

Article

Development of an Analog Gauge Reading Solution Based on Computer Vision and Deep Learning for an IoT Application

João Peixoto ^{1,2} , João Sousa ^{1,2} , Ricardo Carvalho ², Gonçalo Santos ³, Joaquim Mendes ¹ , Ricardo Cardoso ²  and Ana Reis ^{1,2,*} 

¹ Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal

² INEGI—Institute of Science and Innovation in Mechanical and Industrial Engineering, 4200-465 Porto, Portugal

³ GlarVision S.A., 2400-441 Leiria, Portugal

* Correspondence: areis@inegi.up.pt

Abstract: In many industries, analog gauges are monitored manually, thus posing problems, especially in large facilities where gauges are often placed in hard-to-access or dangerous locations. This work proposes a solution based on a microcontroller (ESP32-CAM) and a camera (OV2640 with a 65° FOV lens) to capture a gauge image and send it to a local computer where it is processed, and the results are presented in a dashboard accessible through the web. This was achieved by first applying a Convolutional Neural Network (CNN) to detect the gauge with the CenterNet HourGlass104 model. After locating the dial, it is segmented using the circle Hough transform, followed by a polar transformation to determine the pointer angle using the pixel projection. In the end, the indicating value is determined using the angle method. The dataset used was composed of 204 gauge images split into train and test sets using a 70:30 ratio. Due to the small size of the dataset, a diverse set of data augmentations were applied to obtain high accuracy and a well-generalized gauge detection model. Additionally, the experimental results demonstrated adequate robustness and accuracy for industrial environments achieving an average relative error of 0.95%.

Keywords: computer vision; gauges; object detection; CNN; image processing; IoT



Citation: Peixoto, J.; Sousa, J.; Carvalho, R.; Santos, G.; Mendes, J.; Cardoso, R.; Reis, A. Development of an Analog Gauge Reading Solution Based on Computer Vision and Deep Learning for an IoT Application. *Telecom* **2022**, *3*, 564–580. <https://doi.org/10.3390/telecom3040032>

Academic Editors: Eunmok Yang and Srijana Acharya

Received: 30 September 2022

Accepted: 10 October 2022

Published: 14 October 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

Automation not only reduces human intervention in processes that often are monotonous and repetitive to humans but also increases productivity, accuracy, and quality of products and processes. Moreover, automated systems that use vision to implement human-like capabilities are present in fields such as medicine, inventory control and management, product traceability, commerce, agriculture, and production processes monitoring, which are usually connected to a cloud-based system [1]. Due to this reality, the Internet of Things (IoT) is increasingly important for enabling communication between devices through the internet; therefore, camera sensors can be a good tool to extract relevant information about the environment and communicate with different devices in real time using IoT technologies.

However, in many industrial fields, workers measure physical process values such as pressure, temperature, or voltage by manually reading analog gauges. Due to costs and technical issues, the upgrade to digital gauges that can communicate using wireless or wired connections is often not feasible. Moreover, the analog gauge has the advantage of a simple structure, strong anti-interference ability, and good water, dust, and freeze resistance [2]. Therefore, this topic of research is relevant to automate the monitoring of analog gauges in real-world situations.

These solutions can be divided into traditional approaches or Neural Network (NN) approaches. The traditional approaches to automatic gauge reading, as represented in Figure 1, are normally divided into pre-processing the images, finding the gauge in the

image, and segmenting it, detecting the pointer, detecting the scale marks, and reading the final value.

