

## RESEARCH ARTICLE

# Aspects of image data preparation to extend a classification scheme for cleaning mechanisms to realistic soils

Christian Golla<sup>1</sup>  | Ludwig Boddin<sup>1</sup> | Hannes Köhler<sup>2</sup> | Frank Rüdiger<sup>1</sup> | Jochen Fröhlich<sup>1</sup>

<sup>1</sup>Institute of Fluid Mechanics, Technische Universität Dresden, Dresden, Germany

<sup>2</sup>Institute of Natural Materials Technology, Technische Universität Dresden, Dresden, Germany

## Correspondence

Christian Golla, Institute of Fluid Mechanics, Technische Universität Dresden, Dresden, Germany.  
Email: [christian.golla@tu-dresden.de](mailto:christian.golla@tu-dresden.de)

## Abstract

Knowledge of the cleaning mechanism is necessary to choose a suitable model for a cleaning simulation. In the present work, an existing classification scheme for cleaning mechanisms is considered. Although this framework is quite promising, the generation of training data constitutes a bottleneck, since the labeling was done manually and very roughly in order to supply the necessary amount of samples in a reasonable time. This, in turn, causes the scheme to be inaccurate when applied to more realistic data. The aim of the present work is to improve the preparation of training data preparation by introducing a semi-automatic labeling procedure. The labeling procedure involves a new perspective on the data and the application of a gradient filter procedure. Furthermore, fully convolutional networks (FCNs) are employed to generalize different gradient filter. The labeling procedure is significantly faster and more consistent than manual labeling. Also, a proof of concept is provided showing that the FCNs are a suitable technique for the present classification task.

## 1 | INTRODUCTION

Cleaning and decontamination is one of the most important topics in the food processing industry [1]. Multiple cleanings per day may be required since various products are produced on the same facilities [2]. Various authors claim that cleaning raises high economical and ecological costs. To name an example: Rad and Lewis [3] analyzed the water consumption of market milk processors and found out that cleaning causes 28% of total water consumption, which is the highest among different contributions. Alvarez et al. [4] found out that cleaning of processing machines in dairy industry causes a daily downtime of 4–6 h. Although the mentioned figures were collected a decade ago, the problem is more relevant than ever before in times of energy revolution.

Although cleaning raises high costs, its dimensioning is done mostly empirically [5]. Cleaning processes are not fully optimized yet, due to a lack of methods. Simulation of cleaning processes would be a feasible approach for systematic variation of operating parameters with subject to reduce ecological and economical costs [6]. However, cleaning processes are very complex and often involve time-dependent material properties and turbulent multiphase flows [7]. A cost efficient way of conducting cleaning simulations of film-like soils with accuracies reasonable for industrial application is

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 The Authors. *Proceedings in Applied Mathematics & Mechanics* published by Wiley-VCH GmbH.

the Boundary Condition Cleaning Model (BCCM) approach, first introduced by Joppa et al. [8]. In a preliminary step, the flow within the component of interest is simulated by means of a computational fluid dynamics (CFD) simulation without considering the dimensions of the soil. In the actual cleaning simulation, the flow field is frozen and a scalar transport equation is solved for the transport of the soil within the flow. The knowledge about the soil and its removal is transformed into a suitable boundary condition for the scalar transport equation. In follow-up works of the same authors further cleaning models were developed with the concept of decoupling flow simulation and calculation of soil removal [6, 9, 10]. However, the newer models no longer require solving a scalar transport equation, since soil removal is described differently. The way how this is done depends on the cleaning behavior of the soil, that is, the specific cleaning mechanism that is active in a particular situation [7]. Hence, knowledge about the cleaning mechanism is mandatory for the applicability of the aforementioned models.

