

Article

Convolutional Neural Network for Measurement of Suspended Solids and Turbidity

Daniela Lopez-Betancur ^{1,2,†}, Ivan Moreno ^{2,*,†}, Carlos Guerrero-Mendez ^{2,*,†} , Tonatiuh Saucedo-Anaya ², Efrén González ³, Carlos Bautista-Capetillo ²  and Julián González-Trinidad ² 

¹ Dirección de Posgrados e Investigación, Universidad Politécnica de Aguascalientes, Calle Paseo San Gerardo #201, Fracc. San Gerardo, Aguascalientes 20342, Mexico; danielalopez106@gmail.com

² Unidad Académica de Ciencia y Tecnología de la Luz y la Materia, Universidad Autónoma de Zacatecas, Campus Siglo XXI, Zacatecas 98160, Mexico; tsaucedo@uaz.edu.mx (T.S.-A.); baucap@uaz.edu.mx (C.B.-C.); jgonza@uaz.edu.mx (J.G.-T.)

³ Unidad Académica de Ingeniería Eléctrica, Universidad Autónoma de Zacatecas, Campus Siglo XXI, Zacatecas 98160, Mexico; gonzalez_efren@hotmail.com

* Correspondence: imorenoh@uaz.edu.mx (I.M.); guerrero_mendez@uaz.edu.mx (C.G.-M.)

† These authors contributed equally to this work.

Featured Application: This research covers the development of a soft sensor model for dynamic processes based on convolutional neural networks for the measurement of suspended solids and turbidity.

Abstract: The great potential of the convolutional neural networks (CNNs) provides novel and alternative ways to monitor important parameters with high accuracy. In this study, we developed a soft sensor model for dynamic processes based on a CNN for the measurement of suspended solids and turbidity from a single image of the liquid sample to be measured by using a commercial smartphone camera (Android or IOS system) and light-emitting diode (LED) illumination. For this, an image dataset of liquid samples illuminated with white, red, green, and blue LED light was taken and used to train a CNN and fit a multiple linear regression (MLR) by using different color lighting, we evaluated which color gives more accurate information about the concentration of suspended particles in the sample. We implemented a pre-trained AlexNet model, and an MLR to estimate total suspended solids (TSS), and turbidity values in liquid samples based on suspended particles. The proposed technique obtained high goodness of fit ($R^2 = 0.99$). The best performance was achieved using white light, with an accuracy of 98.24% and 97.20% for TSS and turbidity, respectively, with an operational range of 0–800 mgL^{−1}, and 0–306 NTU. This system was designed for aquaculture environments and tested with both commercial fish feed and paprika. This motivates further research with different aquatic environments such as river water, domestic and industrial wastewater, and potable water, among others.

Keywords: computer vision for measurements; convolutional neural network (CNN); transfer learning; total suspended solids (TSS); turbidity



Citation: Lopez-Betancur, D.; Moreno, I.; Guerrero-Mendez, C.; Saucedo-Anaya, T.; González, E.; Bautista-Capetillo, C.; González-Trinidad, J. Convolutional Neural Network for Measurement of Suspended Solids and Turbidity. *Appl. Sci.* **2022**, *12*, 6079. <https://doi.org/10.3390/app12126079>

Academic Editors: Róbert Hudec, Patrik Kamencay and Milan Dado

Received: 23 April 2022

Accepted: 13 June 2022

Published: 15 June 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 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

In aquatic environments such as intensive aquaculture systems, there is an accumulation of organic and inorganic matter, feed residues, and aquatic microorganisms [1]. This accumulation is associated with total suspended solids (mgL^{−1}) and could be defined as the mass present in a water column. Suspended particles scatter and absorb light, causing turbidity or loss of transparency of the water [2]. Total suspended solids (TSS) is an important parameter in determining water quality. In aquaculture, a high level of TSS reduces the vision and ability of fish to catch their feed. Mathematically, a TSS value can be correlated with turbidity [3], which uses the degree of loss of water transparency due to suspended

solids. Therefore, the greater the number of suspended solids in the liquid, the greater the degree of turbidity [4].

For turbidity measurement, an established protocol is method 180.1 by the U.S EPA. The measurement ranges between 0 and 40 NTU (nephelometric turbidity units), and to achieve higher values, the samples must be diluted in water and the measurement must be rescaled. Otherwise, there are several arrangements for turbidimeters that use different light sources and detectors. However, none of them can be used as a low-cost alternative to monitor water quality in a rapid noninvasive way with a wide dynamic range. For example, some low-cost turbidimeters operate with short ranges; their operating ranges are 2.2–54.2 [5], 0–10 [6], 0–12 [7], and 0–100 NTU [8–10]. Furthermore, these methods do not allow the estimation of water turbidity using a single data record. There are advanced turbidimeters with a wide operating range, but they are expensive and require two data records for turbidity estimation [6,11].

By contrast, recent advances in computer vision, software development, and other technological advances are being implemented to solve some measurement limitations (the lack of a wide operating range, high cost, etc.) of conventional turbidimeters. Recent advances in turbidimeters are focused on image analysis. Examples include the novel turbidimeter technique of Gimenez et al. that developed a turbidimeter based on image degradation analysis; however, this method needs reference samples, and samples need additional treatment (must be sonicated) [12]. Gu et al. applied a random forest ensemble to space remote sensing data to obtain river turbidity measurement based on hyperspectral images, and obtained precision of 67% [13]. Mullins et al. carried out turbidity measurements in the range 10–250 NTU using image processing methods, where they processed and analyzed image-by-image in the measurement process, reaching a precision of 90% with controlled environmental conditions [14]. In addition, with the advent of smartphones, new turbidimeters have been reported. In the case of Bayram et al., they performed turbidity measurements using a smartphone; however, the precision between the samples measured with the calibrated Hach colorimeter and their smartphone colorimeter was 48% [15]. Leeuw and Boss also measured turbidity using a smartphone by remotely detecting water reflectance based on environmental conditions, reaching precision of 74% [16].

In this study, we measure turbidity and suspended solids by using a CNN that could be used as a low-cost alternative to monitor water quality in a rapid noninvasive way with a wide dynamic range. Our research proposes to use revolutionary emerging technology called convolutional neural network (CNN) and open-source technologies to develop a technique capable of measuring TSS and turbidity values simultaneously from a single image of the liquid sample to be measured. CNN is inspired by the working of the human brain and is able to analyze raw data without human intervention. CNNs have been applied successfully in different areas such as images classification [17,18], image segmentation [19], roughness measurement [20,21], and soft sensors [22–24]. Additionally, CNNs have gained popularity due to their ability to approximate any continuous function [25–27], so they could have better precision in the turbidity task than other machine learning methods such as additional trees, multilayer perceptron, naive Bayes, random forest, and support vector machine [13,28].

This research describes how to use a soft sensor [22] model for dynamic processes based on a convolutional neural network, with the highlights of measuring the TSS and turbidity values from a single image. This image is registered by a conventional smartphone (Android and IOS system). The measurements, acquired noninvasively, have high precision and a wide dynamic range for aquatic environments.

The rest of this paper is organized as follows. Section 2 describes the CNN architecture and the experiments; Section 3 shows the results and holds the discussion. Finally, Section 4 reports some conclusions and discusses future work.

2. Materials and Methods

2.1. Proposed Classification CNN

This paper is based on the principles of artificial intelligence, more specifically in operating blocks and several layer of neurons that work together to mimic the functioning of the visual cortex of mammals, which are called convolutional neuronal networks (CNNs). In image classification tasks, a CNN first performs a feature extraction step from the input image, then the features are passed to a neural network and finally we obtain, as an output, the probability assigned to the input image that it belongs to a certain category [29].

To work with CNNs, few people train a CNN from scratch, due to it is rare to have a large dataset, and it is more efficient to use the advantages of transfer learning, where a CNN model that was previously trained to perform a task is reused and trained again to learn a new task without the need to use large databases (e.g., 1000 classes for ImageNet dataset). Figure 1 shows the transfer learning process carried out on our CNN model, whose classification task is changed.