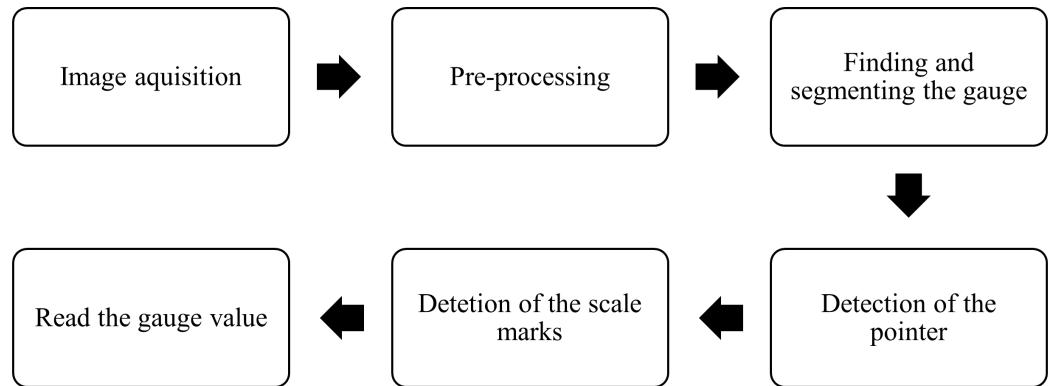


Figure 1. Flowchart of conventional approaches to automatic gauge reading.

For gauge segmentation, most of the traditional approaches were based on fitting the gauge shape. The most common method used through the years has been the Hough transform applied to circles. For example, in 2013, Gellaboina et al. [3] applied a local contrast normalization to deal with uneven lighting and the Canny edge detector to execute the final circle detection using the Hough transform. Region growing has also been applied in some research articles [4,5], taking advantage of the contrast between the white dial and background. This technique can perform poorly if the contrast is low or if there are glares in the dial, which can make the border of the gauge noisy. More recently, in 2019, Li et al. [6] proposed a technique to detect gauges by first using the Sobel filter to extract edges and then finding the border of the gauge by combining a line fitting algorithm with the Random Sample Consensus (RANSAC) method. With the information on the remaining edge points, they proposed a RANSAC ellipse algorithm to find the most fitted ellipse. This method showed to be especially good with images of gauges captured under angled viewpoints.

Straight-line fitting methods were also proposed by many researchers to detect the gauge pointer; these methods include the Hough transform and the least square method. The first method can achieve good results for the dials of gauges that have words and symbols but can be challenging in difficult light conditions [4,7,8]. However, the second method, in general, yields better results with less computational power [9–11]. Moreover, the image subtraction method has also been used to segment the region of the pointer [12,13]. This method is often performed before the straight-line fitting method to increase the robustness of the pointer detection by eliminating the background interference. This is achieved by subtracting images with roughly the same backgrounds with the pointer in different positions. The resulting image can have a lot of noise and produce unreliable results. In 2008, Zhao et al. [9] proposed a noble method to detect the pointer using image subtraction with a dynamic threshold to segment the image. For the scale marks and pointer recognition, the article introduces a technique named region-segmentation to partition the dial image and only process the desired area. Then, the least-squares and the central projection methods were used to detect the pointer and scale marks and determine the gauge value. This central projection method performs a projection from the feature points to the known gauge center, which results in projecting values associated with an angle. The angles that correspond to the largest number of projection points represent the pointer or even the scale marks. Furthermore, region growing has also been applied to segment the pointer [14], but as is the case for the segmentation of the gauges, the performance of this method is shown to be influenced by the lighting conditions and contrast between the pointer and the dial.

For the more difficult task of detecting scale, Yi et al. [15], in 2017, used the K-means clustering method to recognize the tick mark on the scale of an automobile dashboard based on attributes such as area, contour length, and the distance between the centroid of tick marks and pointer rotation center, but this is susceptible to noise. Two years later,

Lauridsen et al. [10] expanded this idea by using parametric distributions to recognize the pointer and scale marks. The parameters were the mass of the objects (number of pixels), the distance to the gauge center, the orientation inaccuracy, and the rotated bounding box ratio. Moreover, Gellaboina et al. [3] presented a method to detect the scale marks and pointer by unwinding the circular gauge, which is achieved by applying the polar coordinates transformation. By using the technique of column projection of the pixels, they detected a bigger spike that represented the pointer and small spikes that were the scale marks. This method was made more robust by Zheng et al. [7] by using color correction and perspective transformation to obtain the front view of the image taken from an arbitrary camera angle. Chi et al. [4] also used the polar coordinates system with an improved central projection to detect the circular scale region.