Various authors distinguish soils according to their cleaning mechanism. A detailed discussion of the literature can be found in Golla et al. [7]. The cleaning mechanisms considered in the present paper are based on the work of Köhler et al. [11]. These cleaning mechanisms are diffusive dissolution, cohesive separation, adhesive detachment, and viscous shifting (depicted in Figure 1 of Golla et al. [7]). Diffusive dissolution is the transport of soil molecules into the cleaning fluid, driven by a concentration gradient. Cohesive separation occurs, when the cohesive strength within the soil is exceeded by hydrodynamic loads and soil chunks are removed. Since cohesive separation of small soil chunks close to the soil surface is similar to diffusive dissolution, the mechanisms are impossible to distinguish from a macroscopic point of view. Therefore, within this paper, the term cohesive separation addresses both aforementioned cleaning mechanisms. Adhesive detachment is the counterpart of cohesive separation. Here, the adhesive strength between soil and substrate is overcome by hydrodynamic loads and large soil patches are removed at once. Finally, in case of viscous shifting the soil is or becomes flowable due to certain physical or chemical effects (e.g., melting).

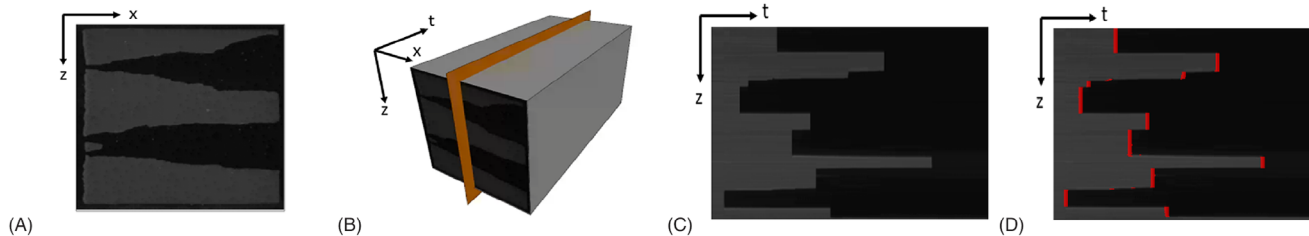
In previous work, the present authors developed a machine learning (ML) algorithm that automates and objectifies the identification of the cleaning mechanism based on grayscale image data from cleaning experiments under usage of feed forward neural networks (NNs) [7]. The NNs were trained using image data generated in cleaning experiments with model soils, exhibiting a clear cleaning mechanism. Later, they were applied to more realistic soils with spatial and temporal variation of cleaning mechanism. Excellent accuracies above  $> 95\%$  were achieved in predicting the dominating cleaning mechanism throughout the whole cleaning process on the model soils. However, when predicting the cleaning mechanism resolved in time, the algorithm achieved only up to 80% accuracy. The application to more realistic soils exhibited a good qualitative agreement, however, no quantitative assessment of the performance was made since the labeling procedure was not applicable to realistic soils. For the data labeling, the cleaning experiments were investigated frame by frame and regions belonging to a certain cleaning mechanism for a certain time were tagged accordingly. No pixelwise investigation was done, since this would have been too cumbersome manually. On the other hand pixel-by-pixel and frame-by-frame labeled data would be necessary to obtain both, a classification scheme working on realistic soils and the possibility to quantitative assessment.

In the present paper, an improved labeling strategy will be developed to fulfill this requirement and a first realization for a classification algorithm processing this data will be provided. A family of algorithms designed to obtain pixel-by-pixel segmentation of image data are FCNs [12]. To label the data, a procedure based on gradient filters is introduced.

## 2 | CLEANING EXPERIMENTS AND DATA PREPARATION

The cleaning experiment utilized in this work are described in detail in Golla et al. [7], so that only a brief summary is given here. The experiments were conducted with dried starch (pregelatinized waxy maize starch), dried ketchup, and petroleum jelly, each soil type being representative for one cleaning mechanism. A duct with rectangular cross section was used, with one of the larger side walls transparent to observe the cleaning process and the opposed wall soiled with the substance of interest. The cleaning process was monitored using a grayscale camera and the measured quantity was the intensity  $I$ . In Figure 2 of Golla et al. [7] sample images and evolutions of gray value over time are shown for each cleaning mechanism.

One major change applied in the present work is the way the data are observed during labeling. Previously, the videos were investigated frame by frame: this makes it easy to recognize cleaning but it is hard to perfectly locate pixels, where a cleaning mechanism is active. In the present work, instead of observing  $x$ - $z$ -planes in each frame  $n$ ,  $z$ - $t$ -planes are now the starting point. An example of how such a view is created is shown in Figure 1. The way the data are presented now makes it easier to locate the cleaning process in space and time, and for the educated viewer, it is very easy to distinguish whether a suggested labeling is suitable or not.