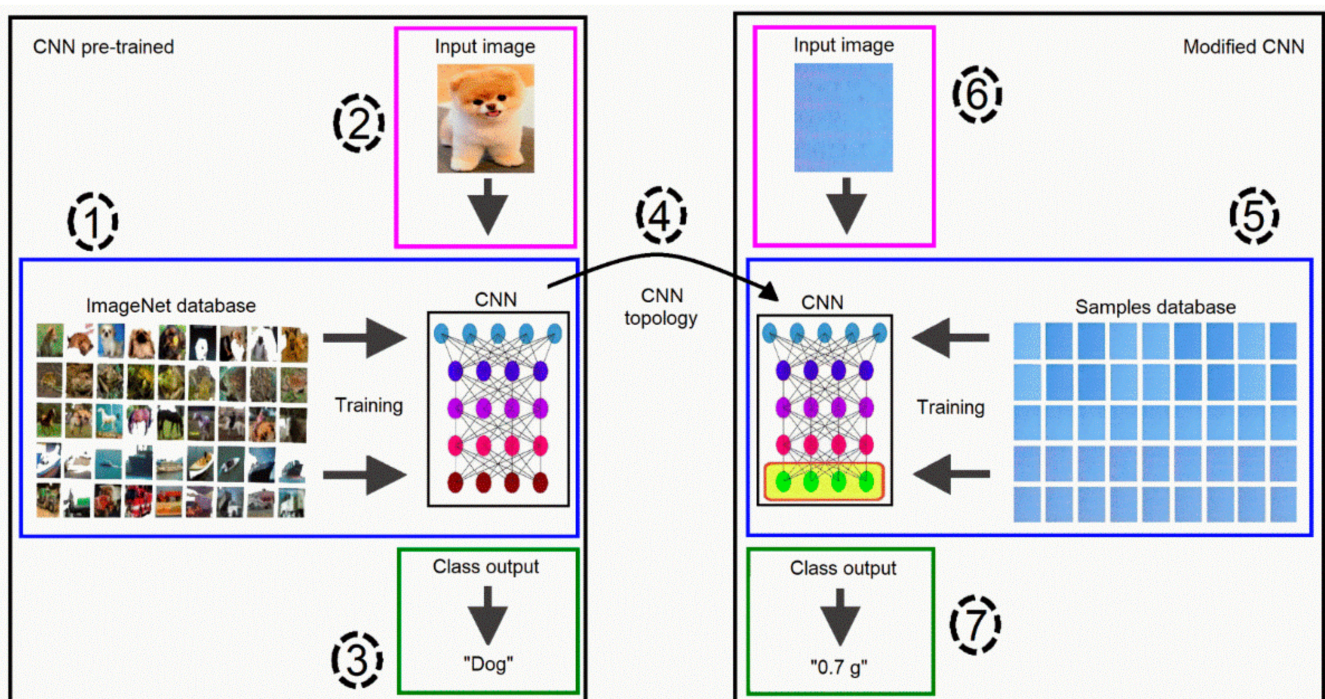


Figure 1. Representation of the TL process in the CNN model used: (1) training dataset (ImageNet) and CNN; (2) input image to classify; (3) classification result; (4) reuse of the pre-trained model for a new task; (5) new training dataset (images of the new task) and modification of the CNN; (6) new input image to be classified; and (7) classification result.

A simple and elementary CNN model with a powerful modeling capability is AlexNet [30]. AlexNet architecture is simple and easy to train and optimize, and has a proven ability to classify and recognize simple images with low visual complexity, such as low resolution [31]. For this, only AlexNet was trained, due to its simplicity and popularity. In this research, an AlexNet model is trained using an image dataset of liquid samples with different values of suspended solids. RGB-image input size for AlexNet is 224×224 [32]. The AlexNet model used is shown in Figure 2.

2.2. Proposed Estimator Based on Multiple Linear Regression (MLR)

Thus, we have a model capable of classifying the images in the dataset. The output vector of the trained model can be treated as a decoded version of the input image since the model extracts implicit information from the liquid sample. Although the CNN model classifies certain liquid samples, it cannot classify images with intermediate values of

suspended solids. Nevertheless, if the feature vectors (CCN output vectors) are used to fit an MLR, we can predict the TSS and turbidity values of any liquid sample. The main requirement is to train the CNN with a number of classes that contain the target dynamic range.

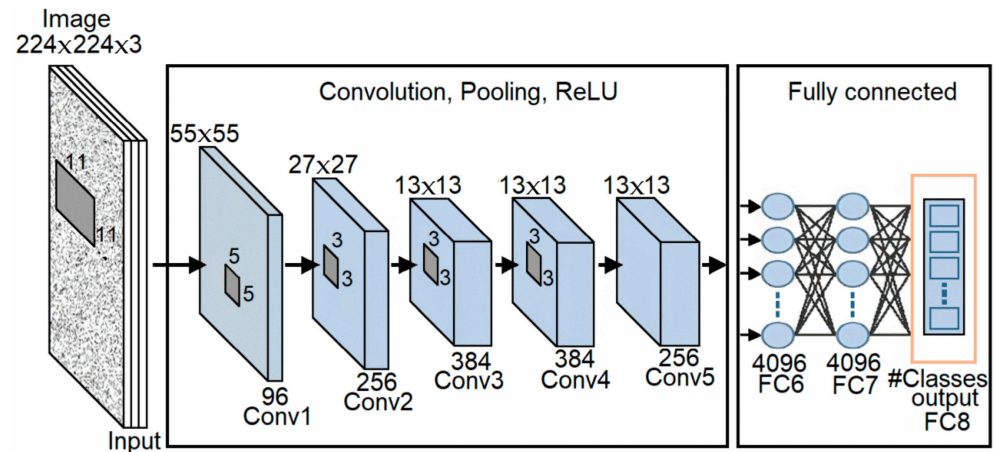


Figure 2. Details of the trained CNN model used in this research. The layer FC8 is marked to indicate the resizing of this layer according to the number of classes in the new task.

In a convolutional neural network, there are two steps: the first is called feature extraction, and the second is classification; the second step is built from a branch of neurons. For example, a single neuron uses the inputs (features) to compute a response (output) or a single logit, which is the value of multiplying the inputs with weights and adding the bias term ($-\infty$ to $+\infty$), and then passed through the activation function to obtain an output. Therefore, a logit vector can be acquired by using the real number calculated in a group of neurons. In CNN models, in the last layer, to obtain the probabilities of the classes, it is required to input the logit values into a SoftMax function to generate a normalized vector of probabilities (0 to 1) with a value for each predicted class.

The MLR is then fitted with the logits vector obtained from the last layer, and a linear equation is created to approximate new and unknown values. Figure 3 shows how the MLR is fitted using the logit vector of the training dataset, and the general operation of the proposed method. When the CNN input is a new image (unknown sample), a new logit vector is created, and used in the MLR to obtain the estimated value. Hence, in this research, class probabilities are used to predict the TSS and turbidity values in a liquid sample.

Furthermore, the CNN and the MLP were trained separately. The CNN was trained using the Cross-entropy function, so a simple classification task was developed. A trained CNN was used to obtain the probabilities based on the input image, then a trained MLP was used to obtain the approximated value according to the values (obtained in the training process) used to fit the MLP. In other words, an MLP was used to predict the value of a variable based on the value of another variable. We then obtained the TSS and turbidity values based on the features (probabilities) found by the CNN.

2.3. CNN Validation as Classifier

One of the main advantages of using a CNN is that it automatically learns the most relevant features in an input image without human supervision. By viewing the convolutional feature maps of an image, we can look at the regions of the image noted by the CNN to perform the classification. In this research, several sets of feature maps were analyzed to confirm that CNN can detect image changes due to the particles suspended in the liquid samples. This would confirm that CNN is only counting differences in suspended solids and ignoring other parameters such as optic aberrations, spurious radiation, mismatch compensated pulse effects, etc. Figure 4 shows a set of convolutional feature maps extracted from the stacked results in each convolutional layer.

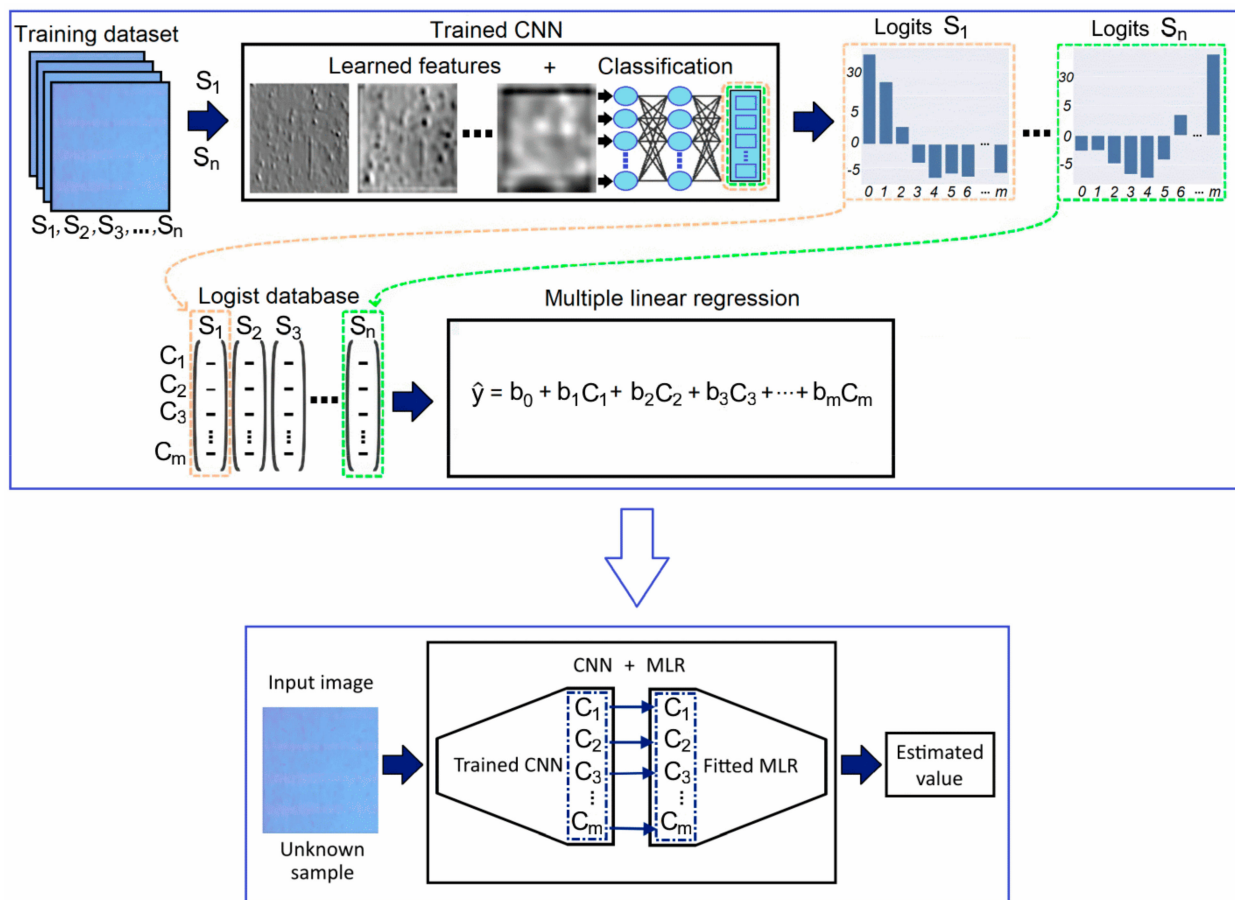


Figure 3. Extraction of logit vectors used to fit the MLR and the complete measurement system.

