Computer Vision and Deep Learning Based Determination Of Flow Regimes, Void Fraction And Resistance Sensor Data In Microchannel Flow Boiling

Mark Schepperle, Shayan Junaid, and Peter Woias

Affiliation not available

November 28, 2023

Abstract

The aim of this article is to introduce a novel approach to identifying flow regimes and void fractions in microchannel flow boiling, which is based on binary image segmentation using digital image processing and deep learning. The proposed image processing pipeline uses adaptive thresholding, blurring, gamma correction, contour detection and histogram comparison to separate vapour from liquid areas, while the deep learning method uses a customized version of a convolutional neural network (CNN) called Unet to extract meaningful features from video frames. Both approaches enabled automatic detection of flow boiling conditions, such as bubbly, slug, and annular flow, as well as automatic void fraction calculation. Especially the CNN has demonstrated its ability to deliver fast and dependable results, presenting an appealing substitute to manual feature extraction. The U-net-based CNN was able to segment flow boiling images with a Dice score of 99.1 % and classify the above flow regimes with an overall classification accuracy of 91 %. In addition, the neural network was able to predict resistance sensor readings from image data and assign them to a flow state with a mean squared error (MSE) < \(10^{-6}\). This sensor signal prediction is a promising first step towards automated, imageless prediction of two-phase flow in microchannels using only the measurement data.
Abstract—The aim of this article is to introduce a novel approach to identifying flow regimes and void fractions in microchannel flow boiling, which is based on binary image segmentation using digital image processing and deep learning. The proposed image processing pipeline uses adaptive thresholding, blurring, gamma correction, contour detection and histogram comparison to separate vapour from liquid areas, while the deep learning method uses a customized version of a convolutional neural network (CNN) called U-net to extract meaningful features from video frames. Both approaches enabled automatic detection of flow boiling conditions, such as bubbly, slug, and annular flow, as well as automatic void fraction calculation. Especially the CNN has demonstrated its ability to deliver fast and dependable results, presenting an appealing substitute to manual feature extraction. The U-net-based CNN was able to segment flow boiling images with a Dice score of 99.1% and classify the above flow regimes with an overall classification accuracy of 91%. In addition, the neural network was able to predict resistance sensor readings from image data and assign them to a flow state with a mean squared error (MSE) \( < 10^{-6} \). This sensor signal prediction is a promising first step towards automated, imageless prediction of two-phase flow in microchannels using only the measurement data from resistance sensors. The approaches discussed in this paper were performed on an ordinary 6 GB NVIDIA laptop GPU using Python and are general enough to be applied to other similar applications. The deep learning model can be downloaded from: github.com/schepperlemark

Index Terms—Computer vision, convolutional neural network, deep learning, image processing, microchannel flow boiling

I. INTRODUCTION

Two-phase flow boiling in microchannel heat sinks is one of the most important topics in the field of fluid flow and heat transfer, especially due to the promising application in high heat flux cooling [1]–[3]. It offers significant advantages for the cooling of power electronics, computer chips, laser diodes and other electronic components due to its large heat transfer surface area and compact design. This approach offers two main benefits: an increased heat transfer coefficient and increased heat dissipation capacity, even at low mass flow rates [4]. Microchannel flow boiling heat transfer is currently one of the most promising approaches for removing significant heat loads from electronic devices [5].

To further improve the flow boiling mechanism it is important to have a better understanding of the vapor bubble dynamics, the flow regime and void fraction inside the microchannel [6]. Typical flow patterns observed in the microchannel during flow boiling are bubbly, slug, and annular flow, as well as the chaotic mixtures and transitions that occur between them [7]. The knowledge of these patterns allows for the derivation of several significant thermal-hydraulic parameters, such as two-phase viscosity and two-phase density, which can be instrumental in forecasting heat transfer coefficients and pressure drop in microchannel cooling devices. The void fraction represents the fraction of the cross-sectional area occupied by the vapour phase compared to the total cross-sectional area [8]. Typically, this void fraction can be determined with high precision by optical investigations at specific locations within the microchannel using high-speed video (HSV) imaging [9]. Alternatively, the flow regime and void fraction is estimated from electrical resistance and impedance measurements, relating resistance and impedance to void fraction and flow regime [10], [11]. However, it is a time-consuming task to manually determine flow regimes and void fractions, whether electrically...
or optically.

