DOI: 10.53660/CLM-2919-24D10

Convolutional Neural Network-Based Pattern Recognition in Natural Circulation Instability Images

Reconhecimento de Padrões Baseado em Rede Neural Convolucional em Imagens de Instabilidade de Circulação Natural

Received: 18-01-2024 | Accepted: 23-02-2024 | Published: 29-02-2024

Sandro Minarrine Cotrim Schott
ORCID: https://orcid.org/0000-0001-7430-454X
Instituto de Pesquisas Energéticas e Nucleares-IPEN/USP, Brazil
E-mail: sandro.schott@usp.br
Marcones Cleber Brito da Silva
ORCID: https://orcid.org/0000-0002-3690-1682
Centro da Fundação Educacional Salvador Arena (CFESA), Brazil
E-mail: marconeseng@gmail.com
Delvonei Alves de Andrade
ORCID: https://orcid.org/0000-0002-6689-3011
Instituto de Pesquisas Energéticas e Nucleares-IPEN/USP, Brazil
E-mail: delvonei@ipen.br
Roberto Navarro de Mesquita
ORCID: https://orcid.org/0000-0002-5355-0925
Instituto de Pesquisas Energéticas e Nucleares-IPEN/USP, Brazil
E-mail: rnavarro@ipen.br

ABSTRACT

Heat removal systems employing natural circulation are key in new nuclear power plant designs for mitigating accidents. This study applies Convolutional Neural Networks (CNNs) to classify 'chugging' instability phases, analyzing 1152 two-phase flow images from a Natural Circulation Circuit. Three CNN models, including one incorporating transfer learning from the ImageNet database, were trained via five-fold cross-validation to fine-tune hyperparameters. This involved comparing models with and without transfer learning against a baseline linear model. A model using a pre-trained Resnet50 with transfer learning accurately classified all 230 samples, outperforming the baseline linear model with an F1-Score of 0.859. The results endorse the use of CNNs with transfer learning for thermohydraulic image analysis in identifying natural circulation instability stages.

Keywords: Convolutional Neural Network; Two-phase flow; Pattern Recognition; Natural Circulation

RESUMO

Sistemas de remoção de calor que empregam circulação natural são fundamentais nos novos designs de usinas nucleares para mitigar acidentes. Este estudo aplica Redes Neurais Convolucionais (CNNs) para classificar fases de instabilidade do tipo 'chugging', analisando 1152 imagens de fluxo bifásico de um Circuito de Circulação Natural. Três modelos de CNN, incluindo um que incorpora aprendizado por transferência da base de dados ImageNet, foram treinados via validação cruzada de cinco dobras para ajustar os hiperparâmetros. Isso envolveu comparar modelos com e sem aprendizado por transferência contra um modelo linear de base. Um modelo usando um Resnet50 pré-treinado com aprendizado por transferência classificou corretamente todas as 230 amostras, superando o modelo linear de base com um F1-Score de 0,859. Os resultados endossam o uso de CNNs com aprendizado por transferência para análise de imagem termohidráulica na identificação de estágios de instabilidade de circulação natural.

Palavras-chave: Redes Neurais Convolucionais; Fluxo Bifásico; Reconhecimento de Padrões; Circulação Natural

INTRODUCTION

Recent designs of nuclear power plant have incorporated heat removal mechanisms that rely on natural circulation phenomena. These systems are passive in nature and do not depend on pumps or other active resources (International Atomic Energy Agency, 2005)). In these systems, the coolant circulation occurs through natural forces such as convection and gravity besides vapor condensation and liquid evaporation (International Atomic Energy Agency, 2005). This intrinsic characteristic enables the continuous operation of the cooling process, in the event of an accident that could potentially damage nuclear power components (International Atomic Energy Agency, 2002).

Natural circulation systems offer simplicity, cost savings, and enhanced safety. However, they have some disadvantages, such as weaker forces involved compared to active systems. It is important to study two-phase flow instabilities that may arise under critical conditions.

Natural circulation instabilities have been studied for many years ((Nayak; Vijayan (2008)). Among the various phases of these instabilities, this study has adopted the classical classification proposed by Boure et al. (1973), which best describes the two-phase flow instability observed in the experimental circuit (Natural Circulation Circuit (NCC) (De Mesquita et al., 2012)) used in this study. In the Bouré classification, this instability is referred as ‘chugging’ and is described as exhibiting three distinct stages of

a cyclic behavior. This characteristic behavior is consistently observed in experiments conducted using NCC (de Mesquita et al., 2012)).

The first stage is called Incubation, which corresponds to the initial formation of vapor bubbles. The second stage, known as Expulsion, occurs when the hot leg of the loop circuit expels the heated refrigerant. The third stage, Refill, describes the moment in the cycle when heated liquid flows back into the hot leg. These three phases occur consecutively and form a continuous cycle.

In two-phase flow scientific investigations, the primary focus is to investigate heat transfer parameters. These parameters exhibit a significant correlation with flow patterns, which, in turn, depend on fluid flow properties such as thermal conductivity, specific heat, density, and viscosity, as well as fluid properties like speed and distribution. Imaging techniques are increasingly being utilized to obtain these parameters and to classify the flow patterns, contributing to improve understanding to these phenomena.

Several studies have focused on investigating two-phase flow in natural circulation using artificial intelligence techniques, such as Fuzzy Logic (de Mesquita et al., 2012; Ghanbarzadeh et al., 2012; Shi, 2007), as well as traditional machine learning methods based on feature extraction. These include Principal Component Analysis (PCA), Support Vector Machine (SVM) (Huang et al., 2011; Shanthi and Pappa, 2017; Xiao et al., 2018).