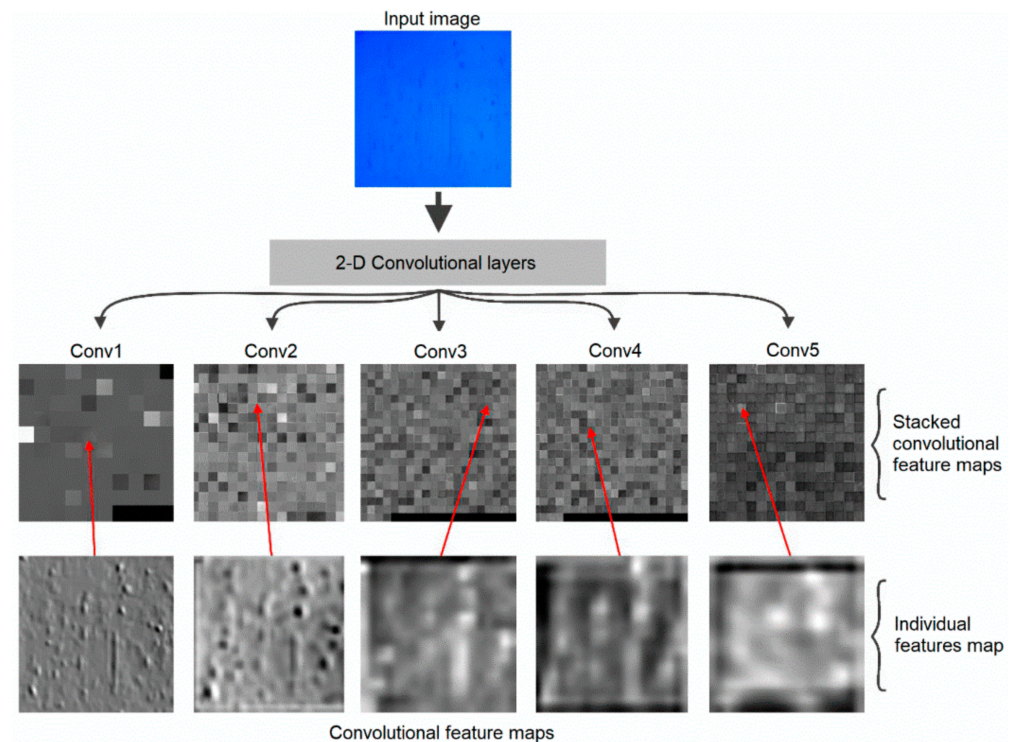


Figure 4. Convolutional feature maps of a liquid sample.

2.4. Samples

Two samples of commercial interest were used: fish feed and paprika. The study was mainly based on fish feed, and to validate our results, paprika (with 2 extra classes) was tested. Commercial fish feed was used as the suspended solid, given that fish feed is one of the components that accumulates the most in intensive aquaculture systems [33,34]. Additionally, because fish feed for tilapia is designed to be in suspension [35], it makes it easy to dose. Twenty liquid samples were prepared by mixing one liter of distilled water with each mass (g) of the fish feed and paprika mass, as shown in Table 1. The masses were created on a Denver Instruments PI-214 high-resolution balance (with four decimal places). The sample concentration ranged from 0 to 0.8 gL^{-1} , chosen because this is the operating range of most commercially available turbidimeters and/or solids meters, such as HANNA and Hach instruments. Concentration samples (fish feed and paprika mass) used for each class are shown in Table 1.

Table 1. Concentration samples (fish feed and paprika mass) used for each class.

Classes	0	1/2	3/4	1	2	3	4	5	6	7	8
Fish mass (g)	0.000	-	-	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800
Paprika mass (g)	0.000	0.002	0.007	0.100	0.200	0.300	0.400	0.500	0.600	0.700	0.800

2.5. Experimental Setup

A transparent container with a cubic shape of approximately $5 \times 5 \times 5 \text{ cm}$ was filled using the liquid sample. A magnetic mixer was used to keep all the particles suspended, for preventing them from settling, and to allow the suspended particles to be recorded by the camera. The mixer speed used was 60 rpm during image registration (images from the training and validation dataset). The experimental setup used for this experiment is shown in Figure 5. The illumination distance was selected to prevent shadows on the liquid sample. The camera distance was selected to avoid imaging the edges of the container, so that the model does not learn the morphology of the container during the CNN training. The experiment was carried out in a dark room. The background was constant throughout the experiment; therefore, the CNN learned the information from the samples for each color, and it did not take information about the background. This is also corroborated in Figure 4, where the activations of the artificial neurons were focused, or the CNN took the information, for the classification of the suspended solid samples. The experiment setup implemented an RGB LED lamp. Red, green, blue, and white were used. Liquid samples illuminated with different colors are shown in Figure 6. The illumination of all colors was kept constant at an irradiance of 0.852 mW/cm^2 . The spectral power distributions of the illumination used in this study are shown in Figure 7, which were measured using a spectrophotometer (USB2000, Ocean Optics, Orlando, FL, USA). The experimental setup was adjusted to the neural network by keeping constant all parameters (smartphone camera mode, LED lamp intensity, etc.) that were not under measurement. The neural network learned to identify the image changing characteristics, and since its training included samples with only distilled water (class 0), and considering that the only parameter which varied was the suspended solids, the effects of spurious light and other sources of noise were minimized.

To create and record the liquid sample dataset, the rear-facing cameras of a Huawei Mate 20 Lite and an iPhone 6 were used. A total of 88,000 images were recorded. For samples with fish feed, 39,600 images were recorded with an Android operating system smartphone, and 48,400 images for paprika samples utilizing the smartphone with an IOS operating system. The size of the images recorded via the smartphones was 2448×2448 pixels; however, the images were center-cropped at 224×224 pixels, according to the input layer of the CNN. The smartphone camera was used in manual mode with a capture speed of 20 FPS (frames per second) in burst mode. The ISO level and the focus (focused at the container wall) option were kept constant during the experiment. Of

the 88,000 recorded images, 80,000 images were randomly selected to create the training images dataset. The remaining 8000 images were used to build the validation dataset, i.e., 90% of the dataset was used for training, while the remaining 10% was used for validation or to test the operation of the CNN, that is, the validation images were not used in the validation process or training test.

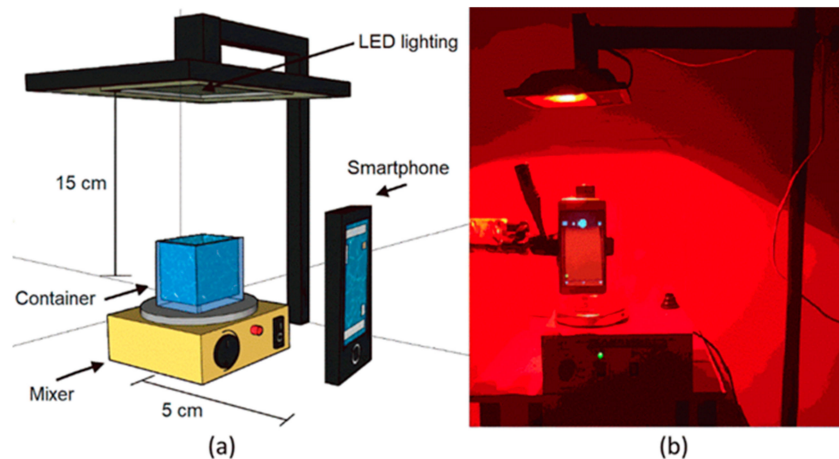


Figure 5. Experimental setup: (a) scheme of the experimental setup; (b) experimental setup used.

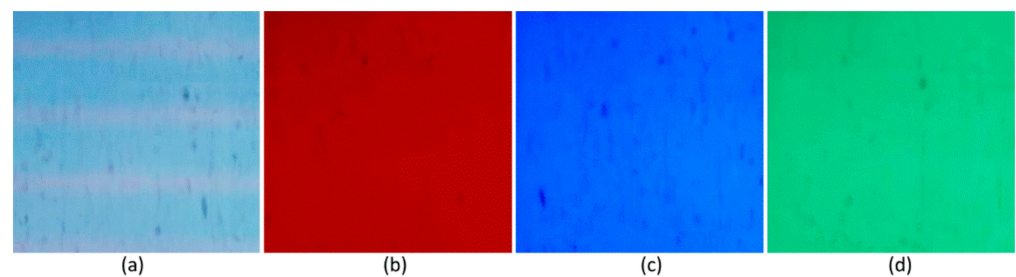


Figure 6. Recorded images of samples illuminated with a concentration of 0.1 gL^{-1} of the fish feed mass in distilled water and under different lighting colors: (a) white light; (b) red; (c) blue; (d) green.