For this reason, in response to the rapid progress of computer vision techniques, a few researchers have started to use image processing and machine learning to automatically analyze HSV images of two-phase flows [12]. The use of these tools not only enables the rapid analysis of several thousand HSV image frames in a matter of seconds while maintaining a standardized quantitative approach, but also serves as a safeguard against the omission of crucial information due to human error.

In 2011, Hanafizadeh et al. [13] employed basic image processing techniques, including color format conversion, image subtraction, median filtering, and threshold segmentation, to create a binary representation of two-phase flow patterns in the up-riser of airlift pumps. However, the data analysis was still conducted manually, using a ruler, rather than utilizing computer vision (CV) methods. In another study, Singh et al. [14] conducted a more advanced image analysis technique that was performed to generate flow regime maps of flow boiling water in silicon microchannels. The captured HSV images were subjected to sophisticated image processing steps using background removal, cropping, color format conversion, histogram equalization, median filtering, edge detection, contour filling, and binary liquid/vapor region segmentation to determine local void fractions at multiple locations along the microchannel. The void fractions were then used to automatically identify and separate bubbly, slug, and annular flow regimes using predefined void fraction thresholds selected based on careful visual observations.

Very recently researchers started to apply machine learning techniques to study flow boiling in microchannels. However, most of the studies focus on the prediction of heat transfer coefficients and pressure drops based on universal consolidated data under the use of artificial neural networks [15]–[20]. So far, only a handful of studies have used machine learning based on convolutional neural networks (CNN) to automatically detect bubbles, classify flow regimes, and calculate void fractions from HSV images taken during two-phase flow processes [21]–[24].

Kim et al. [21] harnessed the Mask R-CNN to create an automated tool for bubble detection and mask extraction in gas-liquid two-phase flows. Their model was trained with a combination of experimental and synthetic bubbly flow images from upward bubbly flows in expansion pipes [25] and from utilizing the BubGAN algorithm [26]. The trained model reached an averaged precision AP$_{90}$ of 98 % on unseen test data from the experimental bubbly flow. Additionally, the model was tested on unseen bubble-swarm flows [27] not included in the training set where it was able to detect 95 % of the bubbles. Although the results are remarkable, the model is limited to detecting bubbly flow at low void fraction values. It lacks the ability to detect more complex two-phase flows such as slug and annular flow. The model also does not include flow regime classification.

In another study, Kadish et al. [22] classified vapor quality and flow regimes of vertical two-phase (vapor-liquid) CO$_2$ flow images captured at frame rates of 10 fps and 30 fps, respectively, in an 8 mm diameter transparent circular channel using a CNN with ResNet101 for image feature extraction and a deep long short-term memory (LSTM) network to incorporate temporal information of image sequences. The model was trained on a dataset of 39,261 manually labeled image frames using cross-entropy loss and the Adam optimization function at a learning rate of $10^{-3}$, a batch size of 256 for 60 epochs on an NVIDIA® Kepler™ K40 M GPU with 12 GB of GPU accelerator memory. For the flow identification task, there were two output layers in the high-level network architecture called FrameNet and FlowNet. The FrameNet output layer came directly after the CNN, skipping the LSTM network. The FlowNet output layer, on the other hand, came after the LSTM network, using the CNN and the LSTM network for flow regime classification. The authors validated the performance of both flow classification models using 5-fold cross-validation on unseen test data, resulting in an accuracy of 91.8 % for FlowNet and 92.3 % for FrameNet, which in our opinion casts doubt on whether the LSTM network really helps to improve the classification accuracy of flow regime classification.