**FIGURE 1** Changed labeling view. (A) Original view on  $x$ - $z$ -plane. (B) Video footage as three-dimensional block. (C) New view on  $z$ - $t$ -plane. (D) Regions associated with cleaning in the  $z$ - $t$  view.

In the present study, a total data basis of four experiments for each cleaning mechanism was employed. Three of these four experiments were used for training purpose. In each of the remaining experiments, one  $z$ - $t$ -plane was labeled manually to allow quantitative assessment of the accuracy. The manual labeling was done by investigating the evolution of the gray value over time for each  $z$ -location and took around 30 min for each  $z$ - $t$ -plane.

### 3 | METHODOLOGY

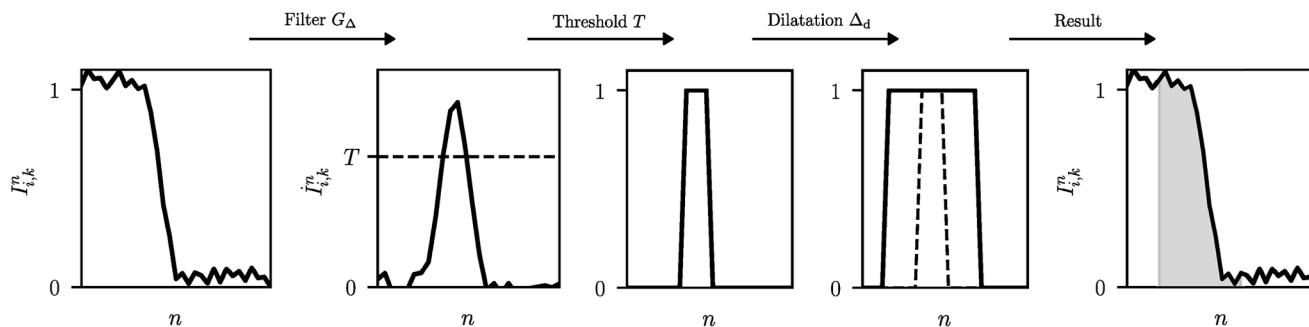
#### 3.1 | Gradient filter procedure

A typical image processing technique is edge detection, which will be a fundamental in the method for detection of cleaning modes presented here. When applying edge detection, the image gradient is computed numerically using finite differences. This is equivalent to performing the convolution of the image with a gradient filter, such as  $G_1 = [-1 \ 0 \ 1]$  (see Gonzales and Woods [13]). The gradient filter procedure applied here consists of three steps and is illustrated in Figure 2. To detect cleaning, a generalized gradient filter of the form

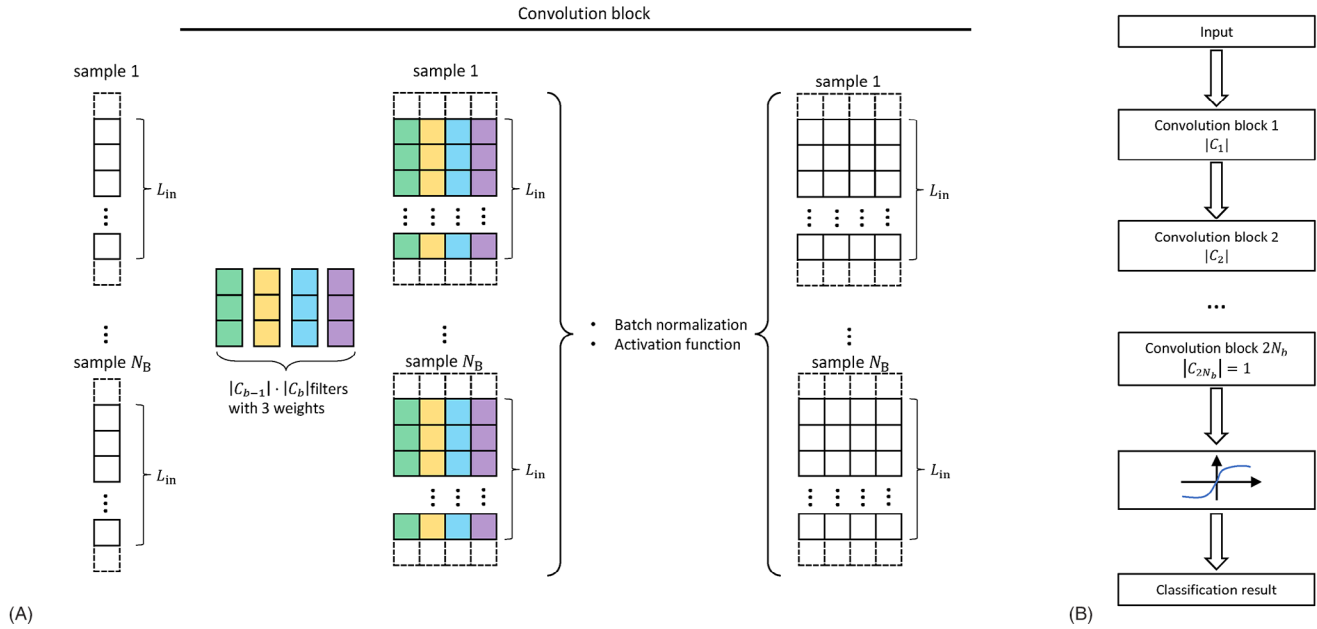
$$G_{\Delta} = \left[ \underbrace{\dots \ -1 \ -1 \ -1}_{\Delta \text{ times}} \ 0 \ \underbrace{1 \ 1 \ 1 \ \dots}_{\Delta \text{ times}} \right], \tag{1}$$