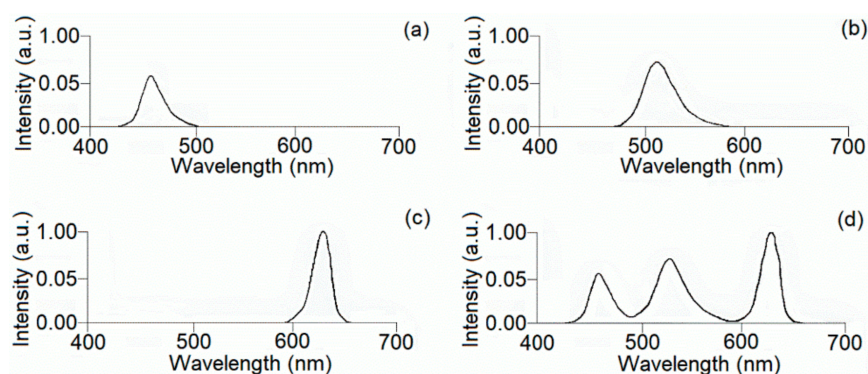


Figure 7. The spectral power distributions of the RGB LED lamp used in this study: (a) blue; (b) green; (c) red; and (d) white.

The training process was developed and implemented using Google Colaboratory, which is a free cloud service for machine learning education and provides a Python notebook (Jupyter) environment running in a dedicated virtual machine on an Nvidia Tesla K80 GPU with 2496 CUDA cores. The AlexNet model (CNN) was taken from the Torchvision package, which offers some popular pre-trained models and other image processing tools.

In artificial intelligence, the process of finding the “best” or “optimal” parameters for the performance of a CNN model is called optimization. The classic optimization

method is called stochastic gradient descent (SGD). It is a simple procedure that involves iteratively finding the values that result in the lowest possible error (loss) based on the training dataset. Although newer and more powerful optimization algorithms exist, SGD provides consistency in the overall training process and results.

One of the most important hyperparameters is the “learning rate”. It is responsible for adjusting the rate at which the model calculates the gradient of the loss function. Therefore, the learning rate controls how much the model changes its predictions as it updates its results based on model error. A high learning rate makes the model change its parameters quickly, while a low learning rate makes the model change its parameters slowly. The best option is the selection of a learning rate value that makes the error decrease correctly (not too quickly), finding the minimal error in the fewest number of epochs. An “epoch” is another relevant hyperparameter; it refers to the number of times that the entire training dataset is passed through the CNN model. However, a model is trained using batches. In the context of a single training epoch, “batch size” refers to the amount of data passed to be processed by the CNN and updates model parameters at a time until an epoch is complete. Larger batches allow for more computational parallelism, and can often lead to better performance.

However, larger batches also require more memory and can cause latency when passed into the training function. Finally, the hyperparameter “momentum” is employed to accelerate the gradient descent by taking into account a fraction of the previous gradients to update to the current one.

For the CNN training, the algorithm executed a total of 75 training epochs. The epoch number was selected by analyzing the loss of training according to previous executions of the training process. The CNN was trained using a stochastic gradient descent algorithm with a momentum of 0.9 and a batch size of 50. The other hyperparameters used in the experiment are listed in Table 2.

Table 2. Hyperparameters used in the training process.

Hyperparameters	Value
Algorithm optimizer	SGD
Learning rate	0.0005
Momentum	0.9
Batch size	50
Number of epochs	75

The selection of the hyperparameters was established based on the image classification examples exposed in the PyTorch documentation. In addition, several executions were carried out in search of minimizing errors, and until a high accuracy of classification (100%) and estimation (98.24% and 97.20% for TSS and turbidity, respectively) was reached.

2.6. Performance Metrics

The performance of the proposed method was evaluated in two stages.

2.6.1. Performance Metrics for Classifier Evaluation

A confusion matrix is used to analyze the performance of a classification tool [36–38]. Four important terms make up a confusion matrix, which describes the following cases: the cases that were predicted as elements that belong to a class and that actually belong to that class are known as true positives (*TP*). Similarly, the cases of elements that were predicted that do not belong to a class and do not belong to that class are known as true negatives (*TN*). On the other hand, the cases of elements that were predicted to belong to a class but which do not really belong to the class are named false positives (*FP*). Finally, the cases that were predicted not to belong to the class and actually belong to the class are known as false negatives (*FN*). The elements of the confusion matrix are used to calculate the following performance metrics for the evaluation of the classifier:

Accuracy is the percentage of the total number of predictions that were classified correctly and is calculated as:

$$Accuracy = (TP + FN) / N \quad (1)$$

where N is the total number of elements to classify.

Precision is the ability of the classifier to predict a sample according to what it really is, and is defined as:

$$Precision = TP / (TP + FP) \quad (2)$$

Recall is the ability of the classifier to find all the positive samples. In other words, how many examples of positive cases were correctly labeled, and can be written as:

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

Similar to *Recall*, *Specificity* is the ability of the classifier to find all the negative samples, and is defined as:

$$Specificity = TN / (TN + FP) \quad (4)$$

F-Score is the harmonic mean of *Precision* and *Recall*, and provides a notion of how precise the classifier is. A high *F-Score* value indicates that the model performs better in positive cases. It is calculated as:

$$F-Score = (2 \times Precision \times Recall) / (Precision + Recall) \quad (5)$$

Receiver operating characteristic (ROC) is a plot of the rate of true positives (*Recall*) versus the rate of false positives (*Specificity*). This graph characterizes the ability of a CNN to identify positive cases as positive, and negative cases as negative. Meanwhile, the area under the ROC curve (AUC) is the probability that a randomly chosen pair of positive and negative cases will be classified correctly.

2.6.2. Performance Metrics for MLR Evaluation

The following metrics were used to evaluate the performance of the MLR, whose task is to estimate the correct measured value.

The coefficient of determination (R^2) is a statistical measure of the goodness of fit or reliability of the model according to the data. This coefficient determines the quality of the model to replicate the results. The values of R^2 are between 0 and 1. Zero implies that there is no linear relationship, and a value of one means that there is a perfect linear relationship. The coefficient of determination is calculated as:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_{predicted_i} - y_{mean})^2}{\sum_{i=1}^n (y_{true_i} - y_{mean})^2} \quad (6)$$

where $y_{predicted}$ is defined as the predicted value, y_{true} as the true value, and y_{mean} as the average of the y data.

Mean absolute error (MAE) is evaluation metric used in regression models. It is the mean of the difference between the original values (y_{true}) and the predicted values ($y_{predicted}$). Mathematically, it is described as:

$$MAE = \frac{\sum_{i=1}^N abs(y_{true_i} - y_{predicted_i})}{N} \quad (7)$$

Mean square error (MSE) is defined as the difference between the original values and the predicted values, and squared by the mean difference; the higher this value, the worse the model. Mathematically, it is represented as:

$$MSE = \frac{\sum_{i=1}^N (y_{true_i} - y_{predicted_i})^2}{N} \quad (8)$$

3. Results and Discussion

This section shows the results of the evaluation of CNN's performance metrics. The CNN model was independently trained four times; in each training we used a color training dataset (white, red, green, and blue datasets). In addition, the efficiency of the MLR was evaluated to estimate the TSS and turbidity values. The validation dataset that was used consisted of nine classes for each color with 100 images in each class. For the CNN evaluation, a confusion matrix was implemented in which the rows are related to the true labels and the columns to the labels predicted by the CNN. Diagonal cells are linked to observations that are correctly classified. A perfect classification is reached when each space of the diagonal elements counts 100. The confusion matrix obtained at the end of the training process for the four color datasets was the same for fish feed and paprika (no extra classes) and is shown in Figure 8. If we visualize the extra classes in the confusion matrix, the diagonal also has 100 elements. The success of this classification may be due both to the size of the dataset (which is relatively large) and to the lack of complexity of the classification objects. In addition, since the MLR evaluation metrics had a percentage of error, we can guarantee that the CNN did not memorize the dataset and was not overfitted. The trained CNN for each color dataset reached a maximum score on its performance evaluation metrics for accuracy, precision, recall, F-score, and ROC (see Figure 9). The performance obtained for the trained CNN for each color dataset is shown in Table 3.

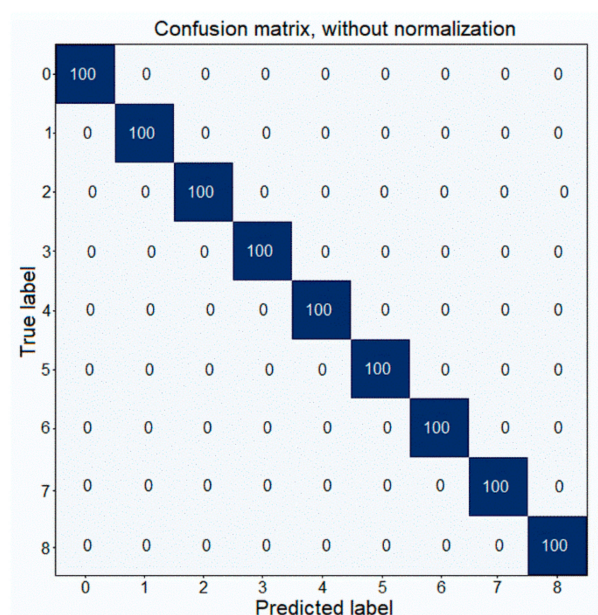


Figure 8. Confusion matrix reached for all colors.

