water

MDPI

*Article*

# Foam Segmentation in Wastewater Treatment Plants

**Joaquín Carballo Mato *, Sonia González Vázquez *, Jesús Fernández Águila, Ángel Delgado Rodríguez, Xin Lin, Lucía Garabato Gándara, Juan Sobreira Seoane and Jose Silva Castro**

Instituto Tecnológico de Galicia, Cantón Grande 9, Planta 3, 15003 A Coruña, Spain; jfernandez@itg.es (J.F.Á.); adelgado@itg.es (Á.D.R.); xlin@itg.es (X.L.); lgarabato@itg.es (L.G.G.); jsobreira@itg.es (J.S.S.); jsilva@itg.es (J.S.C.)
* Correspondence: jcarballo@itg.es (J.C.M.); sgonzalez@itg.es (S.G.V.)

**Abstract:** The excessive accumulation of foam in wastewater treatment plant (WWTP) tanks can impede proper aeration, hindering the effective removal of organic matter from the water. This study proposes a novel technique to monitor in real time the presence of foams in WWTP tanks by using texture segmentation models trained with centralized and federated approaches. These models are designed to segment the foam and quantify the percentage of foam coverage across the entire tank surface. This data provides plant operators with crucial information for identifying the optimal time for foam removal. The proposed methodology is integrated into an image processing pipeline that involves acquiring images using a PTZ camera, ensuring the absence of anomalies in the captured images, and implementing a real-time communication method for event notifications to plant operators. The models exhibit noteworthy performance, achieving an 86% Dice score in foam segmentation, with comparable results obtained through both centralized and federated training. Implemented in a wastewater treatment plant, this integrated pipeline enhances operational efficiency while concurrently reducing costs.

**Keywords:** waste water treatment plants; semantic segmentation; federated learning; Deeplabv3+; One Shot Texture Segmentation; SDG6

check for updates

**Citation:** Carballo Mato, J.; González Vázquez, S.; Fernández Águila, J.; Delgado Rodríguez, Á.; Lin, X.; Garabato Gándara, L.; Sobreira Seoane, J.; Silva Castro, J. Foam Segmentation in Wastewater Treatment Plants. *Water* **2024**, *16*, 390. https://doi.org/10.3390/w16030390

Academic Editor: Min Ji

Received: 4 December 2023
Revised: 15 January 2024
Accepted: 18 January 2024
Published: 24 January 2024

**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

Texture segmentation plays an important part in image analysis and understanding. Basically, it involves the identification of regions with the same texture features, so that further analysis can be performed on the respective regions alone. An effective segmentation algorithm is very useful in many areas, such as industrial monitoring of product quality, medical image analysis, or image retrieval.

This paper focuses on the analysis of images obtained from wastewater treatment plants. In these plants, various processes are carried out to facilitate the removal of organic matter, nitrification, denitrification of the influent, as well as the removal of suspended solids.

In the case of plants such as the ones used in this research, the removal of organic matter and nitrogen in the MBBR (Moving Bed Biofilm Reactor) [1] is achieved through a biological treatment using AnoxKaldnes technology. AnoxKaldnes™ technology is based on the growth of biomass (in the form of a biofilm) on continuously moving plastic media in the biological reactor. These media are small in size but have a high specific surface area per unit volume, allowing for a high level of contact between the wastewater and the biofilm, thus allowing the biofilm to consume the organic matter of the wastewater.

In the event of any abnormal chemical spills in water, aeration issues or the accumulation of suspended solids, large foams can be formed in the tanks of biological treatments [2]. The foams have to be removed because they can cover the tanks causing serious operational problems and preventing proper aeration. Currently, this activity is done manually in the wastewater treatment plant considered in this research, leaving aside any metric that indicates the most appropriate time when defoamers should be used.

Foam analysis has traditionally been performed in a manual manner in different domains of study. However, manual measurement of some foam and liquid properties can be time-consuming, and certain important geometric and dynamic measures are difficult or impossible to quantify using manual methods.

In the work by Collivignarelli et al. [2], two on-site foam measurement methods, namely Foam Surface Covered (FSC) and Foam Volume (FV), were described. It is important to note that these methods come with certain limitations, such as the requirement for the camera to be positioned orthogonally to the tank or the necessity of employing devices such as hydrometers to calculate the foam volume.

In the study by Wang et al. [3], a method based on texture analysis was proposed to effectively segment liquid from foam. Additionally, the approach aims to identify the boundaries of individual bubbles within the foam layer. Furthermore, alternative techniques for foam segmentation based on the watershed method were explored in Refs. [4,5]. Forbes and de Jager [4] presented a novel approach that combines a texture measure with two stages of watershed. This method has been demonstrated to facilitate the segmentation of images containing both large and small bubbles. In the context of Ref. [5], various implementations of the watershed method are presented for both semantic and instance segmentation. However, it is important to note that these papers primarily concentrate on foams formed by relatively large bubbles, which differ from the characteristics of the foams observed in the wastewater treatment plant tanks that will be the focus of our investigation.

In Ref. [6], an image processing method is presented for quantifying foam coverage across the entire surface of tanks in sewage plants. However, the paper notes that the effectiveness of this method may be influenced by complex environmental factors.

Given the limitations observed in classical methods documented in the literature, it was opted for the application of deep learning methods due to their versatility in handling images with diverse characteristics.

To the best of our knowledge, there are currently no deep learning algorithms specifically tailored for foam segmentation. Therefore, the decision was made to employ models created to solve a similar problem, specifically, texture segmentation.

Various deep learning algorithms for segmenting and classifying textures in different domains are documented in the literature. Typically, these models utilize classic Convolutional Neural Networks (CNN) for image segmentation to extract characteristic textures. For instance, in Ref. [7], diverse deep learning models, such as AlexNet, VGG16, or ResNet34, are utilized to identify various diseases in tomato leaves. The prompt identification and timely treatment of these diseases can mitigate potential losses in tomato production. Similarly, in Ref. [8], convolutional neural networks are applied to segment breast density in mammographic imaging.

This way, in Ref. [9], an energy efficient system based on deep convolutional neural networks for early smoke detection in both normal and foggy environments is created. The proposed architecture is a VGG-16 pre-trained in Imagenet. Moreover, this model is fine-tuned in a new dataset, created by the authors of the paper, consisting of 72,012 images from 4 different classes: nonsmoke, smoke, nonsmoke with fog, and smoke with fog. In 2021, a new work was published [10], where they present a CNN-based smoke detection and segmentation framework for both clear and hazy environments. Regarding the detection an EfficientNet architecture is introduced, which outperforms the results acquired in Ref. [9]. Concerning the semantic segmentation of the smoke regions, a DeepLabv3+ model is used. The classification and segmentation models are tested on the dataset presented in Ref. [9], obtaining better metrics than previous models. Finally, in Ref. [11] a highly adaptable model was introduced for texture segmentation, namely One Shot Texture Segmentation (OSTS), which does not require fine-tuning on specific problems or a large amount of annotated data for training. The key feature of the proposed architecture is the utilization of two input images: the image to be segmented and an image of the desired texture to be segmented. Additionally, this model is trained on a novel dataset named CollTex, created from texture images sourced from the Describable Texture Data

(DTD) (https://www.robots.ox.ac.uk/~vgg/data/dtd/ (accessed on 2 December 2022)). The achieved results surpass 90% accuracy on images composed of two textures.

Recognizing potential challenges in acquiring a sufficient quantity of images with adequate variability from individual wastewater treatment plant, the feasibility of employing federated learning instead of a centralized approach was explored to keep data privacy and increase communication efficiency [12]. An analysis of different aggregation methods and major libraries was conducted, examining their functionality, advantages, and disadvantages.

The most commonly used aggregation method is FedAvg (Federated Averaging), which involves alternatively selecting a subset of client nodes in each iteration and applying a weighted average of the local models to update the global model. Another prominent method is FedSGD (Federated Stochastic Gradient Descent), which averages the gradients in each round.