In recent publications, there has been strong trend towards employing Deep Learning techniques, which eliminate the need for specialists to manually select the best image features, as is the case with classical machine learning techniques, in order to achieve optimum results.

One notable Deep Learning technique is the Convolutional Neural Network (CNN), which has emerged as a leading approach in the field of computer vision (LeCun et al., 2015). The use of CNNs in research began in the 1980s and has significantly evolved, particularly in applications such as character and speech recognition in the late ’90s (LeCun et al., 1998). Since then, there has been a remarkable advancement in machine learning techniques, driven by the increasing processing power, mainly due to the technological development of parallel processing using Graphics Processing Units (GPUs) utilization.

An important milestone in this development of machine learning techniques applied to computer vision was achieved through the renowned ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) (Russakovsky et al., 2015), an international

competition that involved tasks such as image classification, localization, and identification, utilizing a training database of over one million images. This competition demonstrated the remarkable performance of Convolutional Neural Networks (CNNs), leading to a significant breakthrough in the field (Krizhevsky et al., 2012).

Since then, the interest in CNNs has been growing, resulting in numerous studies focusing on image classification, including those related to two-phase flow. For instance, (Du et al., 2018) utilized convolutional neural networks to classify two-phase (oil and water) flow patterns, while Hobold and Da Silva (2019) applied CNNs to differentiate two boiling patterns. Both articles recorded significant accuracy (~99%), surpassing traditional classification approaches. In the domain of two-phase flow image analysis, Silva Junior et al. (2023) incorporated algorithms to predict parameters raging from superficial flow velocities to mass flow rate. Kadish et al. (2022) introduced a method for classifying flow regime and vapor quality in vertical flows. GoogleLeNet with attention mechanisms was employed by Zhang et al. (2023) to distinguish flow patterns within microchannels. Meanwhile, Nie et al. (2022) focused on methane and tetrafluoromethane flow images in horizontal circular tubes, and Seal et al. (2021) analyzed condensation patterns of R-134a refrigerant in inclined tubes. Qiumei et al. (2023) introduced Liqnet, a lightweight architecture, and compared it to six other prominent model structures. Xu et al. (2022) adopted a combination approach, leveraging CNN for feature extraction and Support Vector Machine (SVM) for the classification phase. Lastly, Han et al. (2023) developed a detection method based on Faster R-CNN for real-time bubble recognition and void fraction calculations.

While many studies have addressed image classification in two-phase flow scenarios, the classification of natural circulation visual patterns has not been extensively explored. This work employs two CNN-based models to classify Chugging (Boure et al., 1973) instabilities in natural circulation using an experimental image database.

Both CNNs were optimized over five folds, and their performance was evaluated with and without transfer learning. Initial results were presented at a conference (Schott et al., 2021) and were based on a master's thesis (Schott, 2021). This paper builds on that foundation, refining the model selection with k-fold validation, introducing new models, and delivering improved results. A more detailed description of the methods used is also provided.

METHODOLOGY

Convolutional Neural Networks (CNNs)

Historically, image classification tasks relied on two initial phases: preprocessing and, primarily, feature extraction. Feature extraction models typically required design by a specialist who possessed knowledge to select the most appropriate image features that would better characterize the pattern-of-interest. Once these features were extracted from the image database, they were then utilized as input for training a neural network (LeCun et al., 1998).

CNNs do not require the use of pre-extracted features. Instead, they utilize image data as input during training. CNNs are designed to process inputs defined as multidimensional vectors (tensors), enabling them to receive these images while exploiting the strong correlation between neighboring pixels in these multidimensional matrices. As a result, CNNs excel in classifying small image elements into a limited number of classes, such as borders, curves, lines, etc. Conventional neural networks lack this capability when processing images (LeCun et al., 2015).

There are three fundamental concepts in architectures of CNNs when applied to images: locally connected regions, shared weights between neurons, and subsampling. These concepts are implemented through two main types of neural network layers: the Convolutional layer, which incorporates the first two concepts, and the Pooling layer, responsible for implementing subsampling (Goodfellow et al., 2016).

These concepts contribute to an overall invariance of CNNs in relation to displacements, scaling, and distortions, while also reducing the number of parameters that need to be trained. The convolutional layers in CNNs consist of digital image filters that scan the image, performing operations similar to the application of Sobel or Gaussian filters in image processing (Gonzalez and Woods, 2007). Each filtering the convolutional layer produces a feature map as its output. It is important to note that these filters are not fixed during the training process. Instead, they are continually updated to select the relevant features for the recognition task.

CNNs typically consist of convolutional layers, where the initial layers identify simpler features, while the subsequent layers identify more complex features (Goodfellow et al., 2016). Following a convolutional layer, the information passes through an activation function, introducing non-linearity into the training network.