Although the trained CNN achieved the same high score on the CNN performance metrics for each color dataset, it should be noted that the training time was different for each color dataset. The best training time was reached using the white light dataset. On the other hand, the worst training time was obtained using the green dataset. The difference between the CNNs trained with the white and green databases was 266% in training time. The CNN trained with the red dataset was the second-best model, which reached the maximum score in almost double the amount of time in comparison to the white one. These differences could be attributed to the fact that the CNN creates individual feature maps for each RGB color channel [39]. Therefore, the white color, which is the combination of the three color channels, could generate more detailed feature maps using the three color channels, which could allow the CNN to classify all classes more effectively. Training time is a parameter that gives us information about which color dataset is best categorized by the CNN. Once the CNN has been trained, when entering a new image, it calculates its TSS and turbidity values in fractions of a second. The accuracy and loss curves in the training process are

shown in Figure 10. It is easy to see that the blue classifier (CNN model trained with the blue dataset) was the first to achieve a high accuracy score; however, its training process started with a low accuracy value of 42% in its first epoch, and only reached 100% accuracy at the 15th epoch. Another important aspect concerns the green classifier, which started with the lowest accuracy score of 12%, and reached a high accuracy value at the 12th epoch, but continued its training process with some fluctuations. The red classifier started with a low accuracy value of 45% in its first epoch, and reached 100% accuracy at the 12th epoch. Meanwhile, the white classifier obtained an accuracy value of 90% in its first epoch, and obtained an accuracy of 100% in its seventh one.

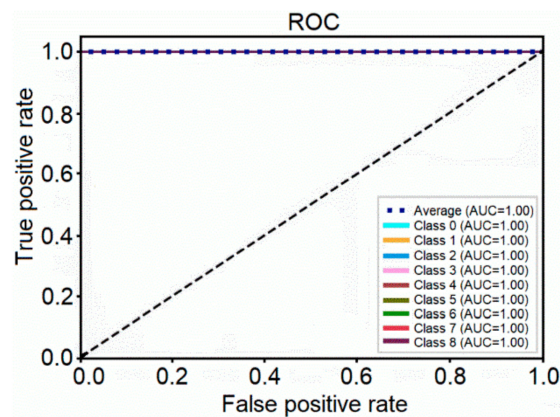


Figure 9. ROC curve obtained in the CNN performance evaluation (all colors). **Note:** all classes overlap in the horizontal line.

Table 3. Performance measures of the CNN as classifier for each color dataset.

Metrics	White	Red	Blue	Green
Accuracy	1.00	1.00	1.00	1.00
Precision	1.00	1.00	1.00	1.00
Recall	1.00	1.00	1.00	1.00
F-Score	1.00	1.00	1.00	1.00
Fish Feed training time (min)	65.38	102.88	156.53	174.11
Paprika training time (min)	162.49	200.58	232.62	236.08

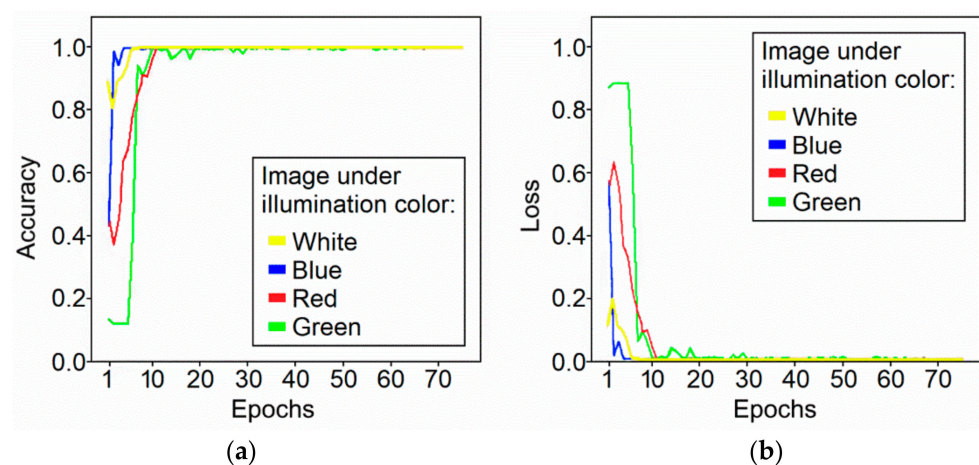


Figure 10. (a) Classifier training accuracy for each validation color dataset. (b) Loss for each validation color dataset.

Differences between RGB colors in terms of accuracy and loss curves can be related to the spectral characteristics of the light. In particular, the sample was a brown color for

fish feed, whose spectrum has more red-light content, and then the red image may have more information, which may improve the red channel analysis. Regarding the green and blue channels, the wavelength peak of green LED illumination is about 40 nm displaced from the peak wavelength of the green pixel responsivity of the CMOS camera, while the wavelength peak of the blue LED lamp is about 10 nm displaced. This spectral mismatch may reduce the information of the green channel in comparison with blue and red channels. This is similar for paprika samples whose spectrum has more red-light content.

MLR performance measures were calculated and the results obtained for the different color datasets are shown in Table 4. When looking at the results in Table 4, it is observed that the MLR for the white dataset had the highest value of the coefficient of determination R^2 , and furthermore, the highest model quality to replicate the results. In addition, the MLR with the white dataset had the lowest MAE and MSE values. Therefore, the best MLR performance for the estimation task was performed with the white dataset, unlike the green illuminated samples, which had the worst MLR performance. The study was mainly focused on fish feed and, to validate our results, paprika was tested. For both samples, the CNN + MLR have the performance listed in Table 4. The liquid samples created in this research have a TSS range of 0–800 mgL⁻¹. The TSS values, estimated by both the CNN + MLR, are shown in Table 5 for the different color datasets.

Table 4. MLR performance metrics for each color dataset for the fish feed and paprika samples.

Metrics	Sample	White	Red	Blue	Green	Sample	White	Red	Blue	Green
R^2		0.991	0.983	0.958	0.913		0.999	0.991	0.997	0.984
MAE		0.016	0.021	0.035	0.052		0.006	0.016	0.009	0.030
MSE		0.0004	0.0009	0.0022	0.0045		0.0002	0.0009	0.0004	0.0004

Table 5. Mean values $\pm$ standard deviation of the TSS estimated by the CNN + MLR for each color dataset for the fish feed and paprika samples.

Samples	Classes	Colors			
		White	Red	Blue	Green
	0	0.019 $\pm$ 0.024	0.021 $\pm$ 0.002	0.021 $\pm$ 0.035	0.083 $\pm$ 0.102
	1	0.109 $\pm$ 0.013	0.117 $\pm$ 0.026	0.118 $\pm$ 0.031	0.120 $\pm$ 0.056
	2	0.189 $\pm$ 0.014	0.203 $\pm$ 0.016	0.215 $\pm$ 0.041	0.215 $\pm$ 0.019
	3	0.296 $\pm$ 0.017	0.303 $\pm$ 0.016	0.310 $\pm$ 0.036	0.262 $\pm$ 0.020
	4	0.396 $\pm$ 0.020	0.397 $\pm$ 0.033	0.399 $\pm$ 0.061	0.319 $\pm$ 0.043
	5	0.494 $\pm$ 0.036	0.499 $\pm$ 0.052	0.496 $\pm$ 0.039	0.512 $\pm$ 0.041
	6	0.592 $\pm$ 0.015	0.592 $\pm$ 0.035	0.590 $\pm$ 0.063	0.549 $\pm$ 0.065
	7	0.705 $\pm$ 0.014	0.713 $\pm$ 0.023	0.685 $\pm$ 0.050	0.693 $\pm$ 0.054
	8	0.809 $\pm$ 0.013	0.778 $\pm$ 0.022	0.785 $\pm$ 0.069	0.848 $\pm$ 0.010
	0	0.000 $\pm$ 0.013	0.000 $\pm$ 0.019	0.000 $\pm$ 0.043	0.000 $\pm$ 0.119
	1/2	0.002 $\pm$ 0.012	0.005 $\pm$ 0.030	0.004 $\pm$ 0.019	0.005 $\pm$ 0.027
	3/4	0.007 $\pm$ 0.015	0.006 $\pm$ 0.019	0.004 $\pm$ 0.019	0.015 $\pm$ 0.020
	1	0.102 $\pm$ 0.011	0.103 $\pm$ 0.025	0.091 $\pm$ 0.040	0.113 $\pm$ 0.107
	2	0.199 $\pm$ 0.021	0.202 $\pm$ 0.019	0.195 $\pm$ 0.045	0.183 $\pm$ 0.024
	3	0.291 $\pm$ 0.019	0.303 $\pm$ 0.027	0.298 $\pm$ 0.060	0.267 $\pm$ 0.027
	4	0.390 $\pm$ 0.010	0.366 $\pm$ 0.045	0.426 $\pm$ 0.041	0.324 $\pm$ 0.049
	5	0.496 $\pm$ 0.011	0.459 $\pm$ 0.032	0.494 $\pm$ 0.038	0.468 $\pm$ 0.041
	6	0.603 $\pm$ 0.015	0.600 $\pm$ 0.055	0.612 $\pm$ 0.058	0.584 $\pm$ 0.060
	7	0.689 $\pm$ 0.020	0.682 $\pm$ 0.030	0.694 $\pm$ 0.045	0.722 $\pm$ 0.049
	8	0.789 $\pm$ 0.016	0.805 $\pm$ 0.020	0.799 $\pm$ 0.058	0.776 $\pm$ 0.035