As for the main frameworks or libraries, Flower (https://flower.dev (accessed on 7 November 2022)), NVIDIA Flare (https://developer.nvidia.com/flare (accessed on 9 November 2022)), TensorFlow Federated (https://www.tensorflow.org/federated (accessed on 8 November 2022)), and IBM Federated Learning Framework (https://www.ibm.com/docs/en/cloud-paks/cp-data/4.7.x?topic=models-federated-learning (accessed on 8 November 2022)) stand out. These libraries, except for NVFlare, are known for their user-friendly nature, making them highly attractive tools for research purposes. However, their commercial use is limited due to the lack of security they provide. NVFlare, on the other hand, is specifically designed for developing commercial products and addresses these security concerns.

The fundamental purpose of this initiative is to achieve a more efficient management of the WWTP. Therefore, our main contributions are the following:

- Implement innovative solutions with the aim of reducing costs and improving efficiency. To do so, we introduced a computer vision system that enables real-time monitoring of foam parameters, ensuring timely and accurate treatment interventions;
- Comparison between centralized and federated data treatment.

To achieve the objectives, two foam segmentation models, DeepLabv3+ [10] and OSTS [11], have been implemented and trained using images from a real WWTP. The former was selected due to the power of DeepLabv3+ segmenting smoke and the similarity between smoke and foam. The latter will contribute with its capacity to get good performance on small training datasets.

The foam percentage on the tank surface can be calculated from the segmented foam. This is crucial for determining the appropriate time to remove excess of foam, therefore enhancing the response speed to foam accumulations.

Moreover, the foam segmentation methodology is part of a complete computer vision pipeline that includes image capture, the selection of the suitable images for further processing, and the transmission of operationally significant data to the responsible personnel at the WWTP.

Regarding federated learning, the possibility of working with multiple clients has been explored in order to create a flexible model that works in different environments maintaining the privacy of each client's data and reducing data communications compared to centralized training.

This study surpasses prior research by leveraging powerful and highly flexible models, utilizing datasets composed of a large number of images with diverse characteristics, and offering a complete remote sensing system based on image analysis to monitor foam coverage on WWTP tanks, which ranges from the automatic and periodic image capturing process to the sending of alerts to WWTP operators.

The rest of this work is organized as follows. Section 2 provides the datasets that are used throughout the work and describes the proposed methodology. The experimental evaluations and the characteristics of the deployment are described in Section 3. Finally, this work is concluded in Section 4 with the key findings and future directions.

## 2. Materials and Methods

### *2.1. Datasets*

In response to the unavailability of publicly accessible datasets of foams and the encountered delays in procuring images from the wastewater treatment plant where the investigation was conducted, the project was developed in three stages.

In the first stage, public datasets of other domains were used. The first dataset, named DeepSmoke dataset, is described in Ref. [10]. It comprises 18,532 smoke images, 18,532 foggy images with smoke, 17,474 smoke-free images, and 17,474 foggy images without smoke. One image with smoke and one image with smoke and fog are shown in Figure 1. The use of this dataset is due to the similarity between smoke and foam textures.



(**a**) (**b**)

**Figure 1.** Images from DeepSmoke dataset: (**a**) Image with smoke. (**b**) Image with smoke and fog.

The second dataset [11], named CollTex, is composed of collages formed with textures (Figure 2).



**Figure 2.** Two images from OSTS dataset.

However, since the problem is the segmentation of foam, and in order to make the dataset resemble the images that will be captured from the tanks of the WWTP as closely as possible, the dataset was slightly modified in order to create images with only two randomly distributed textures throughout the image. To create these new collages, 50 random centroids were selected in the image and we associated each of these points with one of the two arbitrarily chosen textures. Subsequently, the remaining points were assigned to the nearest centroid (Figure 3).

In the second stage of the project, since a camera had not yet been fixed in the WWTP, a series of images were manually captured at the WWTP. We will refer to this dataset as "Manual Dataset". These images were captured using a PTZ camera (Hanwhavision XNP-8250: https://hanwhavision.eu/es/producto/xnp-8250/ (accessed on 5 October 2022)) over a specific day, for an hour, and under particular lighting conditions. In particular, 383 images of size 3328 × 1872 were captured from a unique tank from 12 a.m. to 1 p.m. Figure 4 shows some of the captured images.



(**a**) (**b**) (**c**)

**Figure 3.** Adaptation of the OSTS dataset to generate a collage with only two textures: Given two textures from the original dataset represented on (**a**,**b**); the new collage (**c**) is formed by selecting 50 random centroids and associating each centroid to one of the textures.



**Figure 4.** Two images included on manual dataset.

In the third stage of the project, a set of 5 images every 10 min were captured, with the same PTZ camera used to generate the manual dataset, in 2 tanks of the WWTP with a resolution of 1920 × 1080 from 7 a.m. to 8 p.m. from February to May. We will refer to this dataset as the "Final Dataset".

Figure 5 shows two views taken with the PTZ camera installed in the plant, each centered on one of the two tanks considered in this dataset.

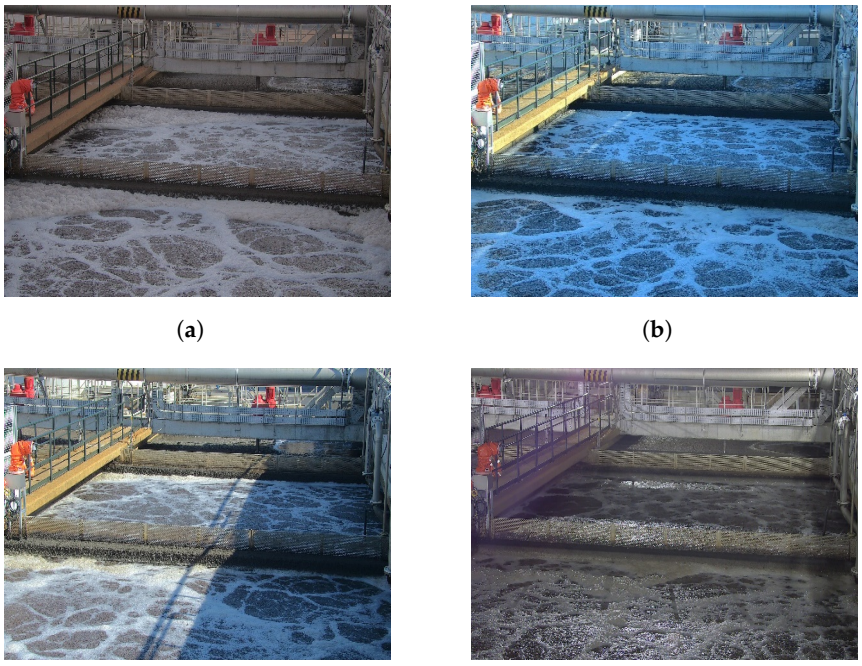


(**a**) (**b**)

**Figure 5.** Tanks of the WWTP used in the research: (**a**) Tank 1. (**b**) Tank 2.

There are two differences between the images of the manual and final datasets. The first difference is the variability in lighting conditions throughout the day. This resulted in images with different colors, shadows, varying levels of darkness, or images that change depending on the weather conditions. The second difference is the resolution due to storage limitations. Nevertheless, the structure of the foam texture is clearly seen in both types of images.

Figure 6 shows an example of images captured at different times on 4 March 2023, showcasing the variations in lighting conditions and visual appearance.



(**a**) (**b**)

(**c**) (**d**)

**Figure 6.** Images from final dataset: (**a**) Image captured at 8 a.m. (**b**) Image captured at 10 a.m. (**c**) Image captures at 12 a.m. (**d**) Image captured at 8:45 p.m.

### *2.2. Labeling*

As the segmentation models used in this work are supervised models, it is necessary to have the ground truth segmentation of the images.

To simplify the annotation process of the manual dataset, an image processing algorithm was developed in order to generate the initial masks to be used in the training. The steps involved, as shown in Figure 7, are as follows:

(1) A homography transform was applied to focus on the relevant part of the image;
(2) The image was converted from RGB to HSV, and the first channel, which defines the hue of the image, was extracted;
(3) An Otsu's thresholding method was applied to retain only the areas of the image where plastic supports were not present;
(4) The original image was converted to greyscale and masked using the result obtained in step 3;
(5) Adaptive thresholding was applied to the resulting image from step 4. This method performs thresholding considering a small number of neighbouring pixels. Adaptive thresholding avoids issues that may arise from global thresholding in images with shadows or varying colour intensities across different areas of the image;
(6) To refine the mask, two morphological operations, opening and closing, were performed. These operations aim to eliminate spurious pixels.