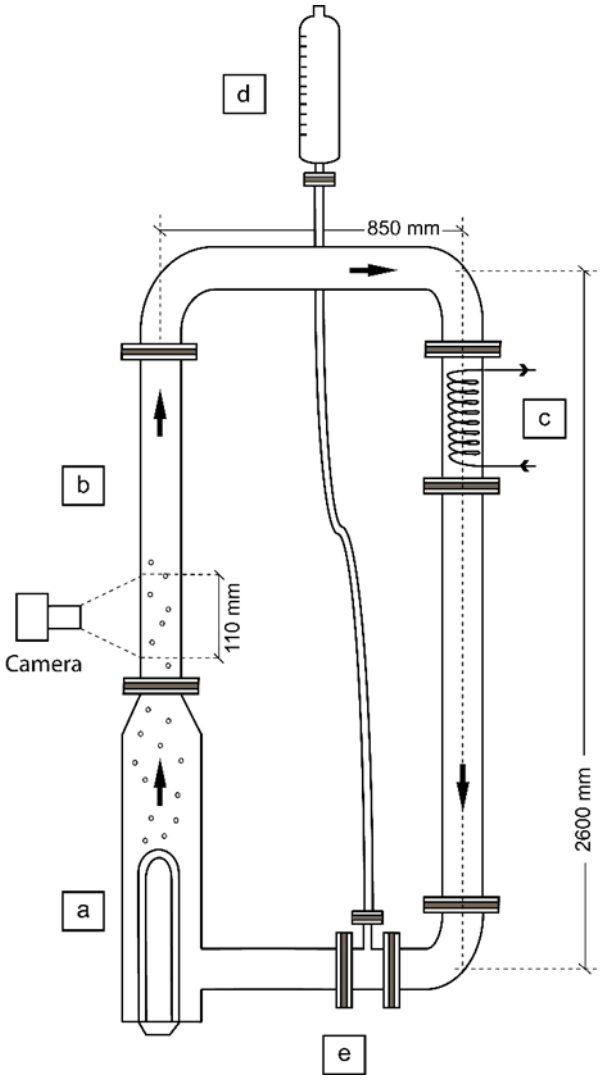
The Pooling layer is responsible to provide robustness to the CNN by enabling invariance to small rotation and scaling. This is achieved through sampling, resulting in dimensional reduction and, consequently, a decrease in the number of CNN parameters. Consequently, overfitting is avoided, and a memory usage is reduced (Goodfellow et al., 2016).

Image database formation

This work utilizes an experimental database previously formed and used in prior studies (de Mesquita et al., 2012, 2018). A detailed description of this database can be found in (de Mesquita et al., 2012). These images were obtained from experiments conducted on an experimental setup called the Natural Circulation Circuit (NCC). This circuit consists of glass tubes arranged in a U-shape, allowing easy visualization of one-phase and two-phase flows under atmospheric pressure.

This circuit supports twelve liters of demineralized water that circulates through the tube loop in a clockwise direction; rising in the hot leg and descending in the cold leg (see Figure 1 – section a). The heat source comprises two 8000W electrical resistors for heating the water flow. The cooler is simulated by a helical heat exchanger positioned in the right leg, which receives tap water flowing internally at room temperature (typically ranging from 15 to 35 °C, depending on local weather conditions). Further details about the operation of this circuit can be found in Mesquita et al. (2009). Figure 1 illustrates the NCC schematic, highlighting various segments, including the visualization section.

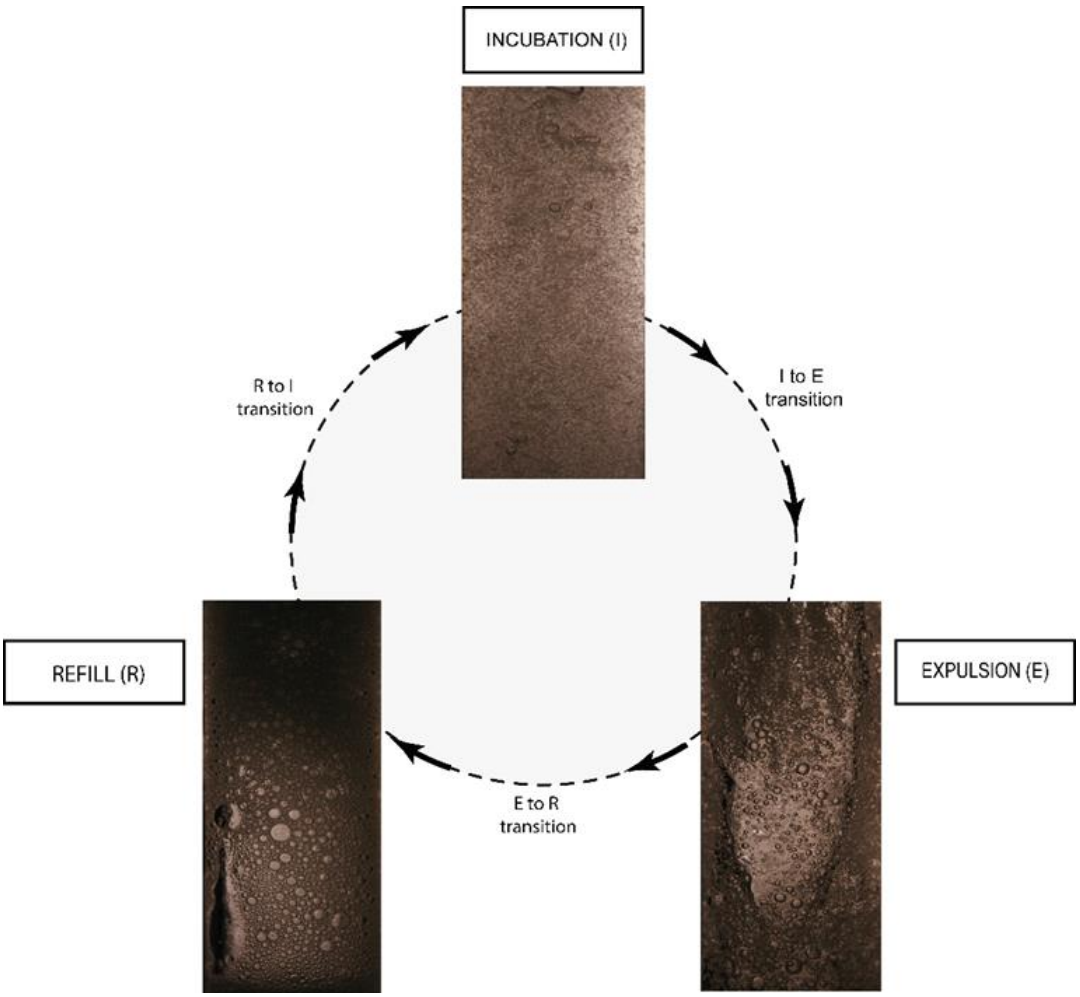
Figure 1 – Schematic of the Natural Circulation Circuit (NCC) showcasing its sections: heating (a), visualization (b), cooling (c), expansion tank (d) and lower (e).