Performance metrics shown in Tables 4 and 5 indicate that the best light illumination for the proposed method was white color for the fish feed sample. This is because the white color showed an error of 2.53% compared to 3.16% for red, 4.16% for blue, and 9.57% for green. The proposed method with the white dataset had an operational range

of 0 to 0.8 gL^{-1} and high goodness of fit ($R^2 = 0.99$). Therefore, the method with the white dataset obtained the lowest error of $\pm 2.53\%$ and a general standard deviation of ± 0.018 , which implies an accuracy of 97.46%. These errors are the measurement errors (observational errors), which were calculated by using the values of the reference samples listed in the Table 1 (taken as the true values). In general, the results indicate that the measuring precision is reasonably good for the fish feed sample. However, the two smaller concentrations (Classes 0 and 1) had lower precisions. To improve this problem and test repeatability, we repeated and expanded the measurement with a different sample, paprika, and a smartphone with a different operating system (iPhone 6, IOS system).

In order to improve and validate our results, we analyzed a set of liquid samples made with paprika. This additional set of samples incorporated two new classes (1/2 and 3/4) into the CNN training process, which significantly reduced the measuring error of classes 0 and 1. Additionally, Table 5 shows TSS values for the paprika samples with each color illumination dataset. In view of the training of the new classes, the CNN + MLR improved its accuracy to 98.2400% for the TSS values with the white dataset.

It should be highlighted that between TSS and turbidity there is a correlation due to the coefficient of proportionality that creates a linear regression between them. This coefficient of proportionality between TSS and turbidity depends on the geometric and optical properties of the suspended solids (i.e., size, shape, refraction index, mass density) [40–42]. In other words, the samples of fish feed and paprika have the same concentration of TSS, but have different concentrations of turbidity since they have a different coefficient of proportionality.

In order to estimate the turbidity values in the liquid samples, the system was trained and validated using reference values measured with another instrument. These turbidity reference values were measured with a HACH DR900 colorimeter, within the operating range 0–263 NTU, and are shown in Table 6. The reference measurements were replicated six times to obtain the standard deviation of the device. Note that when making measurements near 200 NTU with the HACH DR900, the standard deviations increased, due to it being adjusted with a calibration curve using the reading obtained with the 200 NTU formazin standard. Additionally, according to the user manual, the instrument error is ± 21 NTU [43]. Table 6 shows the TSS and turbidity reference values for the fish feed and paprika samples. The turbidity values estimated by our method are shown in Table 7 for the different color datasets. The white dataset showed the lowest standard deviation values among all color datasets. It had a maximum standard deviation of ± 13.68 NTU for the fifth class and, in turn, had a lower standard deviation than the instrument used as reference. In addition, for the fish feed samples, the white color presented an error of 9.84% compared to 11.64% for red, 12.55% for blue, and 42.50% for green. Therefore, the 0–263 NTU range is appropriate for aquatic environments, as the safe turbidity level for aquatic life should not exceed 25 NTU [44]. For the turbidity measurement, the standard deviation in our proposed method was ± 6.98 NTU and an accuracy of 90.16% for the white dataset, which was the best color dataset for the fish feed samples. Additionally, Table 7 shows the turbidity values estimated by our method for paprika samples, and it can be noted that by training the new classes, the CNN + MLR improved its accuracy to 97.20% for turbidity values using the white dataset. In order to test our method, eight new samples with paprika were prepared with fractional concentrations of TSS. To validate the proposed method, these samples were estimated without training the CNN or the MLR for these TSS values. The TSS values estimated by the CNN + MLR are shown in Table 8, with an accuracy of 96.88% for the white dataset. The turbidity values estimated by the proposed method are shown in Table 9, with an accuracy of 96.14% for the white dataset. The CNN + MLR were validated for extra concentration samples.

Table 6. Turbidity samples of the reference measured with the HACH DR900 for each TSS of the fish feed and paprika samples. Turbidity samples are shown as mean data $\pm$ standard deviation.

Samples	Classes	TSS (g/L ⁻¹)	Turbidity (NTU)	Samples	TSS (g/L ⁻¹)	Turbidity (NTU)
	0	0.000	0 $\pm$ 0.55		0.000	0 $\pm$ 0.48
	1/2	-	-		0.002	2 $\pm$ 2.45
	3/4	-	-		0.007	3 $\pm$ 3.69
	1	0.100	38 $\pm$ 2.61		0.100	27 $\pm$ 5.21
	2	0.200	77 $\pm$ 7.06		0.200	61 $\pm$ 2.87
	3	0.300	95 $\pm$ 13.62		0.300	90 $\pm$ 4.95
	4	0.400	132 $\pm$ 12.37		0.400	123 $\pm$ 4.75
	5	0.500	168 $\pm$ 12.67		0.500	145 $\pm$ 7.71
	6	0.600	189 $\pm$ 22.66		0.600	170 $\pm$ 4.92
	7	0.700	245 $\pm$ 21.65		0.700	248 $\pm$ 2.39
	8	0.800	263 $\pm$ 23.73		0.800	306 $\pm$ 3.87

Table 7. Turbidity measured with the proposed method. It shows mean values $\pm$ standard deviation of the turbidity estimated by the CNN + MLR for each color dataset for the fish feed and paprika samples.

Samples	Classes	Colors			
		White	Red	Blue	Green
	0	7.14 $\pm$ 9.00	8.09 $\pm$ 6.08	8.05 $\pm$ 13.41	31.43 $\pm$ 38.65
	1	41.39 $\pm$ 4.94	44.70 $\pm$ 10.03	44.98 $\pm$ 11.66	45.68 $\pm$ 21.28
	2	75.05 $\pm$ 5.36	77.54 $\pm$ 6.19	79.72 $\pm$ 15.58	79.74 $\pm$ 7.11
	3	94.26 $\pm$ 6.38	96.22 $\pm$ 5.96	98.81 $\pm$ 13.76	88.21 $\pm$ 7.49
	4	130.48 $\pm$ 7.71	130.89 $\pm$ 12.65	131.78 $\pm$ 23.06	102.07 $\pm$ 16.30
	5	165.91 $\pm$ 13.68	167.89 $\pm$ 19.61	166.52 $\pm$ 14.70	170.52 $\pm$ 15.58
	6	187.24 $\pm$ 5.77	187.32 $\pm$ 0.04	187.07 $\pm$ 23.83	178.35 $\pm$ 24.55
	7	245.97 $\pm$ 5.24	247.30 $\pm$ 8.81	236.60 $\pm$ 19.04	241.14 $\pm$ 20.56
	8	266.25 $\pm$ 4.79	259.11 $\pm$ 8.39	260.34 $\pm$ 26.07	278.65 $\pm$ 3.99
	0	0.09 $\pm$ 3.56	0.36 $\pm$ 5.10	0.18 $\pm$ 11.69	0.27 $\pm$ 33.43
	1/2	1.91 $\pm$ 2.00	2.85 $\pm$ 5.10	1.16 $\pm$ 11.69	2.65 $\pm$ 33.43
	3/4	2.95 $\pm$ 2.15	1.67 $\pm$ 5.40	1.16 $\pm$ 15.66	5.17 $\pm$ 9.56
	1	27.54 $\pm$ 2.29	27.92 $\pm$ 6.69	24.70 $\pm$ 10.88	31.49 $\pm$ 29.31
	2	60.55 $\pm$ 2.99	61.46 $\pm$ 5.02	59.33 $\pm$ 12.15	55.15 $\pm$ 6.48
	3	87.44 $\pm$ 5.07	91.06 $\pm$ 7.32	89.45 $\pm$ 16.23	80.52 $\pm$ 7.18
	4	119.70 $\pm$ 2.78	111.68 $\pm$ 12.23	128.61 $\pm$ 10.99	97.92 $\pm$ 13.20
	5	144.03 $\pm$ 2.89	136.05 $\pm$ 8.53	143.68 $\pm$ 10.39	137.89 $\pm$ 10.96
	6	172.10 $\pm$ 3.10	170.16 $\pm$ 14.87	179.05 $\pm$ 15.66	166.03 $\pm$ 16.15
	7	239.34 $\pm$ 2.91	234.12 $\pm$ 8.15	243.01 $\pm$ 12.15	260.59 $\pm$ 13.20
	8	299.44 $\pm$ 3.37	307.24 $\pm$ 5.40	305.88 $\pm$ 15.66	292.25 $\pm$ 9.56

Table 8. The estimated TSS of the extra concentration samples with the proposed method. It shows mean values $\pm$ standard deviation of the TSS estimated by the CNN and the MLR for each color dataset for paprika samples.