The final step to automate the reading of analog gauges is determining the gauge value. There are two main techniques: the angle method and the distance method. The angle method determines the value of the gauge by the deflection between the zero-scale mark and the pointer. This technique is simple to use but cannot be applied to non-linear scales. The distance method uses the distance between the nearest scale mark and the pointer to determine the gauge value. This method can be applied to non-linear scales but requires the definition of all the scale marks.

In the last few years, the fast development of deep neural networks has led to the research and development of systems capable of reading gauges using this technology. For example, Liu et al. [16] used a Faster R-CNN model to localize gauges in an image captured by an inspection robot; then, if needed, the camera was adjusted according to the detection box. After this, they used more conventional methods, such as the Hough transform, to read the final value. More recently, end-to-end approaches based on CNNs have been proposed, such as by Cai et al. [17], who described a novel method that generated synthetic images by separating the dial and pointer of the gauge image. The pointer is rotated to combine with the dial, creating a synthetic image labeled with the reading value to train the CNN model. Furthermore, Zuo et al. [18] proposed a method by first using a Mask-RCNN algorithm to classify the gauge and segment the pointer to later use the line fitting algorithm called Principal Component Analysis (PCA) and read the gauge value. In general, end-to-end approaches are promising but are still limited due to the difficulty of generalization.

In this article, a method was proposed based on traditional computer vision techniques complemented with new neural network approaches to read circular gauges even in challenging conditions. Therefore, the primary aim was to develop a robust algorithm capable of reading a large variety of circular gauges.

This article is organized into four sections. Section 1 is the introduction where the context is explained and the state-of-the-art work is analyzed. Section 2 details the proposed solution as well as some tests performed in gauge images of well-known computer vision techniques to help select the best performing. Subsequently, in Section 3, the solution was evaluated in different scenarios to emulate real-world applications. The results are analyzed and compared to some state-of-the-art solutions. Additionally, an IoT application with a camera sensor was tested as a proof-of-concept. Finally, Section 4 provides a final discussion about the work developed and addresses the limitations.

2. Methodological Approach

The proposed solution to automatic read gauges follows the main steps shown in Figure 2. First, the image is captured, and the gauges are detected using a TensorFlow object detection algorithm [19]. After this, traditional approaches of computer vision are used, such as pre-processing and segmentation. These were applied to find the circular dial region and the gauge pointer. In the end, the gauge value was calculated.

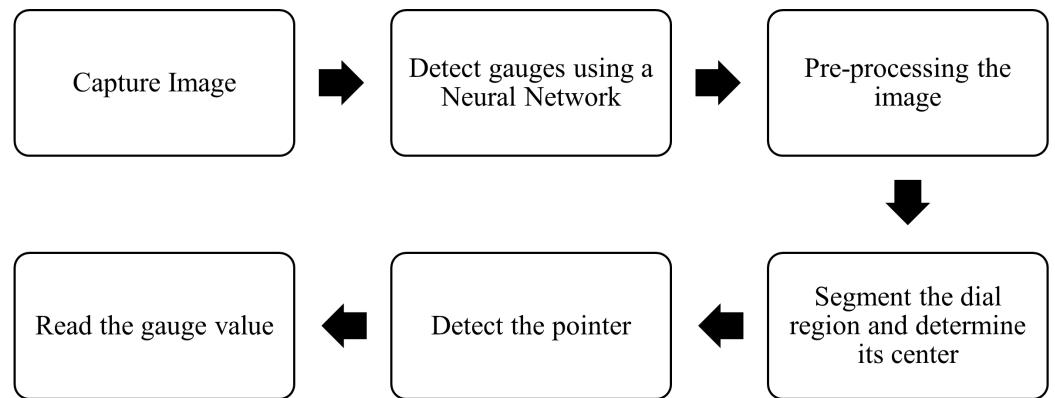


Figure 2. Flowchart of the proposed steps to read analog gauges.

2.1. Object Detection Based on CNN

The first step of the algorithm is the detection of the gauge in an image using a trained CNN. Therefore, a dataset was created with a small number of images—204 images from 3 different sources. Due to the limited number of available datasets, most images were taken from a phone camera in the shop floor of the Institute of Science and Innovation in Mechanical and Industrial Engineering (INEGI) and the other two sources, as shown in Table 1. The dataset was composed of different types of circular gauges, such as gauges with two pointers, multiple scales, multi-color dials, and gauges with different ranges of scale values. There were also images with long-distance and short-distance shots, various shooting angles, and even some with reflections in the dial. Moreover, to create a consistent dataset, all the images were modified to obtain a 1:1 aspect ratio and a resolution of 800×800 pixels, then they were manually split into a train and a test set using a 70:30 ratio.