In this study, NCC experiments were conducted to investigate three distinct stages of a common natural circulation two-phase flow instability known as Chugging: Incubation (I), Expulsion (E), and Refill (R). This cyclical instability was termed by Boure et al. (1973). After the initial two-phase flow stabilizes, each stage display consistently maintains its duration, represented by distinct cyclic periods. These periods collectively make up the overall cycle period, as illustrated in Figure 2.

The image patterns corresponding to each chugging stage are captured through image acquisition conducted in the hot leg using backlight illumination. A database of those experimentally acquired images was organized based on each of these stages. The cyclical occurrence of these consecutive and periodical stages is illustrated in Figure 2, representing typical images for each stage.

Figure 2 – Chugging instability cycle observed in the NCC, depicting its typical phases and corresponding time intervals.



All images were acquired at high resolution using a camera with shutter speed of 250 µs. The images were acquired with a longitudinal section of 110 mm and a transverse section of 43.6 mm, resulting in a resolution of 3888x2692 pixels, corresponding to 0.03 mm/pixel. The images were acquired in RGB mode (Mesquita et al., 2009).

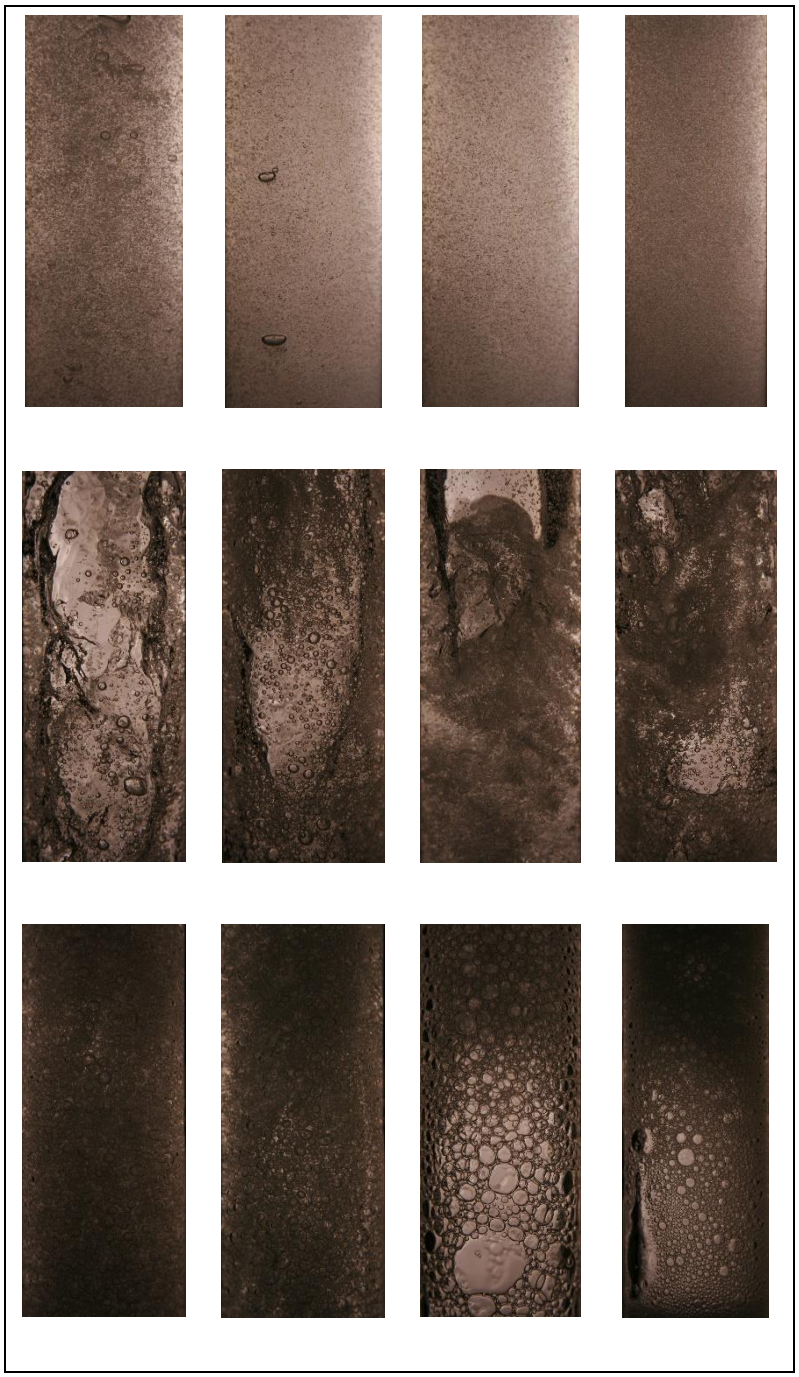
The present work utilized 1152 images from this database. The chugging stages were selected based on their occurrence time within a stable cyclic behavior of the chugging cycle.

The images were processed to extract a Region-of-Interest (ROI), excluding the tube borders. CNNs were trained using these ROIs as direct inputs from these models. Figure 3 displays a representative ROI from this database.

Figure 3 – Typical image of the Incubation stage image with red lines delineating the Region-of-Interest (ROI).

After the initial extraction of the ROI, the image resolution is reduced by a factor of ten using bicubic interpolation (Gonzalez and Woods, 2007) implemented in OpenCV. These transformations were implemented to minimize computational costs. The images were resized to 304x136x3 pixels after interpolation. Examples of typical images after interpolation are shown in Figure 4.