Samples	TSS (g/L ⁻¹)	Colors			
		White	Red	Blue	Green
	0.05	0.05 $\pm$ 0.011	0.02 $\pm$ 0.011	0.004 $\pm$ 0.051	0.013 $\pm$ 0.026
	0.15	0.14 $\pm$ 0.018	0.11 $\pm$ 0.015	0.08 $\pm$ 0.049	0.14 $\pm$ 0.056
	0.25	0.24 $\pm$ 0.016	0.20 $\pm$ 0.021	0.17 $\pm$ 0.049	0.16 $\pm$ 0.050
	0.35	0.33 $\pm$ 0.017	0.30 $\pm$ 0.018	0.29 $\pm$ 0.058	0.26 $\pm$ 0.049
	0.45	0.44 $\pm$ 0.011	0.36 $\pm$ 0.040	0.42 $\pm$ 0.039	0.32 $\pm$ 0.040
	0.55	0.54 $\pm$ 0.015	0.45 $\pm$ 0.030	0.49 $\pm$ 0.040	0.46 $\pm$ 0.040
	0.65	0.63 $\pm$ 0.018	0.60 $\pm$ 0.040	0.61 $\pm$ 0.050	0.58 $\pm$ 0.048
	0.75	0.72 $\pm$ 0.019	0.80 $\pm$ 0.029	0.80 $\pm$ 0.046	0.77 $\pm$ 0.040

Table 9. The estimated turbidity of the extra concentration samples with the proposed method. It shows mean values $\pm$ standard deviation of the turbidity estimated by the CNN and the MLR for each color dataset for paprika samples.

Samples	Turbidity (NTU)	Colors			
		White	Red	Blue	Green
	12	12.69 $\pm$ 2.97	6.31 $\pm$ 2.94	1.16 $\pm$ 13.63	3.56 $\pm$ 6.88
	42	39.90 $\pm$ 4.83	31.48 $\pm$ 3.99	21.89 $\pm$ 13.20	41.51 $\pm$ 15.12
	73	71.55 $\pm$ 4.32	61.46 $\pm$ 5.56	52.53 $\pm$ 13.25	48.35 $\pm$ 13.20
	105	99.43 $\pm$ 4.59	91.05 $\pm$ 4.88	89.44 $\pm$ 15.68	80.51 $\pm$ 13.12
	134	132.30 $\pm$ 2.91	111.68 $\pm$ 10.82	128.61 $\pm$ 10.74	97.92 $\pm$ 10.77
	156	153.95 $\pm$ 4.05	136.04 $\pm$ 8.20	143.68 $\pm$ 10.80	137.89 $\pm$ 10.69
	207	199.26 $\pm$ 4.91	170.15 $\pm$ 10.93	179.04 $\pm$ 13.47	166.02 $\pm$ 12.96
	272	254.33 $\pm$ 5.05	307.24 $\pm$ 8.04	305.88 $\pm$ 12.28	292.25 $\pm$ 10.90

The results showed that the proposed method can be improved by including more classes with small concentrations in the CNN training. This can be noticed by comparing the test results for the fish feed and paprika samples. For example, with white light illumination, the measurement error is reduced from 2.53% to 1.76% for TSS estimation, and from 9.84% to 2.79% for turbidity estimation. These results provide evidence for the effectiveness of the proposed method, and indicate high resolution and accuracy. Nevertheless, despite the high performance this method offers, its associated limitations should be recognized. Among these are the type of samples that can be measured. The proposed method was tested with commercial fish feed and paprika as suspended solids, which do not represent all types of suspended solids. There are other aquatic environments such as river water, domestic and industrial wastewater, drinking water, among others, where the size of the particles may be smaller than that of the fish feed or paprika mass. However, as CNN analyzes the images in depth, with the high-level convolutional layers calculating all the differences between the images, their potential for application to other types of suspended solids is promising. This is because the CNN image analysis not only differentiates particle distribution, but also contrast, brightness, and color. This means that further research should explore this issue, considering other types of samples such as river water, domestic and industrial wastewater, and potable water, among others. In addition, this research is expected to become a prototype, in which we would have an encapsulation box with LED lighting included. In the box, we could place the sample and the smartphone in certain fixed positions and perform the measurement. In a future work, it could be tested if the LED flash (white light) of the smartphone could be used because the white dataset presented the best performance.

4. Conclusions

In this paper, a novel method was proposed to estimate TSS and turbidity values in liquid samples using a CNN and an MLR together with a smartphone. The main conclusions of the study follow:

- The CNN and the MLR developed can estimate the TSS and turbidity values using images recorded by a common smartphone (Android and IOS systems).
- The proposed method is capable of estimating concentration values for unknown classes (validated with samples of unknown concentrations for the CNN).
- The proposed method is capable of estimating TSS and turbidity measurements in homogeneous material samples with different particle size, which are in motion or suspended. The use of a mixer is to keep all the particles suspended, for preventing them from settling, and to allow for the suspended particles to be recorded by the camera.
- This method of measuring turbidity and TSS is inexpensive and reduces human intervention. The results show the effectiveness of the proposed method, indicating high accuracy of 98.24% and 97.20% for TSS and turbidity measurements, respectively,

which is high compared to measurements made with commercial turbidimeters, and prevailing algorithms of machine learning.

- Once the CNN model is trained and the MLR is fitted, the algorithm can be used on a smartphone or other devices with a lower cost Nvidia GPU. In this research, the execution of the TSS and turbidity estimation algorithm was validated on a smartphone (Huawei Nova 3) using “Pydroid 3”, and the same results were obtained according to the Google Colaboratory service.
- As further work, a range extension (adding larger TSS values) and an expansion of the training dataset (more images) could be performed to achieve better method performance. Moreover, further research should also investigate heterogeneous material samples such as river water, domestic and industrial wastewater, potable water, and different sediments on the seabed, among others. In addition, it is expected that this research study will be helpful for developing a device for actual applications; for example, a mobile device where both the camera and the liquid sample are encapsulated, thus avoiding influence on the measurements by external conditions of light and background.

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

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: The authors appreciate the help given by Ana Isabel Veyna Gomez, a collaborator from the Water Geochemistry Laboratory at the Autonomous University of Zacatecas. Furthermore, the authors thank Ivanna Sofia Renteria Estrada for her editorial suggestions. Finally, the authors also thank the reviewers for their helpful comments and suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Avnimelech, Y. Feeding with Microbial Flocs by Tilapia in Minimal Discharge Bio-Flocs Technology Ponds. *Aquaculture* **2007**, *264*, 140–147. [\[CrossRef\]](#)
2. Qin, S.; Cai, X.; Ma, L. A Novel Light Fluctuation Spectrum Method for In-Line Particle Sizing. *Front. Energy* **2012**, *6*, 89–97. [\[CrossRef\]](#)
3. Bin Omar, A.F.; Bin MatJafri, M.Z. Turbidimeter Design and Analysis: A Review on Optical Fiber Sensors for the Measurement of Water Turbidity. *Sensors* **2009**, *9*, 8311–8335. [\[CrossRef\]](#) [\[PubMed\]](#)
4. Yang, Y.; Wang, H.; Cao, Y.; Gui, H.; Liu, J.; Lu, L.; Cao, H.; Yu, T.; You, H. The Design of Rapid Turbidity Measurement System Based on Single Photon Detection Techniques. *Opt. Laser Technol.* **2015**, *73*, 44–49. [\[CrossRef\]](#)
5. Murphy, K.; Heery, B.; Sullivan, T.; Zhang, D.; Paludetti, L.; Lau, K.T.; Diamond, D.; Costa, E.; O'Connor, N.; Regan, F. A Low-Cost Autonomous Optical Sensor for Water Quality Monitoring. *Talanta* **2015**, *132*, 520–527. [\[CrossRef\]](#)
6. Wang, H.; Hu, J.; Wan, W.; Gui, H.; Qin, F.; Yu, F.; Liu, J.; Lü, L. A Wide Dynamic Range and High Resolution All-Fiber-Optic Turbidity Measurement System Based on Single Photon Detection Technique. *Measurement* **2019**, *134*, 820–824. [\[CrossRef\]](#)
7. Toivanen, T.; Koponen, S.; Kotovirta, V.; Molinier, M.; Chengyuan, P. Water Quality Analysis Using an Inexpensive Device and a Mobile Phone. *Environ. Syst. Res.* **2013**, *2*, 9. [\[CrossRef\]](#)
8. Gillett, D.; Marchiori, A. A Low-Cost Continuous Turbidity Monitor. *Sensors* **2019**, *19*, 3039. [\[CrossRef\]](#)
9. Azman, A.A.; Rahiman, M.H.F.; Taib, M.N.; Sidek, N.H.; Bakar, I.A.A.; Ali, M.F. A Low Cost Nephelometric Turbidity Sensor for Continual Domestic Water Quality Monitoring System. In Proceedings of the 2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), Selangor, Malaysia, 22 October 2016; IEEE: Selangor, Malaysia, 2016; pp. 202–207.
10. Godoy, A.C.; Nakano, A.Y.; Siepmann, D.A.B.; Schneider, R.; Pfirmer, F.W.D.; Santos, O.O. Snapshots Analyses for Turbidity Measurements in Water. *Water Air Soil Pollut.* **2018**, *229*, 378. [\[CrossRef\]](#)