Finally, the masks were sent to experts for further refinement.

The algorithm previously presented for mask creation cannot be directly applied to the images from the final dataset because it was specifically designed for images captured under certain lighting conditions. The new algorithm involves directly applying adaptive thresholding, followed by opening and closing operations to remove spurious pixels and refine the masks.

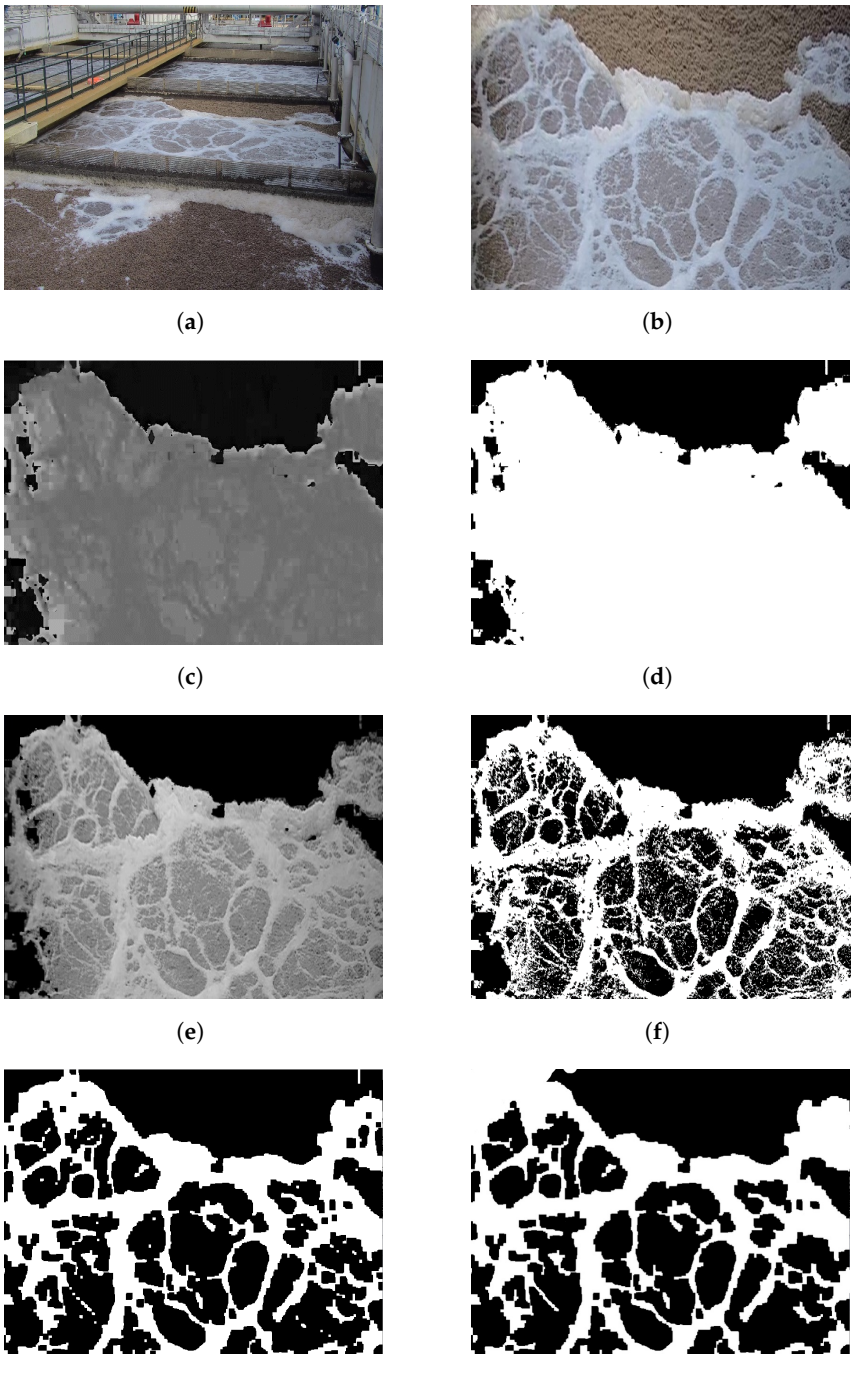


**Figure 7.** Labelling process to get the ground truth mask of images from the manual dataset: (**a**) Original image. (**b**) Step 1: Homography. (**c**) Step 2: H channel of the homography image in HSV. (**d**) Step 3: Thresholding of the step 2 image. (**e**) Step 4: Application of the step 3 mask to the step 1 image in greyscale. (**f**) Step 5: Adaptive thresholding of the step 4 image. (**g**) Step 6: Opening + closing of the step 5 image. (**h**) Step 7: Mask refined by an expert.

Figure 8 shows an example of an image with and without shadows and its corresponding mask calculated using the new algorithm. This demonstrates that the algorithm performs sufficiently well under different lighting variations.

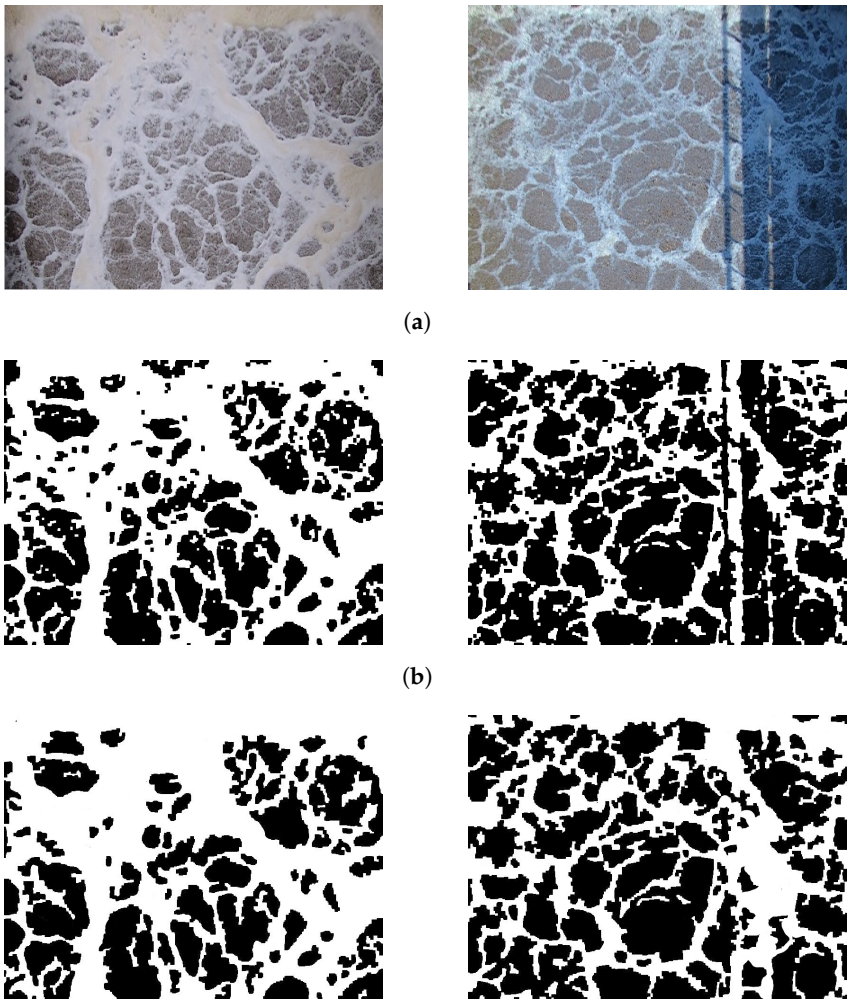


**Figure 8.** Labelling process to get the ground truth mask of images from the final dataset: (**a**) Two homographies of images with and without shadows. (**b**) Initial segmentation masks of both images. (**c**) Masks of both images refined by an expert.

### *2.3. Architecture of Models*

To address the foam segmentation problem, two deep learning models have been analysed. The use of these models is due to their power segmenting similar images to the ones from the WWTP and their capacity working with small texture datasets.

