, T; MASTELARI, N. Performance comparison of artificial neural networks and expert systems applied to flow pattern identification in vertical ascendant gas–liquid flows. **International Journal of Multiphase Flow**, Elsevier, v. 36, n. 9, p. 738–754, 2010.

SCHÜTZ, H; PIETRUSKE, P; MANERA, A; CARL, H; BEYER, M; PRASSER, H-M. **Experiments on upwards gas/liquid flow in vertical pipes**. *[S.l.]*, 2007. Accessed 08.10.2021. Available at: https://www.hzdr.de/publications/Publ-10475.

SESTITO, Guilherme Serpa; BRICEñO, Ricardo Álvarez; RIBATSKI, Gherhardt; da Silva, Maíra Martins; de Oliveira, Leopoldo Pisanelli Rodrigues. Vibration-based multiphase-flow pattern classification via machine learning techniques. **Flow Measurement and Instrumentation**, v. 89, p. 102290, 2023. ISSN 0955-5986. Available at: https://www.sciencedirect.com/science/article/pii/S0955598622001650.

SHANTHI, C.; PAPPA, N. An artificial intelligence based improved classification of two-phase flow patterns with feature extracted from acquired images. **ISA Transactions**, v. 68, p. 425–432, 2017. ISSN 0019-0578. Available at: https://www.sciencedirect.com/science/article/pii/S0019057817302276.

SILVA, MJ Da; SCHLEICHER, E; HAMPEL, U. Capacitance wire-mesh sensor for fast measurement of phase fraction distributions. **Measurement Science and Technology**, IOP Publishing, v. 18, n. 7, p. 2245, 2007.

SILVA, Marco Jose Da. **Impedance sensors for fast multiphase flow measurement and imaging**. 2008. Phd Thesis (PhD Thesis) — Technische Universität Dresden, 2008.

SRIVASTAVA, Nitish; HINTON, Geoffrey; KRIZHEVSKY, Alex; SUTSKEVER, Ilya; SALAKHUTDINOV, Ruslan. Dropout: a simple way to prevent neural networks from overfitting. **J. Mach. Learn. Res.**, JMLR.org, v. 15, n. 1, p. 1929–1958, Jan. 2014. ISSN 1532-4435.

TAITEL, Yehuda; BORNEA, Dvora; DUKLER, AE. Modelling flow pattern transitions for steady upward gas-liquid flow in vertical tubes. **AIChE Journal**, Wiley Online Library, v. 26, n. 3, p. 345–354, 1980.

TAITEL, Yemada; DUKLER, A. E. A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow. **AIChE Journal**, v. 22, n. 1, p. 47–55, 1976. Available at: https://aiche.onlinelibrary.wiley.com/doi/abs/10.1002/aic.690220105.

TIAN, Yingjie; SU, Duo; LAURIA, Stanislao; LIU, Xiaohui. Recent advances on loss functions in deep learning for computer vision. **Neurocomputing**, v. 497, p. 129–158, 2022. ISSN 0925-2312. Available at: https://www.sciencedirect.com/science/article/pii/S0925231222005239.

TORRES, José F.; HADJOUT, Dalil; SEBAA, Abderrazak; MARTÍNEZ-ÁLVAREZ, Francisco; TRONCOSO, Alicia. Deep learning for time series forecasting: A survey. **Big Data**, v. 9, n. 1, p. 3–21, 2021. PMID: 33275484. Available at: https://doi.org/10.1089/big.2020.0159.

VAPNIK, Vladimir N. **The nature of statistical learning theory**. *[S.l.]*: Springer-Verlag New York, Inc., 1995. ISBN 0-387-94559-8.

VASWANI, Ashish; SHAZEER, Noam; PARMAR, Niki; USZKOREIT, Jakob; JONES, Llion; GOMEZ, Aidan N; KAISER, Łukasz; POLOSUKHIN, Illia. Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017.