is applied, where  $\Delta$  is the filter radius. A small filter radius serves well to locate steep gradients, while a large filter radius identifies flat gradients. As a first step, the temporal gradient was computed. Assuming the gray values being stored as  $I_{i,k}^n$ , where  $i$  is the pixel index in  $x$ -direction, and  $k$  the pixel index in  $z$ -direction, respectively. Given a frame with frame index  $n$ , the absolute value of the temporal gradient is calculated as

$$\dot{I}_{i,k}^n = \sum_{\tilde{n}=-\Delta}^{\Delta} \left| I_{i,k}^{n+\tilde{n}} G_{\Delta}^{\tilde{n}} \right|. \tag{2}$$



**FIGURE 2** Gradient filter procedure consisting of three steps, starting from a gray scale signal in time: 1. Application of the filter  $G_{\Delta}$ , 2. Application of the threshold  $T$ , 3. Application of the dilatation radius  $\Delta_d$ . Result is the labeling of the gray scale evolution by a cleaning mechanism, here the gray shaded laps of time.



**FIGURE 3** Data processing through the FCN. (A) Data processing within a convolution block of the FCN. Each sample refers to an evolution of gray value over time taken from a single pixel,  $L_{in}$  is the number of frames, dashed squares indicate padding,  $N_B$  is the number of samples within a batch. For the convolution block shown, the number of channels are  $|C_{b-1}| = 1$ ,  $|C_b| = 4$ . (B) Architecture of the whole network.

To allow the computation of the first and last  $\Delta$  values, a padding was applied by extending the evolutions of the gray value by  $\Delta$  times the first and  $\Delta$  times the last value. In a second step, a threshold  $T$  was applied to the filter: all gradient values above the threshold were considered to be associated with cleaning. The threshold can be set either in form of an absolute or relative value. In the last step, the interval where cleaning is detected can be enlarged by applying a dilatation radius  $\Delta_d$ . Thus, the gradient filter procedure has three hyperparameters: the filter radius  $\Delta$ , the threshold  $T$ , and the dilatation radius  $\Delta_d$ . Applying this procedure, one filter was tuned for each experiment. The result of the filter can be easily evaluated using the new labeling view discussed in Section 2. Note that the gradient filter procedure does not detect which cleaning mechanism is present. This information must be provided manually.

### 3.2 | Fully convolutional networks

FCNs are a subfamily of convolutional neural networks (CNNs) dedicated to pixel-by-pixel segmentation of images [12]. CNNs are based on the convolution operation and during the training procedure, the filters are learned. For that reason, CNNs are well suited for the present task, since it is required to generalize a set of filters generated with the gradient filter procedure. Typically, CNNs consist of two parts: in the first part convolutional layers, pooling layers and activation functions are used to build features from the data. In the second part, the features are passed through fully connected layers, similar to a feed forward NN to receive an output for a certain task. Almost each CNN can be transformed in an FCN by replacing the second part with a convolutional layer and ensuring that the number of inputs matches the number of outputs [14].