#### 2.3.1. One Shot Texture Segmentation

This model, presented in Ref. [11], is based on obtaining a numerical representation of the image to be segmented and the reference texture (the texture to segment in the image), and subsequently comparing which parts of the image contain the reference texture. The main advantage of OSTS with respect to other models is the use of two input images, the collage to be segmented, and the reference texture to segment, which enables the model to achieve better texture segmentation. In addition, OSTS is well known by its power to work well with small datasets. The architecture of the model, presented in the Figure 9, is divided in four parts:

(A) The first part consists of a VGG architecture, which aims to obtain a numerical representation of both the image to be segmented and the reference texture;
(B) The second part is an encoder, designed to ensure that the numerical representation obtained in step A has the same size as the original image. This way, the feature

vector at each spatial position can be considered as a representation of the texture at that position;

(C) In the third part, the numerical representations of the image and reference texture are compared to determine which parts of the image contain the reference texture. The result is an image of the same size as the original image, highlighting the positions with a texture similar to the reference texture;

(D) Finally, a decoder is employed to refine the segmentation.

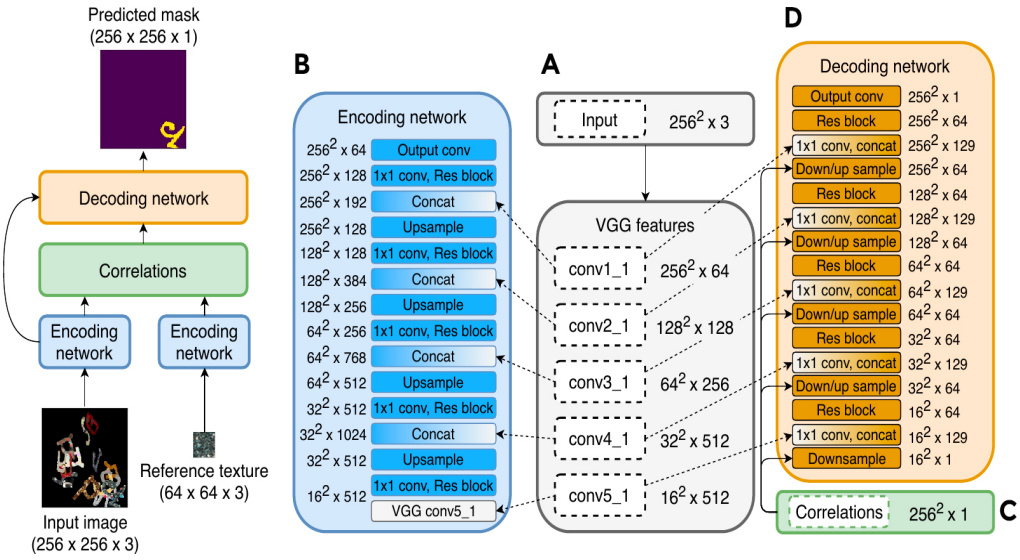



**Figure 9.** Architecture of the model OSTS. Source: Ref. [11].

### 2.3.2. DeepLabv3+

In Ref. [10], the DeepLabv3+ architecture (Figure 10) is employed to address the smoke segmentation problem. This model is based on an encoder-decoder architecture and its main feature is the use of atrous convolutions at different scales. Atrous convolutions are crucial as they reduce the number of parameters in the filters while increasing the receptive field. Additionally, due to the model's flexibility, various encoders can be used. As shown in the results section, in this work the following encoders have been tested:

- MobiletNet [13]: a lightweight network designed for efficient inference on mobile devices;
- ResNet [14]: Residual Neural Network, which introduces skip connections to overcome the vanishing gradient problem;
- HRNet [15]: a deep learning model that addresses the limitations of traditional convolutional neural networks by simultaneously maintaining high-resolution representations at different scales.

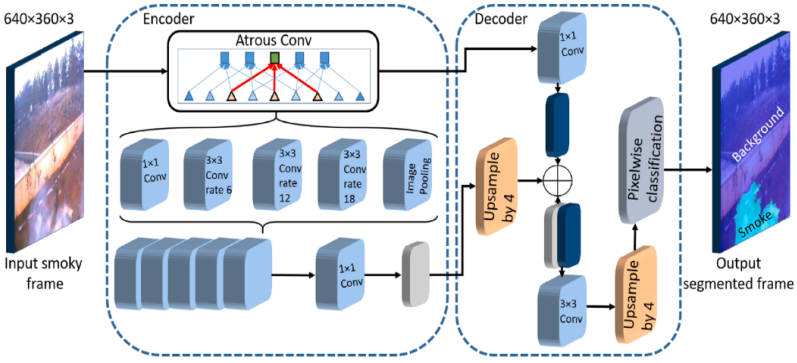



**Figure 10.** Architecture of the model DeepLabv3+. Source: Ref. [10].

The flexibility of the model, together with the reduced number of parameters due to the use of atrous convolutions, makes this architecture outperform previous convolutional models.

### *2.4. Federated Learning*

Federated Learning (FL) emerges with the objective of training a machine learning model on multiple local datasets stored on decentralized peripheral devices without exchanging data samples.

In this privacy-preserving approach, a server typically receives parameters (e.g., gradients or neural network weights) from locally trained models on decentralized peripheral devices and averages these parameters to construct a global model. Subsequently, the averaged parameters of the global model are sent back to the peripheral devices to update their local models. This process is repeated until the global model converges or a stopping condition is met. In this research project, the FedAvg (Federated Averaging) algorithm has been chosen as the aggregation method for parameter averaging. In Figure 11 it is graphically explained how a federated architecture works when having two nodes.

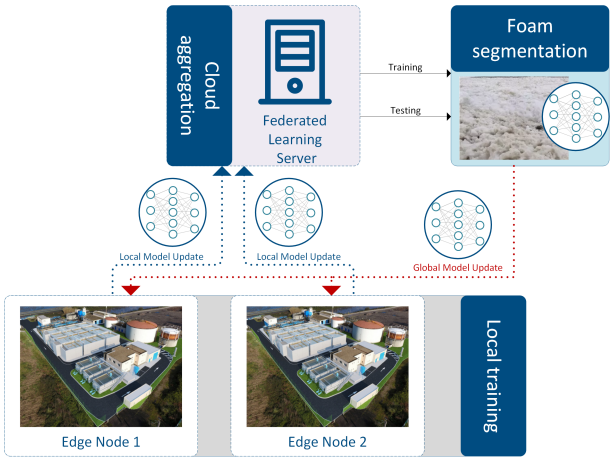



**Figure 11.** Architecture of federated learning.

The application of federated learning in foam segmentation of WWTPs tanks enables the utilization of images from diverse WWTPs. This allows for a larger number of images, encompassing varying lighting conditions and shadows, thereby enhancing the robustness of the trained models. In this research, the training has been conducted on two tanks of the same WWTP as if they were two separate WWTPs. This approach was adopted due to the impossibility of involving another WWTP using the same biological treatment method in the project.

### *2.5. Implementation Details*

Regarding the models, their implementation was performed using Python 3.8 as the programming language and PyTorch 1.12.1 as the deep learning framework. Moreover, the training of the models was conducted on a server equipped with an NVIDIA RTX A6000 GPU.

Furthermore, NVIDIA Flare has been selected as the library for federated learning. This choice is based on its design for developing commercial software, ensuring strong security in the communications between the server and clients.

## 3. Results

In this section, the results obtained from various experiments conducted are described.

First, the outcomes of the model training phase employing publicly available texture datasets are delineated. Subsequent to this, the pretrained models derived from the aforementioned step are retrained using the manual dataset. Then, the training results obtained

using the final dataset are summarized. Lastly, an exhaustive comparison between the two evaluated training paradigms, namely centralized and federated, is expounded upon.

Regarding the evaluation metrics, Dice and IoU scores are utilized to assess the performance:

- IoU is the area of overlap between the predicted segmentation and the ground truth divided by the area of union between the predicted segmentation and the ground truth. This metric ranges from 0 to 1 (0–100%) with 0 signifying no overlap and 1 meaning perfectly overlapping segmentation;
- The Dice metric is defined as twice the Area of Overlap divided by the total number of pixels in both images. It is a metric equivalent to the F1-score in classification problem, so it balances the precision and the recall.

Both metrics presented are positively correlated, so if classifier A is better than B under one metric, it is also better than classifier B under the other metric. However, the difference between the two metrics is that the IoU penalizes under -and over- segmentation more than Dice.