Figure 4 – Typical chugging images after preprocessing. The first, second and third lines represent the Incubation, Expulsion, and Refill patterns, respectively.



The image patterns in the first line of Figure 4 depict the Incubation phase, characterized by the increasing presence of vapor bubbles. The second line of Figure 4 shows images that exhibit collapsing vapor bubbles forming slugs, resulting in the expulsion of the hot leg fluid in jet-like formations, which characterizes the Expulsion phase. The third line in Figure 4 displays Refill images, representing the return of cooled liquid from the superior section to refill the hot leg and create a foam region. These phases occur cyclically.

Splits and Classes

The length of each chugging differs, with Incubation being the most extended and making up the bulk of the cycle. In comparison, the Expulsion phase is briefer. Due to these variations, each stage yielded a different number of acquired images, given that the acquisition rate was consistent at roughly one image per second.

Table 1 – Images distribution by class and Split.

	Train (80%)	**Test (20%)**	**Total**
Incubation	489	122	611 (53%)
Refill	322	80	402 (35%)
Expulsion	111	28	139 (12%)
Total	922	230	1152

The duration of each chugging stage varies. Incubation is the longest stage, constituting the larger part of the cycle, while Expulsion is shorter. These variations in duration result in each class having a different number of acquired images, given the constant acquisition speed (about one image/s).

The database was divided into two splits: Train and Test. The Train split was further subdivided into 5-Folds for hyperparameter selection in two CNNs, one utilizing transfer learning and the other without it. Each image is used once for validation to estimate the achieved training generalization by taking the minimum loss for this Split (Early Stopping) (Goodfellow et al., 2016). Only the best hyperparameter values are reported in this study. To derive measurements, the metrics were averaged across the five folds. Following the determination of the best hyperparameters, a new training session was conducted to evaluate the top-performing model on the Test split. The proportion of images used for each class and split can be found in Table 1.

Training

The models were implemented using the Tensorflow Python library (Pang et al., 2020), with the Keras module. These deep learning techniques require the use of GPUs to training. This Colab environment was utilized in this work to train the proposed models.

The primary regularization technique utilized was the Early Stopping (Pang et al., 2020) where from all training epochs, only the lower-cost state is considered in the Validation split. The model is saved in this state and the Test is performed to evaluate accuracy and F1-Score. Another regularization used was the Dropout (Srivastava et al., 2014), a technique that randomly turns off a portion of the neurons. In this work, the Dropout parameter used was p = 0,5 (He et al., 2016).

As a training stopping criterion, a 'patience' parameter of ten epochs was set, meaning that if there is no improvement in the loss of Validation split for ten consecutive epochs, the training is terminated.

The 16-batch size was kept fixed due to the observed good performance at this value. The ADAM optimization (Kingma and Ba, 2015) was used in all training sessions with α=0.001, β1= 0.9, β2= 0.999 and ε=10-8.

For the baseline models and the one without the use of transfer learning, the pixel values of input images were normalized between 0 and 1 by dividing them by 255. This normalization procedure was performed to enhance convergence speed. For the model using ResNet (He et al., 2016), an appropriate preprocessing provided by the Tensorflow library was employed.

The correct labelling was performed using the one-hot encoding (Goodfellow et al., 2016). In this method, with 'n' being the number of classes, the true labels are represented in a 'y' vector with 'n' positions, where each position represents a class. Therefore, a '1' is placed in the position of the represented class, and '0's are filled in the remaining vector positions.

At the end of the model, the 'n' outputs are passed through a Softmax activation function (Theodoridis, 2015), which normalizes the outputs to values between 0 and 1, with the sum of outputs being equal to 1. Through this transformation, the outputs can be

interpreted as probabilities or confidences for each class prediction. These values constitute the model output vector, ‘y.’ The class considered as the model’s prediction is the one with the highest value in ‘y.’

Subsequently, the loss function is evaluated as input for parameter optimization. The loss function used was cross-entropy (Theodoridis, 2015). The initialization of parameters initialization for convolutional and fully-connected layers was performed using the Uniform Glorot technique (Glorot and Bengio, 2010).

Models

Three models were used, two based on CNN, and one simple linear model as a baseline. To evaluate the necessity and performance of transfer learning, one model was used with the technique, and another without it. The models were named Resnet, CONV, and Linear. A hyperparameter selection was performed using 5-folds, and the best values are described below.

The first model was configured using the Transfer Learning technique, which allows the final and additional training to take on a different database from where the initial training was performed (Weiss et al., 2016). This property can be especially useful for small-sized databases.

The typical procedure for transfer learning involves using a pre-trained neural network, such as Resnet (He et al., 2016), which was previously trained with the ImageNet database (Jia Deng et al., 2009) consisting of millions of examples. This neural network was trained with this large database until achieving satisfactory performance. Subsequently, the Fully-Connected layers were discarded, and only the trained Convolutional layers block remained (Chollet, 2021).

The underlying idea behind this approach is that this remaining block may have learned, through training with such a large number of images, to select general features and not only specific features from the original training session.

This work used the ResNet50 (He et al., 2016) provided by TensorFlow using transfer learning. Thus, the convolutional layers were frozen, and starting from the conv5_block3_three_conv layer, they were unfrozen. At the end of the convolutional layers, a Global Average Pooling (GAP) layer was added. This layer significantly reduces the architectures’ size as the number of neurons is expressively reduced. The inclusion of

the GAP layer not only shrinks the architecture but also acts as a regularization technique (Lin et al., 2013).

After this layer, a fully-connected layer with 1024 neurons using ReLu activation was added. Then, a dropout layer with p=0,5 was added. The classification task in this work involves only three classes; hence, the last layer was defined with three neurons using Softmax activation. This last layer is identical for all the three models, as the classification task is the same.