In the present study, to the best of our knowledge, we are the first to use state-of-the-art image processing techniques and deep learning for binary image segmentation, automatic void fraction calculation, flow regime classification and RTD sensor signal prediction during two-phase flow boiling in microchannels. For machine learning, we used a CNN based on the U-net architecture [28], using a very limited data set of only 4,010 HSV image frames recorded during the flow boiling process in the microchannel as input. The U-net-based CNN classified microchannel flow boiling into 7 flow regimes: bubbly, bubbly-slug, slug, slug-annular with bubbles, slug-annular, annular with bubbles and annular flow. The RTD sensor signal prediction was intended to be the first step towards automated flow regime detection based on electrical sensor data without cost-intensive optical inspection. The information about the present flow regime during microchannel flow boiling by automated evaluation of electrical sensor signals from RTDs could pave the way for automated flow regime control in commercial microchannel heat sinks by initiating AI-based control signals to adjust the present flow regime by means of the read-out sensor data. For example, as we have shown in [29], microheaters could be actuated to adjust the flow regime in microchannels via short low power heating pulses.

II. MATERIALS AND METHODS

In this section the experimental setup, the computer vision and deep learning techniques are described. The computer vision and deep learning code can be downloaded from: github.com/schepperlemark.

A. Experimental Flow Boiling Apparatus

The experimental setup that was used to capture microchannel (MC) two-phase flow boiling videos and sensor data from thin-film platinum RTDs is illustrated in Fig. 1. This setup was also used in similar configurations in [10], [24], [29]–[32], where further details about the setup and its application can be found if interested. The MC with a depth of 0.5 mm, a width
of 1.5 mm and a length of 65 mm was milled into a 5 mm x 8 mm x 68.6 mm stainless steel block. The RTDs where manufactured on a Pyrex glass wafer in a clean room process that is described in detail in [10]. The important components of the setup for the experiments in this paper are: A DI water reservoir, a micro pump to supply DI water from the reservoir to the stainless-steel MC, a 3D printed housing for mechanical fixation and hermetic sealing of the MC, thin-film platinum RTDs on a transparent Pyrex glass lid mounted above the MC, a glass-wool wrapping around the housing for thermal isolation from the environment, heater cartridges inserted at the MC bottom to heat the MC and initiate two-phase flow boiling, a Phantom VEO 410L high-speed camera mounted on a microscope fixture with a 2x magnifying objective to capture the videos of the MC flow boiling process at a frame rate of 2000 fps, a linear stage that moved the microscope fixture including the high-speed camera along the MC to record flow boiling at different MC locations and a MFIA impedance analyzer from Zurich Instruments to read out the RTD measurements.

Fig. 2 shows the housing with the MC, the RTDs and the cartridge heaters in more detail. The housing lid pressed the glass lid with the RTDs on top of the MC in such a way, that all RTDs where in direct contact with the flow boiling fluid. The MC was surrounded by an O-ring which ensured a watertight sealing. The boiling process could be observed through the observation window in the middle of the housing lid. The spring probes provided an electrical contact to the RTDs via the gold contact pads at each side of the RTDs. The MFIA impedance analyzer was connected via a trigger cable to the high-speed camera which provided a time synchronized video and RTD sensor data acquisition. The RTDs detected the temperature induced resistance changes that were caused by the temperature fluctuations of the flowing fluid \( T_{\text{fluid}} \) inside the MC. The resistance/temperature dependency of the RTDs can be described with following formula:

\[
R_{\text{RTD}} = \alpha (T_{\text{fluid}} - T_0) R_0 + R_0, \tag{1}
\]

with the measured RTD resistance \( R_{\text{RTD}} \), the temperature coefficient of resistance \( \alpha = 2.98 \cdot 10^{-3} \, ^\circ\text{C}^{-1} \), the fluid temperature \( T_{\text{fluid}} \), the RTD room temperature resistance \( R_0 \), and the room temperature \( T_0 \) [10].

Vapor bubbles passing a RTD structure could be detected by an increase of the measured resistance \( R_{\text{RTD}} \) due to a temperature increase of \( T_{\text{fluid}} \) beneath the RTD to the boiling temperature of water at atmospheric pressure (\( \approx 100 \, ^\circ\text{C} \)).