### *3.1. Training with Public Texture Datasets*

Since the manual dataset is formed by a limited number of images with little variation in terms of lighting conditions, firstly the models were trained using public datasets to further perform transfer learning to the manual dataset.

#### 3.1.1. OSTS

The model OSTS was trained using 2000 images of collages as presented in Figure 3, where 1500 form the train set, 250 were used for validation, and 250 for testing. Regarding the training parameters, the model was trained for 29 epochs, which took approximately 1 h and 15 min.

The metrics on the test set are shown in Table 1. In addition to the metrics, Figure 12 presents the segmentation results for three images from the test set.

**Table 1.** Metrics of OSTS in the modified public dataset CollTex.

| | **Test** |
|---|---|
| Dice | 0.963 |
| IoU | 0.930 |

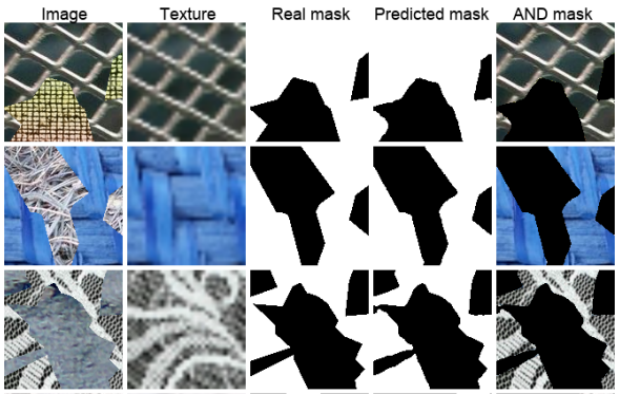



**Figure 12.** Results of the model OSTS in CollTex dataset. The first column corresponds to the image to be segmented, the second column displays the reference textures to be segmented, the third column shows the mask, the fourth column represents the texture segmentation, and the last column shows the intersection (AND) between the segmentation and the original image.

#### 3.1.2. Deeplabv3+

The DeepLabv3+ model was trained using 395 images of size 128 × 128 from the DeepSmoke dataset presented in Ref. [10]. Out of these images, 75% were used for training, while the remaining 25% were divided between validation and testing. The model underwent 38 epochs of training, which lasted approximately 11 min.

This model allows the use of different encoders. In order to find the model that provides the best results, the metrics on the test set were compared using MobileNet, ResNet50, and HRNetv2-32 as encoders (Table 2). Although the metrics for the different models are similar, ResNet50 was chosen as the encoder due to its simplicity.

Figure 13 displays the segmentation results on the test set using this model.

**Table 2.** Metrics of DeepLabv3+ using different encoders in the public dataset DeepSmoke.

| | **MobileNet** | **ResNet50** | **HRNetv2** |
|---|---|---|---|
| Dice | 0.926 | 0.961 | 0.948 |
| IoU | 0.862 | 0.820 | 0.746 |

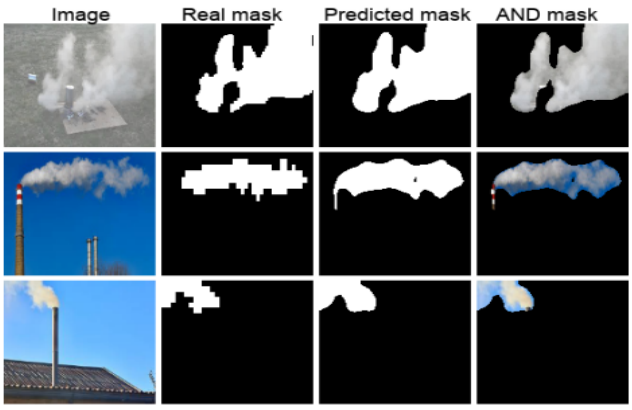



**Figure 13.** Results of the model DeepLabv3+ in the DeepSmoke dataset. The first column corresponds to the image to be segmented, the second column represents the mask, the third column shows the model's prediction, and the fourth column displays the overlay of the original image and the model's prediction.

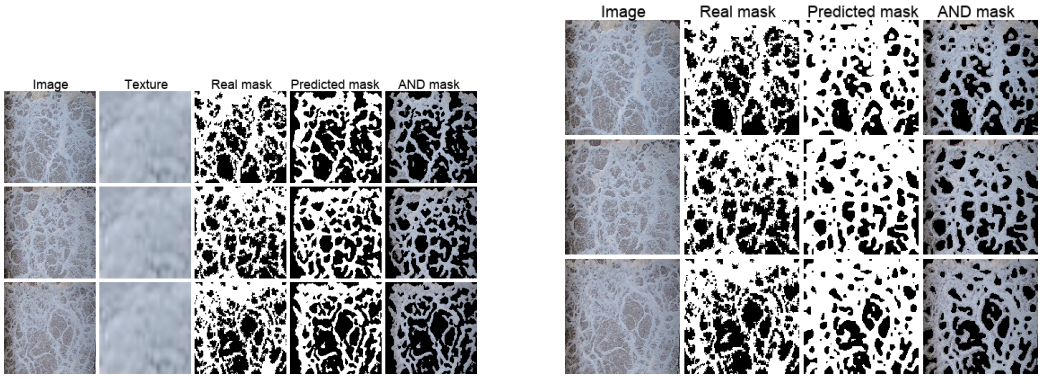
### *3.2. Transfer Learning to Manual Dataset*

Both the OSTS and DeepLabv3+ models were retrained with the manual dataset. The results obtained in the test set are shown in Table 3.

**Table 3.** Metrics of OSTS and DeepLabv3+ applying transfer learning in the manual dataset.

| | **Dice** | | **IoU** | |
|---|---|---|---|---|
| | **OSTS** | **DeepLabv3+** | **OSTS** | **DeepLabv3+** |
| Test | 0.747 | 0.842 | 0.596 | 0.727 |

The segmentation results for three different images are shown for both the OSTS and DeepLabv3+ models (Figure 14). These results show good metrics for both models, DeepLabv3+ achieving relatively better results in either IoU and Dice. Moreover, although metrics can be improved, qualitative results are fairly good since it is very difficult to decide what is or is not foam.




(**a**) (**b**)

**Figure 14.** Resulting images after applying transfer learning in the manual dataset: (**a**) Results after applying OSTS model. (**b**) Results after applying Deeplabv3+ model.

### *3.3. Training with the Final Dataset*

The installation of the PTZ camera at the WWTP made it possible to capture a sufficient number of images to train the models without the need for transfer learning. Additionally, the variability in lighting conditions (Figure 6) throughout the day allows for training a robust model capable of handling changes in image brightness. Having such diverse images (with varying lighting, shadows, and different weather conditions such as rain or wind) enables the construction of a more flexible model. However, it also poses challenges in creating a more powerful predictive model.

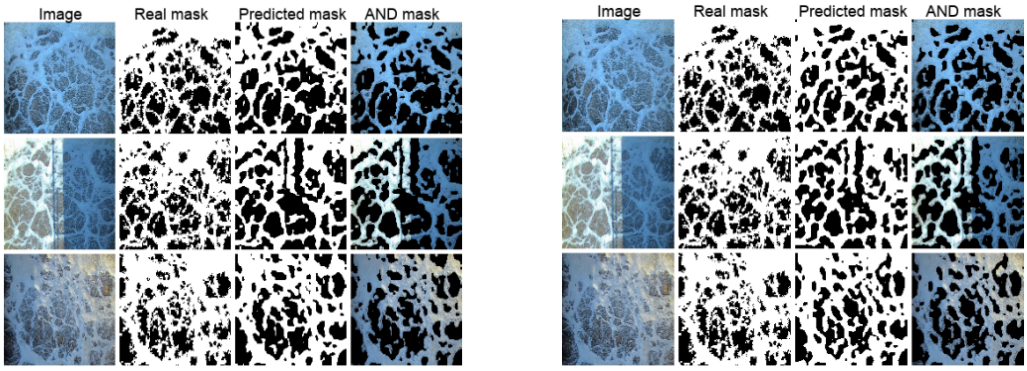
To train the best predictive model, images from both tanks of the WWTP were utilized. Specifically, the training process involved 2000 images from each tank, with 1500 images used for training, 250 for validation, and 56 for testing.