The architecture used in the present paper only consists of convolution blocks and is illustrated in Figure 3. Pooling layers are omitted to keep the architecture simple. Each convolution block  $b$  has a set of channels  $C_b = \{1_b, 2_b, \dots\}$ . The data were passed batch-wise through the network, that means the data were partitioned in batches and the trainable parameters of the algorithm updated after every batch. After the whole training data are seen once by the algorithm, this is called an epoch. Within each convolution block  $b$ , three operations were used:

1. One-dimensional convolution
2. Batch normalization
3. Activation function

The result of the first step was calculated using

$$e_{c_b}^{n,s} = \sum_{c \in C_{b-1}} \sum_{k=1}^3 w_{c,c_b}^k a_c^{n-2+k,s}. \quad (3)$$

Herein,  $n$  is the frame index,  $s$  is the sample number within the present batch, and  $c_b$  the channel of interest. The first sum iterates over all channels belonging to the previous block, while the inner sum corresponds to the discrete cross-correlation operator. The upper bound, 3, equals the size of the kernel and could be any odd number. The weights  $w_{c,c_b}^k$  represent the filter kernels and are the trainable parameters of the FCN. Each block  $b$  contains  $3 \cdot |C_{b-1}| \cdot |C_b|$  trainable parameters, so that the number of parameters is independent of the input size. The values  $a_c^{n-2+k,s}$  represent the output of the channel  $c$  belonging to the previous block. If the sum is evaluated for the first block ( $b = 1$ ), the values  $a$  correspond to the input data only consisting of a single channel. For evaluating the sum at the first and last entry, the data were extended by applying zero padding. As a second step, batch normalization was applied, where the results from the first step were normalized with mean and standard deviation, which were computed considering the entire batch. In the third step, the data were passed through a nonlinear activation function. This is necessary to filter for particularly important features. In the present study, ReLU and tanh were tested, observing that the results are similar. The results shown later on were obtained with tanh. To ensure that only one output is generated per input, the final convolution block was configured with two channels representing cleaning and no cleaning, respectively. The result was finally passed through a softmax activation function to decide which label to assign. The rest of the network was built symmetrically. A employed number of convolution blocks was  $2N_b$ , with  $N_b$  the block depth. The number of channels within each block was computed as

$$|C_b| = \begin{cases} |C_{b-1}| \delta & \text{if } b \leq N_b, \\ |C_{b-1}| / \delta & \text{if } N_b < b < 2N_b, \\ 2 & \text{if } b = 2N_b. \end{cases} \quad (4)$$

The procedure just described involves the hyperparameters  $|C_1|$ ,  $N_b$ ,  $\delta$  defining the architecture. The FCNs were implemented in Python 3.11 using the PyTorch library.