### B. Video Data Pre-Processing

Due to the length of the MC of 65 mm, it was not feasible to capture the entire flow boiling phenomenon at once in one video. Therefore, several videos of subsections were recorded along the MC to cover the whole channel length. Fig. 3 shows at the top a raw unprocessed video frame of such a MC subsection. Each raw frame of a subsection video underwent a pre-processing pipeline to extract only the meaningful flow boiling pixel data. As shown at the bottom of Fig. 3 the pixel area of the flowing fluid inside the MC was considered for further processing, while the rest of the pixel area was discarded. The thin platinum RTDs in Fig. 3 were critical for this cropping process. Three vertical platinum RTDs were present in the video frames of each subsection. These platinum RTDs acted as separator lines and were manufactured similarly to the thin-film platinum RTDs for resistance measurements, but with a width of only 20 µm to minimize obstruction of the view into the MC. An algorithm was developed to detect these vertical lines and crop the area between the outer two
platinum RTD separator lines (ignoring the platinum RTD in the middle) and the MC walls. Therefore, each video represented a subsection covering the entire length of the flow boiling in the MC. This cropping process prevented overlapping pixels of subsection areas and focused solely on the flow boiling in each channel section. Another important reason for the pre-processing is the substantial reduction of computational cost during the application of computer vision and deep learning. The cropping reduced the effective frame size by 92% from 800 x 1280 pixels to 112 x 700 pixels. It is important to be aware of the fact, that in this pre-process one separator line of a subsection is always part of a neighboring subsection. E.g. the outer left channel part next to the left separator line in Fig. 3 which gets cropped is visible in the neighboring subsection on the left where this line represents the separator line on the right. The pre-processing algorithm is robust to varying input frame pixel sizes and varying lightning conditions.

C. Computer Vision For Binary Image Segmentation

During MC flow boiling the fluid prevails in two states of matter: gaseous and liquid. This is tailor-made for the application of binary image segmentation to further reduce the image complexity, bit depth and memory size of the pre-processed video frames by setting the pixel intensity of liquid areas to 255 (white) and the pixel intensity of vapor areas to 0 (black). In preceding work on computer vision-based vapor bubble segmentation in two-phase flows, like in [33] and [34] the vapor bubbles that occurred were small and had regular spherical or elliptical shapes. In this case the Hough circle transform can be used to detect the presence of vapor bubbles whereas the rest of the section is considered as liquid. However, in our MC there was a recurring instance of more complex irregular shapes of vapor bubbles which could not be detected by the Hough Transform. Other fundamental challenges were textured lines at the MC background as well as differentiating the inside of vapor bubbles from the liquid part. The textured MC background was removed by an image optimization process (Fig. 4) that used adaptive thresholding, blurring and gamma correction. This process created an initial segmentation that did not yet differentiate between vapor bubble inside and liquid sections. A contour detection (Fig. 4) and histogram comparison (Fig. 6) finally enabled the differentiation of pixel areas inside the vapor from pixel areas of the liquid. This was done by correlating the pixel intensity histograms of these pixel areas with the same pixel areas of a single liquid only background photo (Fig. 5) of the same MC subsection captured at adiabatic conditions before the two-phase flow boiling videos were recorded. The histograms were correlated utilizing the magnitude of the Pearson correlation coefficient \(d\) which takes values between 1 (full correlation) and 0 (no correlation). It is defined as:

\[
d(x, y) = \left| \frac{\sum_{i=1}^{n} (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} \right|, \\
\]

where

\[
\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i, \quad \bar{y} = \frac{1}{n} \sum_{i=1}^{n} y_i,
\]

with the histogram size \(n\), the individual histogram data points \(x_i, y_i\), and the histogram mean values \(\bar{x}, \bar{y}\).