Regarding the model, DeepLabv3+ was chosen. This decision was made because, although the OSTS model is also suitable for segmenting images without variations in lighting conditions, the challenge of selecting a valid reference texture for all types of images makes it less effective for the entire image set.

Table 4 and Figure 15a display the metric results obtained by DeepLabv3+ in the test set, as well as the application of the trained model to three images from the test set, respectively. The obtained metrics are good, but the variability of lighting conditions in the image dataset achieved higher scores. Moreover, it is important to point out that the differences between IoU and Dice metric in the final dataset is lower than when doing transfer learning. This is due to the fact that when doing transfer learning, more instances of bad segmentation appear and IoU tends to penalize this cases. Furthermore, it can be observed in the test set images that it is not always clear what is and what is not foam, which in general makes this segmentation problem difficult to solve.

**Table 4.** Metrics of DeepLabv3+ in the final dataset. Comparison between metrics in centralized and federated training.

| | **Centralized Training** | **Federated Training** |
|---|---|---|
| Dice | 0.857 | 0.862 |
| IoU | 0.751 | 0.757 |




(**a**) (**b**)

**Figure 15.** Resulting images of training with the full dataset: (**a**) Results of the centralized training. (**b**) Results of the federated training.

### *3.4. Federated Training*

Regarding the model parameters, two clients were selected, each containing the data from one of the tanks in the WWTP. In this case, similar to centralized training, each client was trained with 2000 images.

As shown in Table 4, the results are similar to those obtained in centralized training. This closeness can be attributed to the similarity of the datasets from both clients. Results of the application of the federated model to the final dataset can be seen in Figure 15b.

The parameters used in the different training processes are as follows: an image size of 128 × 128, a training of 200 epochs with early stopping of 10 epochs without improving the validation loss, a learning rate of 0.001 with step size of 0.1 each 20 epochs, a batch size of 32, and Adam as optimizer.

### *3.5. Deployment*

The deployment process was conducted on a server equipped with an NVIDIA GeForce RTX 3060 GPU. It consisted of four steps: image capture, validation of captured images for correctness or anomalies, prediction of the percentage of foam over the total surface area of each tank, and sending the data and images to the event server to store and monitor the foam measurement in the tanks and notify the WWTP operator in case the amount of foam exceeds a certain threshold (Figure 16).

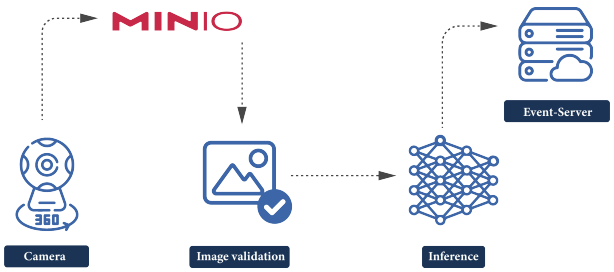



**Figure 16.** Steps in the deployment of the project.

#### 3.5.1. Capture of Images

The image capture process is set to acquire 5 images from each tank of the WWTP every 10 min. The captured images are stored in a MinIO (https://min.io/ (accessed on 17 January 2023)) bucket for further processing by the vision algorithms. The acquisition time for 5 images from each of the 2 camera positions capturing each tank is 24 s.

#### 3.5.2. Image Validation against Anomalies

Method

The purpose of introducing anomaly detection algorithms in the preprocessing stage is to identify images that do not correspond to the expected reality. This can be due to problems with the position of the PTZ camera, the presence of unexpected elements nearby tanks, such as people or seagulls, and cyberattacks. Three types of checks are performed to determine if an image is anomalous or not:

(i) An image is considered anomalous if its size differs from the defined one;
(ii) An image is considered anomalous if it is exactly the same as the previous image. To verify if two images are exactly the same, the Mean Square Error (MSE) between them is calculated. A value of zero indicates that the images are exactly identical. This method helps detect possible "man-in-the-middle" attacks that provide fake images through loops or repetitions;
(iii) An image is considered anomalous based on similarity if its similarity percentage with a reference image falls below a threshold.

To further understand the possible anomalies, a more in-depth explanation of the checks is provided.

The first two checks are straightforward. In case of the third, given a non-anomalous reference image, four areas or patches have been selected as reference zones (marked with a green rectangle in Figure 17). Then, to check if an image is anomalous based on similarity, the Learned Perceptual Image Patch Similarity (LPIPS [16]) is calculated between the reference areas of the reference image and the reference areas of the newly captured image. What LPIPS does is to calculate the similarity between the activations of two image patches given a predefined neural network. One advantage of this measure is that it has been shown to align well with human perception. In this case, a VGG neural network is used as the reference network. Moreover, in order to make the method robust to certain occlusions, only the top 3 patches with the highest similarity to the patches of the reference image are considered.



**Figure 17.** Image of the WWTP showing the patches used for anomalous image detection (marked in green).

Regarding the algorithm for detecting anomalies, it is based on comparing the newly captured image from the PTZ camera with a reference image. The reference image is set as the first image captured on the previous day. Subsequently, if the first image captured today matches the reference image, it becomes the new reference image. The second image captured today is compared to the first image of the day, provided that it was classified as correct. If the second image of the day is also deemed correct, it becomes the new reference image, and so on. If an image capture is detected as anomalous, it is not selected as the new reference image, and the last image detected as correct remains the reference. The objective of this process is to compare a new image with the most recent reference image to mitigate the influence of lighting changes.

Validation

In order to validate the method, a dataset consisting of 47 images, including both correct and anomalous images, has been created. The anomalous images encompass variations in size, repeated images, and discrepancies in similarity compared to the reference image patches.

Since the first two cases are trivial, the focus will be on the anomalous images based on their similarity to a reference image. Thus, several investigated cases are presented as examples:

(i) Changes in the camera parameters: the reference image was captured with the following camera parameters: pan is set to 281, tilt to 9.81, and zoom to 2.2 (Figure 17). To simulate different anomalous images, the images were captured with varying parameter values. For the pan parameter, images were captured in the range 277–285. Regarding tilt, the dataset contains images from tilt equals to 5.81 to tilt equals to 13.81. As for the zoom parameter, images range from zoom equal to 1.3 to zoom equal to 3.2. Some examples are shown in Figure 18, with the differing parameter value highlighted in bold;
(ii) Occlusions in the received images: to validate the algorithm's robustness against certain occlusions, we will refer to Figure 19a, where one of the patches is occluded by a bird. Since the method considers only the top 3 patches with the highest similarity to a reference image, the occlusion of a patch will not affect the accurate prediction of an image as anomalous or non-anomalous;
(iii) Images completely different from the expected ones: to validate this point, a completely different image from the expected ones, presented in Figure 19b, was introduced, which was detected as anomalous by the applied method.

Once the LPIPS values were calculated for the validation image set, a threshold of 0.3 was set to classify an image as anomalous or non-anomalous. By using this threshold, all images in the validation set were correctly classified as either anomalous or non-anomalous.

#### 3.5.3. Inference

Once the images are captured and verified to be correct, the segmentation process is performed using the corresponding model. The resulting output includes the segmented image, the original image, and the foam percentage over the total surface area of the tank. Figure 20 displays the segmentation results for images captured under different atmospheric conditions, showcasing the model's effectiveness in each case.

The time required for the anomaly detection and performing inference is 17 s for every set of 5 images.