VIEIRA, Saon Crispim; van der Geest, Charlie; FABRO, Adriano Todorovic; CASTRO, Marcelo Souza de; BANNWART, Antonio Carlos. Intermittent slug flow identification and characterization from pressure signature. **Mechanical Systems and Signal Processing**, v. 148, p. 107148, 2021. ISSN 0888-3270.

WANG, HX; ZHANG, LF. Identification of two-phase flow regimes based on support vector machine and electrical capacitance tomography. **Measurement Science and Technology**, IOP Publishing, v. 20, n. 11, p. 114007, 2009.

WANG, Zhiguang; YAN, Weizhong; OATES, Tim. Time series classification from scratch with deep neural networks: A strong baseline. *In*: **Proceedings of the IEEE International Joint Conference on Neural Network**. *[S.l.: s.n.]*, 2017. p. 1578–1585.

YAN, Yong; WANG, Lijuan; WANG, Tao; WANG, Xue; HU, Yonghui; DUAN, Quansheng. Application of soft computing techniques to multiphase flow measurement: A review. **Flow Measurement and Instrumentation**, Elsevier, v. 60, p. 30–43, 2018.

YAQUB, Muhammad Waqas; MARAPPAGOUNDER, Ramasamy; RUSLI, Risza; Reddy Prasad, D.M.; PENDYALA, Rajashekhar. Flow pattern identification and measurement techniques in gas-liquid-liquid three-phase flow: A review. **Flow Measurement and Instrumentation**, v. 76, p. 101834, 2020. ISSN 0955-5986. Available at: https://www.sciencedirect.com/science/article/pii/S0955598620301758.

YOSINSKI, Jason; CLUNE, Jeff; BENGIO, Yoshua; LIPSON, Hod. How transferable are features in deep neural networks? *In*: **Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2**. Cambridge, MA, USA: MIT Press, 2014. (NIPS'14), p. 3320–3328.

ZEILER, Matthew. Adadelta: An adaptive learning rate method. v. 1212, 12 2012.

ZERVEAS, George; JAYARAMAN, Srideepika; PATEL, Dhaval; BHAMIDIPATY, Anuradha; EICKHOFF, Carsten. A transformer-based framework for multivariate time series representation learning. *In*: **Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining**. New York, NY, USA: Association for Computing Machinery, 2021. (KDD '21), p. 2114–2124. ISBN 9781450383325. Available at: https://doi.org/10.1145/3447548.3467401.

ZHANG, Yongchao; AZMAN, Amirah; XU, Ke-Wei; KANG, Can; KIM, Hyoung-Bum. Two-phase flow regime identification based on the liquid-phase velocity information and machine learning. **Experiments in Fluids**, v. 61, p. 1–10, 10 2020.

ZHANG, Yongchao; AZMAN, Amirah Nabilah; XU, Ke-Wei; KANG, Can; KIM, Hyoung-Bum. Two-phase flow regime identification based on the liquid-phase velocity information and machine learning. **Experiments in Fluids**, Springer, v. 61, n. 10, p. 1–16, 2020.

ZHAO, Bowen; XING, Huanlai; WANG, Xinhan; SONG, Fuhong; XIAO, Zhiwen. Rethinking attention mechanism in time series classification. **Information Sciences**, v. 627, p. 97–114, 2023. ISSN 0020-0255. Available at: https://www.sciencedirect.com/science/article/pii/S0020025523000968.

ZHENG, Jianqiao; RAMASINGHE, Sameera; LUCEY, Simon. **Rethinking Positional Encoding**. 2022. Available at: https://openreview.net/forum?id=fG9WttDhAaa.

ZHU, Jian Hua; MUNJAL, Ritesh; SIVARAM, Ani; PAUL, Santhiyapillai Rajeevan; TIAN, Jing; JOLIVET, Guillaume. Flow regime detection using gamma-ray-based multiphase flowmeter: a machine learning approach. **International Journal of Computational Methods and Experimental Measurements**, WIT Press, v. 10, n. 1, p. 26–37, 2022.