Fig. 6 compares pixel intensity histograms of a liquid contour (H2 in Fig. 6b) and a contour detected inside a vapor bubble (H4 in Fig. 6c) of a flow boiling video frame (Fig. 6a).
Fig. 6. Pixel intensity histogram correlations \(d(H_1, H_2)\) and \(d(H_3, H_4)\) of a flow boiling video frame (a) of the liquid contour \(H_2\) (b) and the contour detected inside the vapor bubble \(H_4\) (c) with the pixel intensities \(H_1\) and \(H_2\) of the same pixel areas of a liquid only background image shown in Fig. 5.

with the pixel intensity histograms (\(H_1\) and \(H_3\) respectively) of the same pixel areas of a liquid only background photo and states the histogram correlations \(d(H_1, H_2)\) and \(d(H_3, H_4)\). The pixels that have been part of the contours and therefore have been used as data sets for the pixel intensity histograms are highlighted in green yellow. The pixel areas in dark purple are not included in the respective histogram data sets. The liquid only background images in Fig. 6 highlight and discard the same pixel areas as the flow boiling video frames, even when the whole area was composed of liquid. This was important, so that always the same pixel areas were compared with each other. The histogram correlation of pixel area 1 was \(d(H_1, H_2) = 0.97\) and the histogram correlation of pixel area 2 was \(d(H_3, H_4) = 0.76\). After analyzing several flow boiling video frames with a histogram correlation sweep it was found that for a histogram correlation of \(d \geq 0.85\) most of the compared contour pixel areas were liquid. Respectively, below this value, most of the pixel areas were vapor inside areas. Therefore, 0.85 was set as threshold value for labelling contours as liquid. As a last step of the binary segmentation process, all pixel intensity values of pixels labeled as liquid were set to 255 (white). All other pixel intensities were labeled as vapor inside and set to 0 (black). The cross-sectional void fraction which represents the proportion of the area taken up by the vapor phase in relation to the total cross-sectional area was automatically calculated for each binary segmentation frame as follows:

\[
\text{Void fraction} = \frac{\text{Pixels with a pixel intensity of 0}}{\text{Total No. of MC subsection pixels}} \tag{3}
\]

The computer vision algorithm saved all input two-phase flow boiling video frames and the resulting binary segmented output frames in designated folders on the computer hard disk.

D. Deep Learning For Binary Image Segmentation, Flow Regime Classification and RTD Data Prediction

Using classical computer vision methods to distinguish vapour from liquid areas in the MC gave good results (see section III), but some pixel areas were still misclassified as liquid or vapour despite all efforts to fine-tune the parameters of adaptive thresholding, blurring, gamma correction and histogram correlation. This led to the question of whether or not deep learning would help to improve the results of
binary image segmentation. The first step was to manually inspect the binary segmentation directory for mis-segmented images, which were then manually corrected using a custom-built manual segmentation function. A training and test dataset was then created in an 80:20 ratio. For this purpose, 4,010 pre-processed input video frames and manually corrected output binary segmentation frames were selected from randomly chosen sections of the respective directories to ensure that data from all recorded videos were selected in both the training and test datasets. The Dice coefficient below was used as the loss function:

\[
\text{Dice coefficient} = \frac{2 \cdot \text{Overlapping pixel area}}{\text{Total No. of pixels}}
\]

The overlapping pixel area is the area of overlap between the predicted binary segmentation of the model and the training or test data respectively. The Dice Coefficient is one of the most common loss functions used for image segmentation [35].

The deep learning model used for binary segmentation is a modified form of the U-net architecture published by Ronneberger et al. 2015 [28]. The following modifications were made to the original model:

- Input layer image channels were set to 3, as the video frames are in RGB format.
- Output layer image channels were set to 1 since the masked images obtained were in grey scale form, with 0 denoting pixels identified as vapour and 1 denoting pixels identified as liquid.
- The kernel size of the first convolutional layer was set to 7x7 with a Rectifier Linear Unit (ReLU) activation function. This change helped to process larger images with half the computational cost.

As with the original U-net model, concatenated skip links were used to ensure feature reusability. The DL model for binary segmentation was trained using Dice coefficient loss and the Adam optimization function [36] at a learning rate of 10\(^{-5}\), a batch size of 32 for 50 epochs on a 6 GB NVIDIA GTX3060 laptop GPU and is referred to as VoidNet in this paper. The simplified network architecture for all DL models is illustrated as a flowchart in Fig. 7.