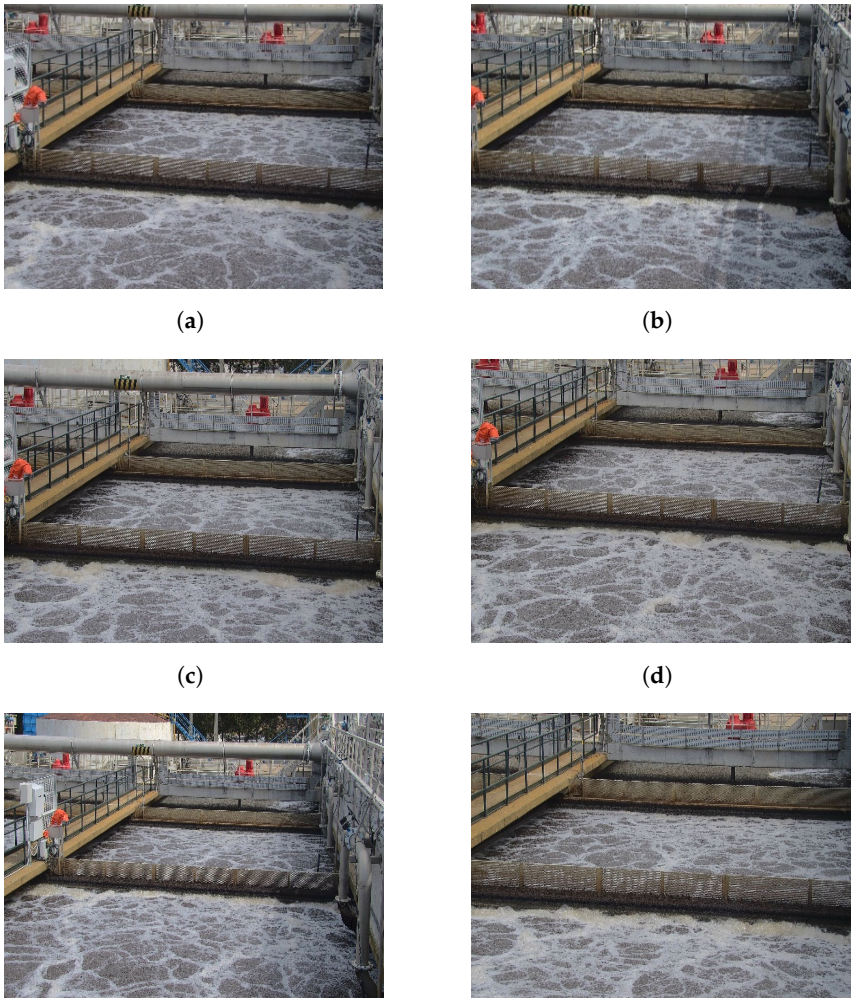
#### 3.5.4. Data Submission to the Event-Server

The Event-Server, or event consumer, serves as a global repository for storing images and alphanumeric values resulting from the application of various AI models. Additionally, the Event-Server is capable of processing these events to generate, record, and send notifications to end users if the event inferred meets the established criteria based on user-defined parameters for a relevant event.

Regarding anomalies, three types of events are being sent based on the detected anomaly: repeated image, incorrect image size, and incorrect camera position or erroneous image. As for inference, an event is sent to record and track the segmentation results, including the following information:

(i) The original image with the overlaid segmentation percentage (Figure 20, left);
(ii) The resulting image obtained by performing an AND operation between the original image after applying homography and the foam segmentation result (Figure 20, right);
(iii) A numeric value representing the percentage of the tank occupied by foam.

Additionally, an alert has been set up in the Event Server based on the foam percentage value. This alert triggers a notification to the designated person when the foam percentage exceeds a predefined threshold specified in the alert.



(**a**) (**b**) (**c**) (**d**) (**e**) (**f**)

**Figure 18.** Anomalous images by the position of the camera: (**a**) Image with parameter **pan = 282**, tilt = 9.81 and zoom = 2.2. (**b**) Image with parameter **pan = 280**, tilt = 9.81 and zoom = 2.2. (**c**) Image with parameter pan = 281, **tilt = 10.81** and zoom = 2.2. (**d**) Image with parameter pan = 281, **tilt = 8.81** and zoom = 2.2. (**e**) Image with parameter pan = 281, tilt = 9.81 and **zoom = 2.7**. (**f**) Image with parameter pan = 281, tilt = 9.81 and **zoom = 1.7**.



(**a**) (**b**)

**Figure 19.** Other anomalous image: (**a**) Image with occlusion. (**b**) Image of a wrong scenario.

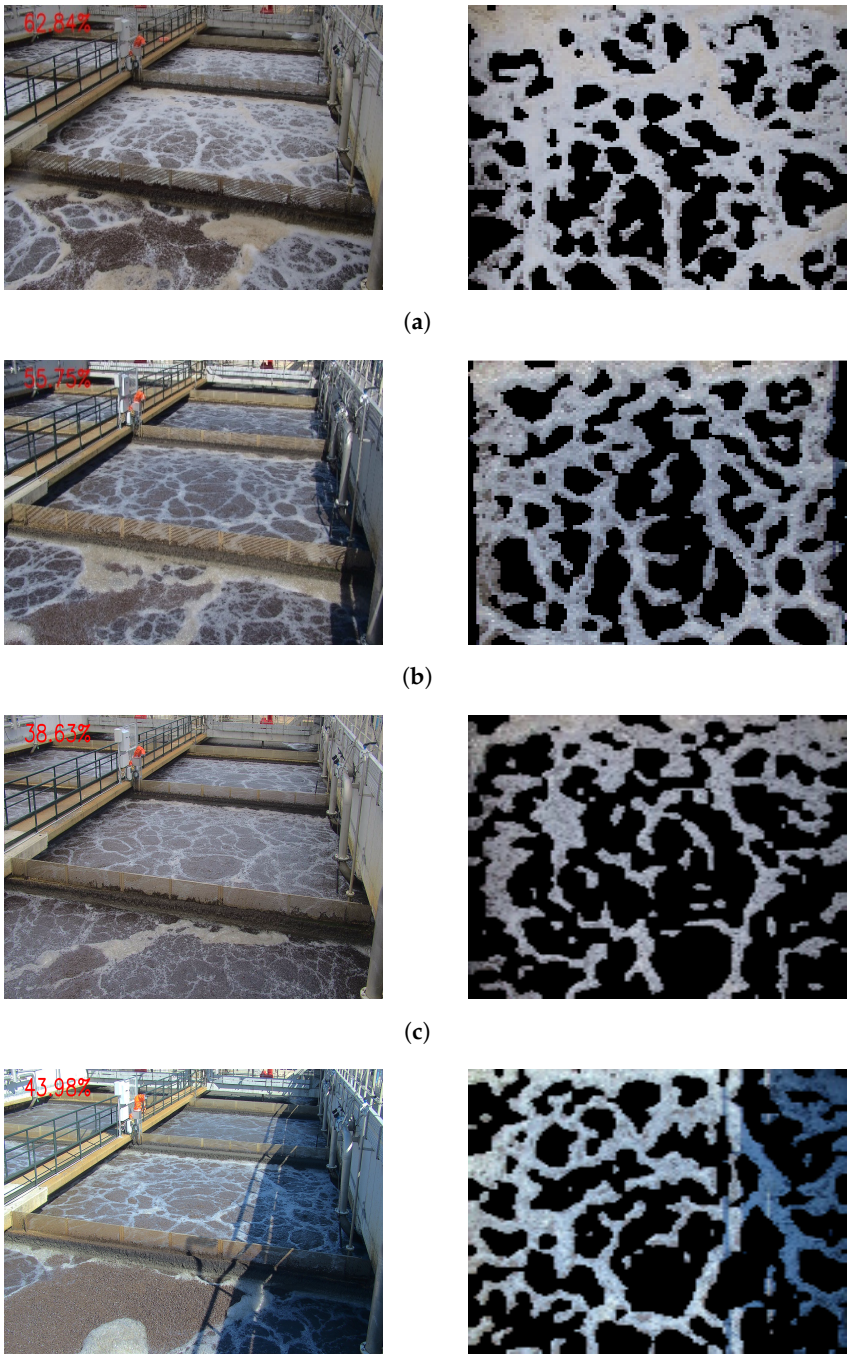



**Figure 20.** Original images with the percentage of foam superimposed and segmented images: (**a**) Image with a lot of foam. (**b**) Image with little foam. (**c**) Image raining. (**d**) Image with shadow.

## 4. Discussion

The calculation of foam percentage on the total surface area of a WWTP tank using computer vision algorithms can become a highly complex problem.

This work presented a complete remote sensing system based on imagen analysis to segment foams and monitor in real time the amount of foam in WWTP tanks. The developed segmentation models have been applied and validated in tanks of a biological reactor of a wastewater treatment plant with an MBBR (Moving Bed Biofilm Reactor) system. Due to the lack of a large set of tank images in the early stages of the research, the segmentation models have been trained on large public datasets used in similar segmentation applications and then retrained on the small foam dataset. In a later stage, the dataset collected at the plant

was large enough to train the models from scratch using only foam images. Additionally, the research carried out a comparison between two types of training: centralized and federated. The proposed methodology involves a complete computer vision pipeline that includes automatic image capture, ensuring the absence of anomalies in the captured images, image processing to segment foams and quantify the foam percentage of the total surface area of a tank, and the transmission of operationally significant data to the responsible personnel at the WWTP to support the foam removal at the right time.