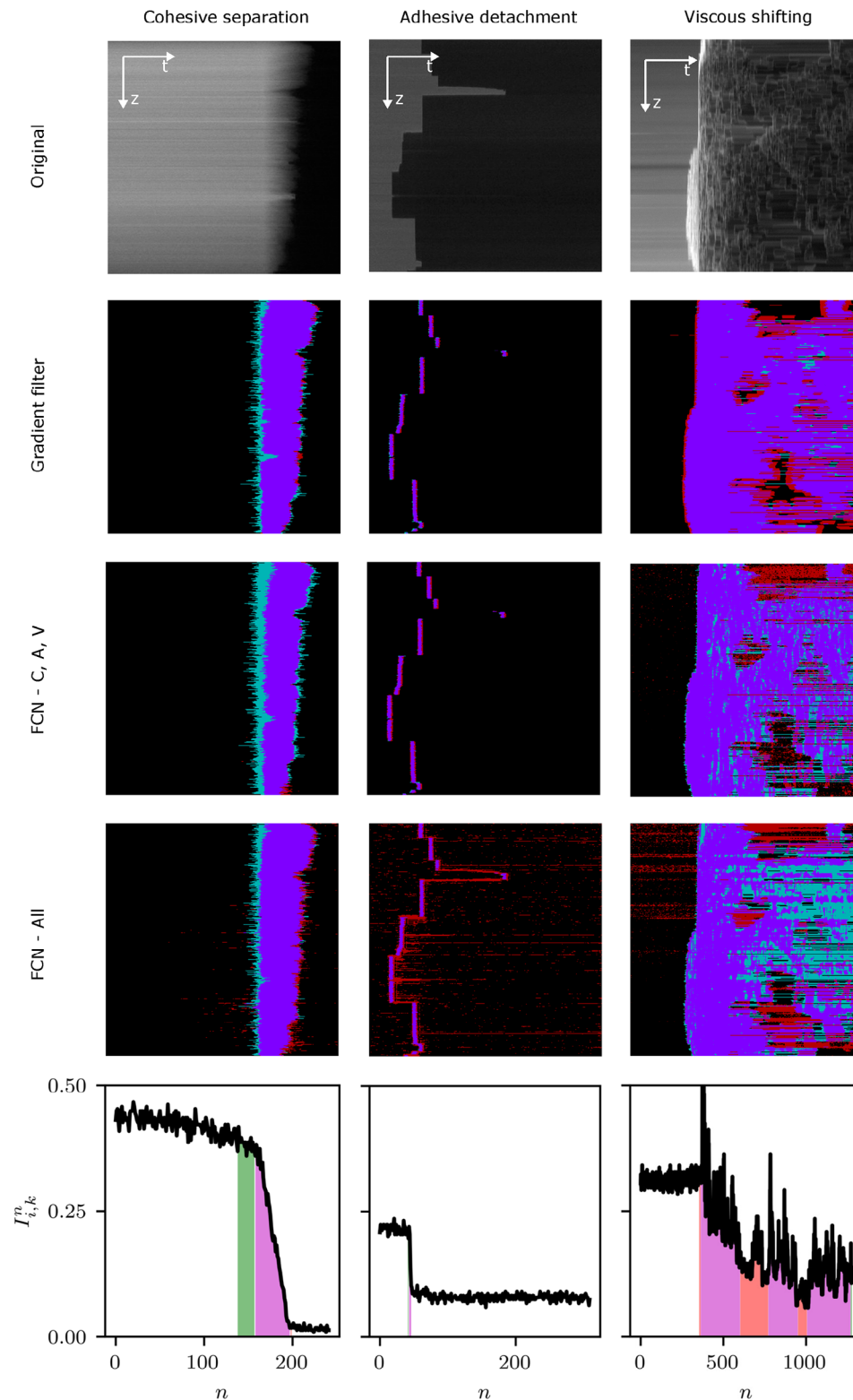
### 3.3 | Training

In total, four different FCNs were trained. One of the FCNs was trained with all nine experiments with the objective to identify cleaning in general (FCN-All). Each of the remaining FCNs was trained just to detect one single cleaning mechanism, each using three experiments (FCN-C, A, V). The training was conducted over a maximum of 100 epochs, while a batchsize of 10 000 was used and each experiment provides around 8000 samples. The samples were taken in a way that the length of the samples  $L_{in}$  was always the same, and the gray values were always normalized with the first value [7]. To avoid imbalances, class weights were assigned in a way that each class contributed equally to the loss, in total. Throughout the training, cross entropy loss was utilized. At the beginning of the training, the ADAM optimizer was used with an initial learn rate of 0.01. After each 30 epochs, the learn rate was reduced by a factor of 0.5. If the validation loss was not improving for 20 epochs, the optimizer was switched to SGD with a momentum of 0.9. This procedure provided better results. Every time the validation loss improved, the current model parameters were stored overwriting the recent parameters. If the validation loss did not improve for 30 epochs, the training was stopped early.

## 4 | RESULTS

### 4.1 | Gradient filter results

First, the gradient filter procedure was applied to the manually labeled experiments. An individual filter could be tuned for each experiment within a minute and the resulting classification mask is compared to the manual result in Figure 4. Quantitative comparison is done using the accuracy metric, which is defined as the number of correctly classified samples divided by the total number of samples classified. Since the samples for each class are not equally distributed, the inter-



**FIGURE 4** Comparison of the classification results. Green regions are only tagged manually, red regions are tagged only by the respective technique employed, and purple indicates overlapping regions, where both labelings are positive. First row: original image, second row: results obtained with gradient filter procedure, third row: results obtained by FCNs trained for each cleaning mechanism, fourth row: results obtained by FCN trained to detect cleaning in general, fifth row: sample evolutions of gray value over time with labeling obtained by gradient filter procedure.