11. Zhu, Y.; Cao, P.; Liu, S.; Zheng, Y.; Huang, C. Development of a New Method for Turbidity Measurement Using Two NIR Digital Cameras. *ACS Omega* **2020**, *5*, 5421–5428. [\[CrossRef\]](#)
12. Gimenez, A.J.; Medellín-Rodríguez, F.J.; Contreras-Martínez, L.M.; López-Romero, J.M.; Sánchez, I.C.; Villasenor-Ortega, F.; Luna-Barcenas, G. Turbidimetry by Image Degradation Analysis. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 7574–7579. [\[CrossRef\]](#)
13. Gu, K.; Zhang, Y.; Qiao, J. Random Forest Ensemble for River Turbidity Measurement from Space Remote Sensing Data. *IEEE Trans. Instrum. Meas.* **2020**, *69*, 9028–9036. [\[CrossRef\]](#)
14. Mullins, D.; Coburn, D.; Hannon, L.; Jones, E.; Clifford, E.; Glavin, M. A Novel Image Processing-Based System for Turbidity Measurement in Domestic and Industrial Wastewater. *Water Sci. Technol.* **2018**, *77*, 1469–1482. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Bayram, A.; Yalcin, E.; Demic, S.; Gunduz, O.; Solmaz, M.E. Development and Application of a Low-Cost Smartphone-Based Turbidimeter Using Scattered Light. *Appl. Opt.* **2018**, *57*, 5935–5940. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Leeuw, T.; Boss, E. The HydroColor App: Above Water Measurements of Remote Sensing Reflectance and Turbidity Using a Smartphone Camera. *Sensors* **2018**, *18*, 256. [\[CrossRef\]](#)
17. Maeda-Gutiérrez, V.; Galván-Tejada, C.E.; Zanella-Calzada, L.A.; Celaya-Padilla, J.M.; Galván-Tejada, J.I.; Gamboa-Rosales, H.; Luna-García, H.; Magallanes-Quintanar, R.; Guerrero-Mendez, C.A.; Olvera-Olvera, C.A. Comparison of Convolutional Neural Network Architectures for Classification of Tomato Plant Diseases. *Appl. Sci.* **2020**, *10*, 1245. [\[CrossRef\]](#)
18. Jafarzadeh Ghouschi, S.; Ranjbarzadeh, R.; Najafabadi, S.A.; Osgooei, E.; Tirkolaee, E.B. An Extended Approach to the Diagnosis of Tumour Location in Breast Cancer Using Deep Learning. *J. Ambient. Intell. Humaniz. Comput.* **2021**. [\[CrossRef\]](#)
19. Romanuke, V. A Prototype Model for Semantic Segmentation of Curvilinear Meandering Regions by Deconvolutional Neural Networks. *Appl. Comput. Syst.* **2020**, *25*, 62–69. [\[CrossRef\]](#)
20. Beemaraj, R.K.; Chandra Sekar, M.S.; Vijayan, V. Computer Vision Measurement and Optimization of Surface Roughness Using Soft Computing Approaches. *Trans. Inst. Meas. Control* **2020**, *42*, 2475–2481. [\[CrossRef\]](#)
21. Vijayan, V.; Senthilkumar, G. Performance Analysis of Surface Roughness Modeling Using Soft Computing Approaches. *Appl. Math. Inf. Sci.* **2018**, *12*, 1209–1217. [\[CrossRef\]](#)
22. Brunner, V.; Siegl, M.; Geier, D.; Becker, T. Challenges in the Development of Soft Sensors for Bioprocesses: A Critical Review. *Front. Bioeng. Biotechnol.* **2021**, *9*, 730. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Yuan, X.; Qi, S.; Shardt, Y.A.W.; Wang, Y.; Yang, C.; Gui, W. Soft Sensor Model for Dynamic Processes Based on Multichannel Convolutional Neural Network. *Chemom. Intell. Lab. Syst.* **2020**, *203*, 104050. [\[CrossRef\]](#)
24. Wang, K.; Shang, C.; Liu, L.; Jiang, Y.; Huang, D.; Yang, F. Dynamic Soft Sensor Development Based on Convolutional Neural Networks. *Ind. Eng. Chem. Res.* **2019**, *58*, 11521–11531. [\[CrossRef\]](#)
25. Hajian, A.; Styles, P. Artificial Neural Networks. In *Application of Soft Computing and Intelligent Methods in Geophysics*; Springer: Berlin/Heidelberg, Germany, 2018; pp. 3–69.
26. Lopez-Betancur, D.; Duran, R.B.; Guerrero-Mendez, C.; Rodríguez, R.Z.; Anaya, T.S. Comparison of Convolutional Neural Network Architectures for COVID-19 Diagnosis. *Comput. Sist.* **2021**, *25*, 601–615.
27. Guerrero-Mendez, C.; Saucedo-Anaya, T.; Moreno, I.; Araiza-Esquivel, M.; Olvera-Olvera, C.; Lopez-Betancur, D. Digital Holographic Interferometry without Phase Unwrapping by a Convolutional Neural Network for Concentration Measurements in Liquid Samples. *Appl. Sci.* **2020**, *10*, 4974. [\[CrossRef\]](#)
28. Batista, L.V. Turbidity Classification of the Paraopeba River Using Machine Learning and Sentinel-2 Images. *IEEE Lat. Am. Trans.* **2022**, *20*, 799–805. [\[CrossRef\]](#)
29. Ferentinos, K.P. Deep Learning Models for Plant Disease Detection and Diagnosis. *Comput. Electron. Agric.* **2018**, *145*, 311–318. [\[CrossRef\]](#)
30. Han, X.; Zhong, Y.; Cao, L.; Zhang, L. Pre-Trained AlexNet Architecture with Pyramid Pooling and Supervision for High Spatial Resolution Remote Sensing Image Scene Classification. *Remote Sens.* **2017**, *9*, 848. [\[CrossRef\]](#)
31. Zhang, Z.; Ning, G.; Cen, Y.; Li, Y.; Zhao, Z.; Sun, H.; He, Z. Progressive Neural Networks for Image Classification. *arXiv* **2018**, arXiv:1804.09803.
32. Krizhevsky, A. One Weird Trick for Parallelizing Convolutional Neural Networks. *arXiv* **2014**, arXiv:1404.5997.
33. Avnimelech, Y. Biofloc Technology. In *A Practical Guide Book*; The World Aquaculture Society: Baton Rouge, LA, USA, 2009; Volume 182.
34. Lopez-Betancur, D.; Moreno, I.; Guerrero-Mendez, C.; Gómez-Meléndez, D.; Macías, P.D.M.J.; Olvera-Olvera, C. Effects of Colored Light on Growth and Nutritional Composition of Tilapia, and Biofloc as a Food Source. *Appl. Sci.* **2020**, *10*, 362. [\[CrossRef\]](#)
35. Begum, N.; Islam, M.S.; Haque, A.; Suravi, I.N. Growth and Yield of Monosex Tilapia *Oreochromis niloticus* in Floating Cages Fed Commercial Diet Supplemented with Probiotics in Freshwater Pond, Sylhet. *Bangladesh J. Zool.* **2017**, *45*, 27–36. [\[CrossRef\]](#)
36. Deng, X.; Liu, Q.; Deng, Y.; Mahadevan, S. An Improved Method to Construct Basic Probability Assignment Based on the Confusion Matrix for Classification Problem. *Inf. Sci.* **2016**, *340–341*, 250–261. [\[CrossRef\]](#)
37. Qiu, S.; Xu, H.; Deng, J.; Jiang, S.; Lu, L. Transfer Convolutional Neural Network for Cross-Project Defect Prediction. *Appl. Sci.* **2019**, *9*, 2660. [\[CrossRef\]](#)
38. Sammut, C.; Webb, G.I. *Encyclopedia of Machine Learning*; Springer: New York, NY, USA, 2011; ISBN 978-0-387-30768-8.
39. Schwarz, M.; Schulz, H.; Behnke, S. RGB-D Object Recognition and Pose Estimation Based on Pre-Trained Convolutional Neural Network Features. In Proceedings of the 2015 IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 26–30 May 2015; IEEE: Seattle, WA, USA, 2015; pp. 1329–1335.

40. Hannouche, A.; Chebbo, G.; Ruban, G.; Tassin, B.; Lemaire, B.J.; Joannis, C. Relationship between Turbidity and Total Suspended Solids Concentration within a Combined Sewer System. *Water Sci. Technol.* **2011**, *64*, 2445–2452. [[CrossRef](#)]
41. Rügner, H.; Schwientek, M.; Beckingham, B.; Kuch, B.; Grathwohl, P. Turbidity as a Proxy for Total Suspended Solids (TSS) and Particle Facilitated Pollutant Transport in Catchments. *Environ. Earth Sci.* **2013**, *69*, 373–380. [[CrossRef](#)]
42. Schwarz, K.; Gocht, T.; Grathwohl, P. Transport of Polycyclic Aromatic Hydrocarbons in Highly Vulnerable Karst Systems. *Environ. Pollut.* **2011**, *159*, 133–139. [[CrossRef](#)]
43. Hach USA DR900 Multiparameter Portable Colorimeter—Overview. Available online: <https://www.hach.com/dr900-multiparameter-portable-colorimeter/product?id=15684103251> (accessed on 1 August 2019).
44. Lloyd, D.S. Turbidity as a Water Quality Standard for Salmonid Habitats in Alaska. *N. Am. J. Fish. Manag.* **1987**, *7*, 34–45. [[CrossRef](#)]