Some modifications were made to the deep learning model described above for flow regime classification (designated as FlowBoilNet) and RTD sensor data prediction (designated as SensorNet). As shown in Fig. 7, FlowBoilNet and SensorNet use only the Conv2D downscaling path to classify flow regimes and predict RTD signals from extracted downscaled image features. However, the randomized selection of training and test data remained constant at an 80:20 ratio.

For flow regime classification, the 4,010 pre-processed video frames were manually labeled as expected output according to the current flow regime in the form of 3x1 tensors. Fig. 8 shows the labeled training and test data distribution for FlowBoilNet. The model output layer of the modified U-net model was adapted accordingly to output the predicted flow regime in the form of 3x1 tensors. All possible tensors are listed in Tab. I. The loss function was changed to cross-entropy, one of the most common loss functions for classification problems [37], and the learning rate was set to 10\(^{-4}\).

For RTD sensor data prediction, 12,340 video frames were automatically labeled with the corresponding measured RTD resistance and split 80:20 into training and test data. A trigger cable connected between the impedance analyzer and the high-speed camera synchronized the video frames with the measured RTD data. The output layer of the modified U-net model was adjusted to output real numbers representing the predicted RTD resistance. The mean square error (MSE) was used as the loss function to solve the linear regression problem. An overview of all hyperparameters is shown in Tab. II.

**III. RESULTS AND DISCUSSION**

Segmented HSV image frames using the CV method described in Subsection II-C are shown in Fig. 9. The majority...
of all video frames could be correctly segmented in binary using CV. From separated and overlapping bubbles of different shapes and sizes (Fig. 9a) to slug and annular flow (Fig. 9b), the CV method was able to successfully segment most of the HSV images even for complex vapor/liquid mixtures. However, approximately 15% of all frames were incorrectly segmented. This was mainly because the image processing could not always distinguish the inside of the gas bubbles from the liquid segments, as shown in Fig. 9c, despite careful fine-tuning of the histogram correlation \( d \) to a liquid only background image (Fig. 5).

Fig. 10 shows an input frame from the DL test data set and the corresponding prediction of the trained VoidNet model after 50 epochs compared to the binary segmentation of the CV method. It can be seen that VoidNet reliably distinguishes between liquid and gaseous regions. Overall, the VoidNet model achieves a dice score of 99.1% after 50 epochs in both the training and test runs, significantly outperforming the CV method.

Fig. 11 shows the classification result of FlowBoilNet for all flow regimes given in Tab. I. FlowBoilNet was able to classify flow regimes with an overall classification accuracy of 91%. The normalized confusion matrix shown in Fig. 12 illustrates the ratio between the true and predicted values of the trained FlowBoilNet model for all classified flow rates (Tab. I), with the individual flow regime accuracies shown in the diagonal from top left to bottom right. A normalized confusion matrix is a good way to visualize the accuracy of each class. This is especially helpful in the case of imbalanced data sets, as the overall accuracy of 91% does not reveal classification biases across classes. In Fig. 12, it can be seen that for some flow rates like bubbly, bubbly slug and annular flow FlowBoilNet performs very well and classifies the flow regime correctly with an accuracy between 97 to 100%. However, for other flow rates the FlowBoilNet model performs not that good and the classification accuracy goes even below 50% like in the case of slug flow.

This difference in classification accuracy of the trained deep learning model can be described with the imbalanced data distribution of the test and training data set shown in Fig. 8. The accuracy of flow regime classification decreases significantly as the amount of training data for each flow regime decreases. For example, the slug flow regime is the least represented with 57 frames during training run. Accordingly, the slug flow regime has the lowest accuracy with 33%, followed by annular with bubbles with 57% and slug annular with 63%. However, the normalized confusion matrix demonstrates that flow regimes are merely confused with very similar neighboring flow regimes. E.g. the slug flow is only confused with the very similar slug annular and annular flow.