**TABLE 1** Resulting accuracy and IoU metrics obtained by the different techniques employed.

	<b>Cohesive separation</b>		<b>Adhesive detachment</b>		<b>Viscous shifting</b>	
	<b>Acc.</b>	<b>IoU</b>	<b>Acc.</b>	<b>IoU</b>	<b>Acc.</b>	<b>IoU</b>
Gradient filter	0.96	0.82	0.99	0.71	0.87	0.81
FCN—C, A, V	0.80	0.70	0.99	0.68	0.87	0.82
FCN—all	0.78	0.79	0.94	0.20	0.72	0.58

section over union (IoU) metric was used additionally. The IoU metric compares the pixel mask obtained manually to the mask obtained by a procedure to be assessed. As the name indicates, it is defined as the ratio between the intersection and the union of these masks.

All metrics computed are listed in Table 1. For the gradient filter procedure, the lowest accuracy is obtained for viscous shifting and the highest accuracy for adhesive detachment. The accuracy metric is very good in all cases and Figure 4 confirms that misclassifications only arise by slight under- or overestimation of the time when the cleaning process starts or ends. For all cases, IoU metrics above 70% were achieved. To gain more insight into the quality of the classification result, an evolution of gray value over time was considered for each cleaning mechanism and the labelings were compared (Figure 4, last row). In case of cohesive separation, the labels obtained manually start earlier than the result of the filter. The example highlights that it is hard to tell when the cleaning process really starts. The advantage of the filter procedure is that it would rather introduce a systematic error, while manual labeling always involves random uncertainties. Systematic errors can be circumvented using a suitable post processing technique, for example, by applying a one-sided dilatation to the classification here. The example of viscous shifting is the most complex, since it might involve reattachment of the soil to the wall. In this case, using the filter procedure more regions are associated with cleaning than manually assigned.

## 4.2 | Fully convolutional networks

Finally, FCNs were considered. From inspecting the masks in Figure 4, only a slight difference between the results of the gradient filter procedure and the FCNs for each cleaning mechanism (FCN—C, A, V) is visible. In case of cohesive separation, the size of the green region at the beginning of the cleaning is increased, indicating a stronger deviation from the manual labels. Table 1 shows that the metrics only decreased in case of cohesive separation.

The FCN—All that is trained to detect cleaning in general has worse performance than previous candidates. Horizontal red stripes in Figure 4, fourth row, show that the FCN—All sometimes misclassifies samples further away from the cleaning process. In case of the mechanism cohesive separation the FCN—All benefits from more variation in the data achieving better metrics. The opposite effect occurs for detecting adhesive detachment. Here, image noise is misinterpreted as viscous shifting.

The present results can now be compared to the results obtained with the online network from the previous work [7]. In this reference, accuracies up to 80% were achieved when the algorithm was trained with 10–20 experiments for each cleaning mechanism and substantial time was invested to design appropriate features. Juxtaposing both results demonstrates that the metrics obtained with the present approach are promising. Especially when considering the smaller amount of training data.

## 5 | CONCLUSIONS

The present paper proposed an improved labeling strategy, which supports the data curation procedure for an existing classification scheme for cleaning mechanisms. The existing algorithm was analyzed and it was concluded that the level of detail of the training labels is the bottleneck for further enhancement of the algorithm and application towards realistic soils. Hence, an improved labeling strategy was developed, which is based on three entities: first, the perspective of investigation, which makes it easier to decide whether a labeling can be considered as good or bad. Second, the gradient filter procedure, which allows to obtain a fast and consistent labeling suggestion by only tuning three hyperparameters. Third, generalizing filters using FCNs.

While the FCN can be used as labeling assistance, they are also a promising architecture for a classification algorithm, which will be investigated in the future. In the present work, the FCNs only consider the gray value evolution over time of a single pixel. To also include macroscale information, multiple gray value evolutions will be considered. Furthermore, active learning strategies can be used to improve the performance of FCNs used for labeling assistance.