Regarding limitations, managing lighting variations emerged as a primary challenge in achieving superior metrics. Additionally, accurately defining what is foam on the tank surface presented another notable limitation.

In this research, two texture segmentation models applied to foam segmentation are presented, yielding results surpassing 85% in the Dice metric and 75% in IoU. Furthermore, the percentage of foam on each tank calculated with the models has been validated by experts from the wastewater treatment plant using different images, thus verifying the proper functioning of the model. The experts' confidence implies that the system developed in this research is ready to be used in new WWTPs.

The comparison between centralized and federated training yielded similar results as both clients in the federated training have used images from the same WWTP. Nevertheless, the conducted research has allowed for the validation of the federated architecture. Consequently, it can be extrapolated for future use in different wastewater treatment stations.

Furthermore, the methodology explained in this paper has already been integrated into a real WWTP for their use, obtaining the following benefits:

- Reduction of maintenance requirements;
- Prevention of oxygen transfer issues;
- Avoidance of biomass washout;
- Enhanced monitoring and control;
- Avoid odour problems;
- Energy savings;
- Mitigation of risks accidents.

Moreover, due to the fact that foam formation can also commonly occur in other types of WWTPs reactors and processes, it is important to note that the technology presented can be implemented in plants using other biological treatments such as in Activated Sludge Processes, Anaerobic Reactors, or Sedimentation Tanks. However, such applicability needs to be tested and verified by applying the models in additional WWTPs.

Finally, it is important to state the relevance of this work in achieving the SDG6, as the techniques presented help improving the wastewater treatment, which is one of the six outcome targets of the goal.

For future work, the use of the federated framework in different wastewater treatment plants and the comparison of results with centralized training remains to be explored. A successful outcome in the utilization of various WWTP in the federated training could lead to a global foam segmentation model for WWTPs. Moreover, the utilization of distinct models depending on the month of the year may serve to mitigate the impact of atmospheric conditions on image segmentation.

**Author Contributions:** Conceptualization, J.C.M., S.G.V., J.F.Á., Á.D.R. and J.S.C.; methodology, J.C.M., S.G.V., Á.D.R. and J.S.C.; software, J.C.M., S.G.V., X.L. and J.S.C.; validation, J.C.M., S.G.V. and Á.D.R.; formal analysis, J.C.M. and S.G.V.; investigation, J.C.M., S.G.V., J.F.Á., X.L. and J.S.C.; resources, J.C.M., S.G.V., J.F.Á., Á.D.R. and L.G.G.; data curation, J.C.M. and J.S.C.; writing—original draft preparation, J.C.M.; writing—review and editing, J.C.M., S.G.V. and J.F.Á.; visualization, J.C.M.; supervision, S.G.V.; project administration, S.G.V., J.F.Á., Á.D.R., L.G.G. and J.S.S.; funding acquisition, S.G.V., L.G.G. and J.S.S. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research has been supported by the project EDAR360 (IN852B 2021/14), part of the Conecta Hubs 2021 programme, subsidized by the Galician Innovation Agency (GAIN) and co-financed by ERDF funds within the framework of the Feder Galicia 2014–2020 operational programme and also by the projects CEL.IA (CER-20211022) and CONFIA, financed by the CERVERA Research Program of CDTI, the Industrial and Technological Development Centre of Spain, in 2021 and 2023 respectively.

**Data Availability Statement:** The data from public datasets presented in this study are openly available in [10,11]. The data from the WWTP presented in this study are available on request from the corresponding author. The data are not publicly available due to privacy restrictions.

**Conflicts of Interest:** The authors declare no conflicts of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript; or in the decision to publish the results.

## Abbreviations

The following abbreviations are used in this manuscript:

| | |
|---|---|
| OSTS | One Shot Texture Segmentation |
| WWTPs | Waste Water Treatment Plants |
| MBBR | Moving Bed Biofilm Reactor |
| SDG6 | Sustainable Development Goal 6 |

## References

1. Madan, S.; Madan, R.; Hussain, A. Advancement in biological wastewater treatment using hybrid moving bed biofilm reactor (MBBR): A review. *Appl. Water Sci.* **2022**, *12*, 141. [CrossRef]
2. Collivignarelli, M.C.; Baldi, M.; Abba, A.; Caccamo, F.M.; Milno, M.C.; Rada, E.C.; Torretta, V. Foams in Wastewater Treatment Plants: From Causes to Control Methods. *Appl. Sci.* **2020**, *10*, 2716. [CrossRef]
3. Wang, W.; Huang, X.; Esmaili, A. Texture-Based Foam Segmentation and Analysis. *Ind. Eng. Chem. Res.* **2011**, *50*, 6071–6081. [CrossRef]
4. Forbes, G.; de Jager, G. Texture measures for improved watershed segmentation of froth images. In Proceedings of the Fifteenth Annual Symposium of the Pattern Recognition Association of South Africa, Grabouw, South Africa, 25–26 November 2004.
5. Kornilov, A.; Safonov, I.; Yakimchuk, I. A Review of Watershed Implementations for Segmentation of Volumetric Images. *J. Imaging* **2022**, *8*, 127. [CrossRef]
6. Zhu, L. A Research on Foam-Detection Based on Image Analysis in the Process of Sewage Treatment. In Proceedings of the 11th International Symposium on Distributed Computing and Applications to Business, Engineering & Science, Guilin, China, 19–22 October 2012; pp. 429–431. [CrossRef]
7. Tan, L.; Lu, J.; Jiang, H. Tomato Leaf Diseases Classification Based on Leaf Images: A Comparison between Classical Machine Learning and Deep Learning Methods. *AgriEngineering* **2021**, *3*, 542–558. [CrossRef]
8. Saffari, N.; Rashwan, H.A.; Abdel-Nasser, M.; Singh, V.K.; Arenas, M.; Mangina, E.; Herrera, B.; Puig, D. Fully Automated Breast Density Segmentation and Classification Using Deep Learning. *Diagnostics* **2020**, *10*, 988. [CrossRef]
9. Khan, S.; Muhammad, K.; Mumtaz, S.; Baik, S.W.; de Albuquerque, V.H.C. Energy-Efficient Deep CNN for Smoke Detection in Foggy IoT Environment. *IEEE Internet Things J.* **2019**, *6*, 9237–9245. [CrossRef]
10. Khan, S.; Muhammad, K.; Hussain, T.; Ser, J.D.; Cuzzolin, F.; Bhattacharyya, S.; Akhtar, Z.; de Albuquerque, V.H.C. DeepSmoke: Deep Learning Model for Smoke Detection and Segmentation in Outdoor Environments. *Expert Syst. Appl.* **2021**, *182*, 115125. [CrossRef]
11. Ustyuzhaninov, I.; Michaelis, C.; Brendel, W.; Bethge, M. One-shot Texture Segmentation. *arXiv* **2018**, arXiv:1807.02654. [CrossRef]
12. Kone, J.; McMahan, H.B.; Yu, F.X.; Rich, P.; Theertha, A.; Bacon, D. Federated Learning: Strategies for Improving Communication Efficiency. *arXiv* **2016**, arXiv:1610.05492. [CrossRef]
13. Howard, A.; Zhu, M.; Chen, B.; Kalenichenko, D.; Wang, W.; Weyand, T.; Andreetto, M.; Adam, H. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *arXiv* **2017**, arXiv:1704.04861. [CrossRef]
14. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778. [CrossRef]

15. Wang, J.; ke, S.; Cheng, T.; Jiang, B.; Deng, C.; Zhao, Y.; Liu, D.; Mu, Y.; Tan, M.; Wang, X.; et al. Deep High-Resolution Representation Learning for Visual Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *43*, 3349–3364. [CrossRef]
16. Zhang, R.; Isola, P.; Efros, A.; Shechtman, E.; Wang, O. The Unreasonable Effectiveness of Deep Features as a Perceptual Metric. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 586–595. [CrossRef]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.