The second neural network model, named CONV, was used to evaluate the need for transfer learning; therefore, this model does not utilize this technique. This model was designed with two convolutional layers interspersed with max pooling layers, followed by a fully-connected layer composed of 256 neurons, and it uses dropout with p=0,5. The convolutional parameters were: 64 filters in the first convolutional layer and 128 filters in the second one. Both of these layers used 3-sized filters, a stride size of one, and padding used the Keras configuration named 'same'. The activation function used was ReLU. The max-pooling layers used a 2-sized filter, and the stride was set to two.

The LINEAR model is formed by the remaining three neurons, directly connected to all image pixels, and there is no hidden layer of neurons. In this model, there is no ReLU activation function that would insert non-linearity. This model was used as the performance baseline for this work.

Classification Performance Metrics

The performance evaluation of the proposed models was based on the traditional methodology used in machine learning (Murphy, 2012). The used metric is defined as follows. Based on N examples, the accuracy could be expressed as shown in the Equation 1:

$$Accuracy = (TP+TN)/(TP+TN+FP+FN) \quad (1),$$

where:

- *TP* is the number of True Positives,
- *TN* is the number of True Negatives,
- *FP* is the number of False Positives, and
- *FN* is the number of False Negatives.

It is common to express these values in percentage format. The sum of all these parameters should equal N.

The accuracy metric could lead to distorted evaluations when there is an imbalance between the number of samples for each class. Therefore, other metrics are more commonly used, such as Precision, Recall and F1-Score. These metrics are usually applied for each class, and then one can take the average across the classes, and the average-weighted metric by the number of examples of each class, as defined in Equations 2, 3, and 4:

$$Precision = {}^{TP}/_{(TP + FP)} \quad (2),$$

$$Recall = TP/(TP + FN) \quad (3),$$

$$F1 - Score = 2 * Precision * Recall/(Precision + Recall) \quad (4).$$

RESULTS

The results of the classification task for each proposed model are described in this section. Each model was detailed in section 3.5. The training performance metrics obtained for each model are shown in Table 3, with the average for each class of metrics.

Table 2 – Performance metrics of trained models on 5-Folds averaged across the three classes.

	Precision	**Recall**	**F1-Score**	**Accuracy**
Resnet	1	1	1	1
CONV	0,976	0,979	0,977	0,987
Linear	0,860	0,860	0,859	0,921

By analyzing Table 2, it is possible to observe that Resnet achieved the maximum performance for the 5-fold. The maximum values indicate that the network correctly classified all samples in all five folds. This fact suggests that the architecture is sufficient or even more than sufficient for this classification task's complexity.

Another important result of the training session is that the CONV has only misclassified one out of the 230 Validation samples. These results demonstrate that the convolutional deep learning methods (using or not transfer learning) are more favorable to easily surpass classical neural networks not based on convolutional layers, such as MLP. Additionally, during the training phase, a slight advantage has been observed for the two first two models that utilized transfer learning.

The Resnet model achieved superior performance using transfer learning compared to CONV; therefore, it was considered the best model and will be used for evaluation in the test split. However, given the simplicity of the CONV model, it can be considered a good result, especially when compared to the baseline. This demonstrates that the non-linearities between the layers and the convolutional layers exhibit their ability to extract relevant features for the current classification task, as argued by (LeCun et al., 2015).

This result also suggests that is preferable to customize a CNN model based on original Resnet and transfer learning than to develop one CNN from scratch and try to optimize it by varying the number of layers or the number of filters for instance. Another result is that based on the baseline measures, it is also possible to infer that this classification task is not too challenging for machine learning models that use thousands of parameters and heavily rely on computational power through GPUs for training assignment.

Since Resnet was the best model in the 5-Folds, it was evaluated in the test split as shown in Table 3.

Table 3 – Performance metrics of trained models on the test split.

	Precision	**Recall**	**F1-Score**
Incubation	1	1	1
Expulsion	1	1	1
Refill	1	1	1
Average	1	1	1

The results indicate that Resnet accurately classifies all images within the test split, underscoring the aptitude of CNNs with transfer learning for this specific challenge. Notably, this performance is attained without the manual feature extraction often seen in

prior studies, such as the approach by de Mesquita et al. (2012). The findings suggest the potential for tackling more complex tasks related to this issue, like introducing a broader range of classes encompassing intermediate Chugging stages.

Additionally, this study sought to gauge the CNN's efficiency when faced with a reduced training sample size. Such an exploration aids in determining the minimum image count necessary to compile a representative database that ensures reliable performance. To this end, a subset of the training data was used, with the evaluation carried out on the unchanged test split. Detailed results can be found in Table 4.

Table 4 – Performance metrics of the reduced dataset of Resnet.

Percentage of the training split used	Precision	Recall	F1-Score	Accuracy
1%	0,573	0,658	0,611	0,870
5%	0,993	0,976	0,984	0,991
10%	0,989	0,996	0,992	0,996
25%	1	1	1	1
50%	1	1	1	1

From Table 4, it can be observed that even when reducing the training set to 25% and 50% of the original training data size, the performance remains consistent. Additionally, with just 5%, the F1-Score of 0,984 is still higher than the baseline of 0,859. Therefore, it can be concluded that, for this problem, Resnet does not require many examples to achieve satisfactory performance; approximately 250 training images are sufficient

CONCLUSION

This research explored the use of CNNs to classify images of unstable two-phase flow captured from a natural circulation loop, the Natural Circulation Circuit (NCC). The NCC experiments simulated 'chugging' instability with regular periodic cycles, with three well-defined stages with regular time duration. The names of the stages of the instability

periodic cycle (Incubation, Expulsion and Refill) were used to label a set of experimentally acquired images.