## ACKNOWLEDGMENTS

This research project is supported by Industrievereinigung für Lebensmitteltechnologie und Verpackung e.V. (IVLV), the Arbeitsgemeinschaft industrieller Forschungsvereinigungen “Otto von Guericke” e.V. (AiF) and the Federal Ministry of Economic Affairs and Climate Action (IGF 21334 BR).

Open access funding enabled and organized by Projekt DEAL.

## ORCID

Christian Golla  <https://orcid.org/0009-0007-5793-3169>

## REFERENCES

- Landel, J. R., & Wilson, D. I. (2021). The fluid mechanics of cleaning and decontamination of surfaces. *Annual Review of Fluid Mechanics*, 53, 147–171.
- Pettigrew, L., Blomenhofer, V., Hubert, S., Groß, F., & Delgado, A. (2015). Optimisation of water usage in a brewery clean-in-place system using reference nets. *Journal of Cleaner Production*, 87, 583–593.
- Rad, S. J., & Lewis, M. J. (2014). Water utilisation, energy utilisation and waste water management in the dairy industry: A review. *International Journal of Dairy Technology*, 67, 1–20.
- Alvarez, N., Daufin, G., & Gésan-Guizieu, G. (2010). Recommendations for rationalizing cleaning-in-place in the dairy industry: Case study of an ultra-high temperature heat exchanger. *Journal of Dairy Science*, 93, 808–821.
- Tsai, J.-H., Huang, J.-Y., & Wilson, D. I. (2021). Life cycle assessment of cleaning-in-place operations in egg yolk powder production. *Journal of Cleaner Production*, 278, 123936.
- Köhler, H., Liebmann, V., Golla, C., Fröhlich, J., & Rüdiger, F. (2021). Modeling and CFD-simulation of cleaning process for adhesively detaching film-like soils with respect to industrial application. *Food and Bioproducts Processing*, 129, 157–167.
- Golla, C., Freiherr Marschall, W., Kricke, S., Köhler, H., Rüdiger, F., & Fröhlich, J. (2023). Identification of cleaning mechanism by using neural networks. *Food and Bioproducts Processing*, 138, 86–102.
- Joppa, M., Köhler, H., Rüdiger, F., Majschak, J.-P., & Fröhlich, J. (2017). Experiments and simulations on the cleaning of a swellable soil in plane channel flow. *Heat Transfer Engineering*, 38, 786–795.
- Golla, C., Köhler, H., Liebmann, V., Fröhlich, J., & Rüdiger, F. (2022). CFD-based three-dimensional modeling of an adhesively detaching soil layer in a channel flow with sudden expansion. *Food and Bioproducts Processing*, 136, 176–183.
- Golla, C., Köhler, H., & Fröhlich, J. (2022). CFD-based modeling of a cohesively separating soil layer in consideration of local soil distribution. In *Proceedings of 14th International Conference on Heat Exchanger Fouling and Cleaning 2022*, Wagram, Austria.
- Köhler, H., Liebmann, V., Joppa, M., Fröhlich, J., Majschak, J.-P., & Rüdiger, F. (2019). On the concept of CFD-based prediction of cleaning for film-like soils. In *Proceedings of 13th International Conference on Heat Exchanger Fouling and Cleaning 2019*, Warsaw, Poland.
- Wang, W., Zhou, T., Yu, F., Dai, J., Konukoglu, E., & Van Gool, L. (2021). Exploring cross-image pixel contrast for semantic segmentation, arXiv:2101.11939 [cs.CV].
- Gonzales, R. C., & Woods, R. E. (2008). *Digital image processing* (Vol. 3). Prentice Hall.
- Long, J., Shelhamer, E., & Darrell, T. (2015). Fully convolutional networks for semantic segmentation, arXiv:1605.06211 [cs.CV].

**How to cite this article:** Golla, C., Boddin, L., Köhler, H., Rüdiger, F., & Fröhlich, J. (2024). Aspects of image data preparation to extend a classification scheme for cleaning mechanisms to realistic soils. *Proceedings in Applied Mathematics and Mechanics*, e202300142. <https://doi.org/10.1002/pamm.202300142>