**TABLE I**

<table>
<thead>
<tr>
<th>Flow regime</th>
<th>3x1 output tensor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bubbly</td>
<td>[1 0 0]</td>
</tr>
<tr>
<td>Slug</td>
<td>[0 1 0]</td>
</tr>
<tr>
<td>Annular</td>
<td>[0 0 1]</td>
</tr>
<tr>
<td>Bubbly slug</td>
<td>[1 1 0]</td>
</tr>
<tr>
<td>Annular with bubbles</td>
<td>[1 0 1]</td>
</tr>
<tr>
<td>Slug annular</td>
<td>[0 1 1]</td>
</tr>
<tr>
<td>Slug annular with bubbles</td>
<td>[1 1 1]</td>
</tr>
</tbody>
</table>

**TABLE II**

<table>
<thead>
<tr>
<th>Hyperparameter</th>
<th>VoidNet</th>
<th>FlowBoilNet</th>
<th>SensorNet</th>
</tr>
</thead>
<tbody>
<tr>
<td>Loss</td>
<td>Dice coefficient</td>
<td>Cross entropy</td>
<td>MSE</td>
</tr>
<tr>
<td>Optimizer</td>
<td>Adam</td>
<td>Adam</td>
<td>Adam</td>
</tr>
<tr>
<td>Learning rate</td>
<td>(10^{-5})</td>
<td>(10^{-4})</td>
<td>(10^{-6})</td>
</tr>
<tr>
<td>Batch Size</td>
<td>32</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>Epochs</td>
<td>50</td>
<td>50</td>
<td>50</td>
</tr>
</tbody>
</table>
The prediction of the RTD data is shown in Fig. 13 as an example for different kinds of flow patterns. In general, an impressive loss of $< 10^{-6}$ was achieved in the training run. However, two divergence peaks occurred in the test run, indicating problems with overfitting or lack of generalization. Overall, the trained deep learning models produced reliable results thanks to the state-of-the-art U-net architecture, which was able to extract meaningful representations from the microchannel flow boiling video frames. However, for better results, the use of more training data could further improve the accuracy of the model. Increasing the training data set is one of the most effective ways to address the lack of generalization. The experimental setup is subject to different lighting conditions and slightly tilted microchannel positions. Training the model on a variety of HSV image frames will result in more accurate predictions, especially between very similar flow regimes such as slug, slug-annular and annular flow.

REFERENCES


Mark Schepperle was born in Lorrach, Germany, in 1992. He received his master's degree in Microsystems Engineering from the University of Freiburg, Germany, in 2020. During his bachelor thesis in 2016, he developed a new process for manufacturing ceramic multilayer devices at the Laboratory for Materials Process Technology, University of Freiburg. At the same time, he founded a startup in 2015, which he successfully managed until the end of 2017. Since 2020, he has been working as a PhD student at the Laboratory for Design of Microsystems, Department for Microsystems Technology (IMTEK), where he studies evaporation processes in microchannels.

Shayan Junaid was born in Karachi, Pakistan, in 1994. He received his master's degree in Embedded Systems Engineering from the University of Freiburg, Germany, in 2023. His master's thesis was with regards to the application of artificial intelligence tools such as computer vision and deep learning in the field of two-phase flow boiling. He also holds a bachelor's degree in the field of electrical engineering and has worked as an assistant manager at the largest vertically integrated utility company in Pakistan that serves a population of more than 20 million people. He is currently involved with Fraunhofer IPM as a software developer.
Peter Woias was born in Passau, Germany, in 1962. He received the Diploma degree in electrical engineering and the Ph.D. degree in electrochemical sensing from TU Munich in 1988 and 1995, respectively. From 1995–2000, he has been the Department Director of the Fraunhofer-Institute for Solid State Technology and Integrated Circuit Technology, Munich. Since 2000, he has been a Full Professor and the Head of the Laboratory for Design of Microsystems, Department for Microsystems Technology (IMTEK), University of Freiburg, Germany. Since 2007, he is a member of the technical program committee and the international steering committee of the PowerMEMS conference.