Three different models were created and were used to classify these images into three Chugging-stage classes. The image database consisted of 1152 image samples. These models underwent 5-fold training to determine the best hyperparameters for two of them. During this stage, the objective was to evaluate which architecture would perform better, using transfer learning or without the use of this technique. For better comparison, a third linear model was used as a baseline.

One of the models was based on a pre-trained Resnet50 (He et al., 2016) using transfer learning from the ImageNet (Jia Deng et al., 2009) database. The other two models did not utilize transfer learning and were designed in decreasing complexity order: the first of these three still employed two convolutional layers and the last one did not include a fully-connected layer.

This last model, referred as LINEAR, served as a baseline for evaluating the task's difficulty. This model achieved an average F1-Score of 0,859, suggesting that the problem might not be too challenging for the remaining models.

The comparison between two different models was conducted using the five folds, indicating a clear superior performance of the model using transfer learning resources. The best performance among the five folds was achieved by the architecture based on Resnet, utilizing 1024 neurons in a fully-connected layer and adding a Global Average Pooling layer. This model was selected for testing on a separate test split. Resnet obtained an average F1-Score of 1, corresponding to correctly classifying all 230 samples in the Test split.

It was also observed that the model with convolutional layers without using transfer learning exhibited favorable results in the 5-fold training phase, suggesting its potential for application in similar classification tasks with limited computational resources.

The findings of this study underscore the efficacy of CNNs in classifying two-phase flow, especially when applied to experimentally captured images of natural circulation instabilities. The standout performance of the Resnet-based architecture,

which uses transfer learning, demonstrates the potential of deep learning in discerning intricate patterns in such experimental data.

Deep Learning techniques, especially CNNs, are relatively new to thermohydraulic scientific investigations. This work's success with experimental images of natural circulation instabilities highlights the potential of deep learning in this field, suggesting further research in similar applications.

ACKNOWLEDGMENTS

We acknowledge the financial support provided by the Comissão Nacional de Energia Nuclear (CNEN) under the project number 01342.003313/2022-25.

REFERENCES

Boure, J. A., Bergles, A. E., and Tong, L. S.: Review of Two-phase flow instability, Nuclear Engineering and Design © North-Holland Publishing Company, 25, 165–192, https://doi.org/10.1016/0029-5493(73)90043-5,1973.

Chollet, F.: Deep Learning with Python, Second Edition, 2021.

Du, M., Yin, H., Chen, X., and Wang, X.: Oil-in-water two-phase flow pattern identification from experimental snapshots using convolutional neural network, IEEE Access, 7, https://doi.org/10.1109/ACCESS.2018.2888733, 2018.

Ghanbarzadeh, S., Hanafizadeh, P., and Hassan Saidi, M.: Intelligent image-based gas-liquid two-phase flow regime recognition, Journal of Fluids Engineering, Transactions of the ASME, 134, https://doi.org/10.1115/1.4006613, 2012.

Glorot, X. and Bengio, Y.: Understanding the difficulty of training deep feedforward neural networks, in: Proceedings of Machine Learning Research, 9: 249-256, https://proceedings.mlr.press/v9/glorot10a.html, 2010.

Gonzalez, R. C. and Woods, R. E.: Digital Image Processing (3rd Edition), 2007.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep learning An MIT Press Book, 2016.

Han, B., Ge, B., Wang, F., Gao, Q., Li, Z., and Fang, L.: Void fraction detection technology of gas-liquid two-phase bubbly flow based on convolutional neural network, Exp Therm Fluid Sci, 142, https://doi.org/10.1016/j.expthermflusci.2022.110804, 2023.

He, K., Zhang, X., Ren, S., and Sun, J.: ResNet-dev, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9908 LNCS, 2016.

Hobold, G. M. and da Silva, A. K.: Automatic detection of the onset of film boiling using convolutional neural networks and Bayesian statistics, Int J Heat Mass Transf, 134, https://doi.org/10.1016/j.ijheatmasstransfer.2018.12.070, 2019.

Huang, G., Ji, H., Huang, Z., Wang, B., and Li, H.: Flow regime identification of mini-pipe gas-liquid two-phase flow based on textural feature series, in: Conference Record - IEEE Instrumentation and Measurement Technology Conference, https://doi.org/10.1109/IMTC.2011.5944218, 2011.

International Atomic Energy Agency: Natural circulation data and methods for advanced water cooled nuclear power plant designs, VIENNA, https://www-pub.iaea.org/MTCD/Publications/PDF/te_1281_prn.pdf, 2002.

International Atomic Energy Agency: Natural circulation in water cooled nuclear power plants: phenomena models and methodology for system reliability assessments., International Atomic Energy Agency, 635 pp., https://www-pub.iaea.org/MTCD/Publications/PDF/TE_1474_web.pdf, 2005.

Jia Deng, Wei Dong, Socher, R., Li-Jia Li, Kai Li, and Li Fei-Fei: ImageNet: A large-scale hierarchical image database, https://doi.org/10.1109/cvprw.2009.5206848, 2009.

Kadish, S., Schmid, D., Son, J., and Boje, E.: Computer Vision-Based Classification of Flow Regime and Vapor Quality in Vertical Two-Phase Flow, Sensors, 22, https://doi.org/10.3390/s22030996, 2022.

Kingma, D. P. and Ba, J. L.: Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, https://doi.org/10.48550/arXiv.1412.6980, 2015.

Krizhevsky, A., Sutskever, I., and Hinton, G. E.: ImageNet classification with deep convolutional neural networks, in: Advances in Neural Information Processing Systems, https://doi.org/10.1145/3065386, 2012.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P.: Gradient-based learning applied to document recognition, Proceedings of the IEEE, 86, https://doi.org/10.1109/5.726791, 1998.

LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning, Nature, 436-444, https://doi.org/10.1038/nature14539, 2015.

Lin, M., Chen, Q., and Yan, S.: Network In Network, http://arxiv.org/abs/1312.4400, 2013.

Mesquita, R. N., Libardi, R. M. P., Masotti, P. H. F., Sabundjian, G., Andrade, D. A., Umbehaun, P. E., Torres, W. M., Conti, T. N., Macedo, L. A., and Engineering Conferences International.: Two-phase Flow Patterns Recognition and Parameters Estimation Though Natural Circulation Test Loop Image Analysis, in: 7th ECI International Conference on Boiling Heat Transfer 2009, 541–547, https://inis.iaea.org/collection/NCLCollectionStore/_Public/40/079/40079244.pdf, 2009.

de Mesquita, R. N., Masotti, P. H. F., Penha, R. M. L., Andrade, D. A., Sabundjian, G., Torres, W. M., and Macedo, L. A.: Classification of natural circulation two-phase flow

patterns using fuzzy inference on image analysis, Nuclear Engineering and Design, 250, https://doi.org/10.1016/j.nucengdes.2012.06.014, 2012.

de Mesquita, R. N., Castro, L. F., Torres, W. M., Rocha, M. D. S., Umbehaun, P. E., Andrade, D. A., Sabundjian, G., and Masotti, P. H. F.: Classification of natural circulation two-phase flow image patterns based on self-organizing maps of full frame DCT coefficients, Nuclear Engineering and Design, 335, https://doi.org/10.1016/j.nucengdes.2018.05.019, 2018.

Murphy, K. P.: Machine Learning: A Probabilistic Perspective. Cambridge, MA: MIT Press, https://www.cs.ubc.ca/~murphyk/MLbook/pml-toc-1may12.pdf, 2012.

Nayak, A. K. and Vijayan, P. K.: Flow instabilities in boiling two-phase natural circulation systems: A review, https://doi.org/10.1155/2008/573192, 2008.

Nie, F., Wang, H., Song, Q., Zhao, Y., Shen, J., and Gong, M.: Image identification for two-phase flow patterns based on CNN algorithms, International Journal of Multiphase Flow, 152, https://doi.org/10.1016/j.ijmultiphaseflow.2022.104067, 2022.

Pang, B., Nijkamp, E., and Wu, Y. N.: Deep Learning With TensorFlow: A Review, Journal of Educational and Behavioral Statistics, 45, 227–248, https://doi.org/10.3102/1076998619872761, 2020.

Qiumei, Z., Yukun, H., Fenghua, W., Zhang, P., and Chao, L.: Liqnet: A real-time monitoring network for two-phase flow patterns, Flow Measurement and Instrumentation, 90, https://doi.org/10.1016/j.flowmeasinst.2023.102313, 2023.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge, Int J Comput Vis, 115, https://doi.org/10.1007/s11263-015-0816-y, 2015.

Schott, S. M. C.: Classificação de Padrões de Escoamento Bifásico Por Meio de Redes Neurais Convolucionais, Master thesis, USP - University of São Paulo, São Paulo, https://doi.org/https://doi.org/10.11606/D.85.2021.tde-10062022-125100, 2021.

Schott, S. M. C., Da Silva, M. C. B., and De Mesquita, R. N.: Classification of two-phase flow instability phases using convolutional neural networks, in: 10º edition of the international Nuclear Atlantic Conference - INAC 2021, https://inac2021.aben.com.br/resumos/R0290-1.pdf, 2021.

Seal, M. K., Noori Rahim Abadi, S. M. A., Mehrabi, M., and Meyer, J. P.: Machine learning classification of in-tube condensation flow patterns using visualization, International Journal of Multiphase Flow, 143, https://doi.org/10.1016/j.ijmultiphaseflow.2021.103755, 2021.

Shanthi, C. and Pappa, N.: An artificial intelligence based improved classification of two-phase flow patterns with feature extracted from acquired images, ISA Trans, 68, https://doi.org/10.1016/j.isatra.2016.10.021, 2017.

Shi, L.: Fuzzy recognition for gas-liquid two-phase flow pattern based on image processing, in: 2007 IEEE International Conference on Control and Automation, ICCA, https://doi.org/10.1109/ICCA.2007.4376595, 2007.

Silva Junior, L. H., Barbosa, J. R., and da Silva, A. K.: Multi-parameter classification and quantification of R-134a condensation using machine learning, Appl Therm Eng, 231, 120880, https://doi.org/10.1016/j.applthermaleng.2023.120880, 2023.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R.: Dropout: A simple way to prevent neural networks from overfitting, Journal of Machine Learning Research, 15, https://jmlr.org/papers/v15/srivastava14a.html, 2014.

Theodoridis, S.: Neural Networks and Deep Learning, in: Machine Learning, 875–936, https://doi.org/10.1016/b978-0-12-801522-3.00018-5, 2015.

Weiss, K., Khoshgoftaar, T. M., and Wang, D. D.: A survey of transfer learning, J Big Data, 3, https://doi.org/10.1186/s40537-016-0043-6, 2016.

Xiao, J., Luo, X., Feng, Z., and Zhang, J.: Using artificial intelligence to improve identification of nanofluid gas-liquid two-phase flow pattern in mini-channel, AIP Adv, 8, https://doi.org/10.1063/1.5008907, 2018.

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

Zhang, J., Wei, X., and Wang, Z.: The Recognition Algorithm of Two-Phase Flow Patterns Based on GoogLeNet+5 Coord Attention, Micromachines (Basel), 14, https://doi.org/10.3390/mi14020462, 2023.