Table 1. Dataset composition.

Source	Nr. of Images	Types of Gauges
INEGI	118	21
AWS [20]	37	1
Jcvision [21]	49	3
Total	204	25

Several data augmentation techniques were applied while training the CNN models to mitigate the problems of a small dataset. These augmentations can be divided into position and color augmentation. In position augmentation, the pixel position of an image is changed, such as cropping and padding images, which helps to imitate the zoom in and out effects. Another example is the common augmentation of flipping the image. On the other hand, color augmentation only alters the values of the pixels, maintaining the same position. Good examples of this are the manipulation of the contrast, saturation, hue, and brightness of the image but also adding noise, which is considered a powerful augmentation method that improves robustness without compromising accuracy [22]. The augmented images show notable changes from the original images, as illustrated in Figure 3, which help to create a diverse set of training images from a small dataset.

The model used to detect gauges was selected from a TensorFlow collection of object detection models pre-trained on the COCO 2017 dataset [19]. First, a selection was made by choosing only models with an input image resolution close to the resolution of the dataset (800×800 px), and the ones with the highest Mean Average Precision (mAP). Afterward, the models selected were trained until 7000 steps with the target dataset to determine the best-performing model. To avoid overfitting, the evaluation and train loss functions were continuously monitored to verify their convergence. These functions are shown in Figure A1, where it is noticeable that the training and the evaluation loss stay close and decrease over time, which indicates that the models do not have overfitting problems.

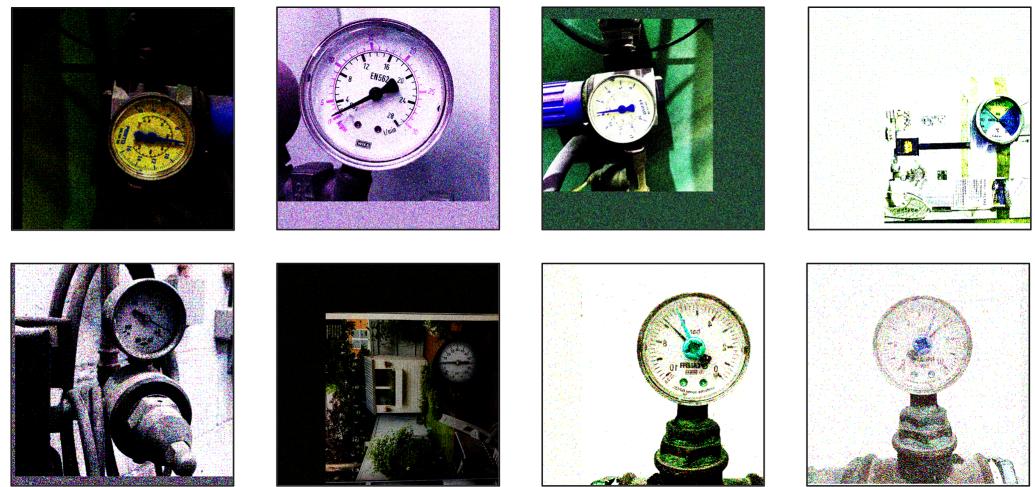


Figure 3. Examples of the data augmentation applied to some dataset images.

The results in Figure A2 demonstrate that most of the models reached close to peak performance in as early as 1000 steps, which can be explained by the small dataset that these models were trained on. Furthermore, the maximum mAP and recall for most models were above 90 % and very close to each other, which indicates that choosing any of these gauge detection algorithms would yield good results. However, for the proposed algorithm, the model selected was the CenterNet HourGlass104, which yielded the best performance out of all the models tested; these results are shown in Figure 4. Therefore, this model was implemented as the first step in the algorithm to find the Regions of interest (ROIs), as shown in Figure 5.

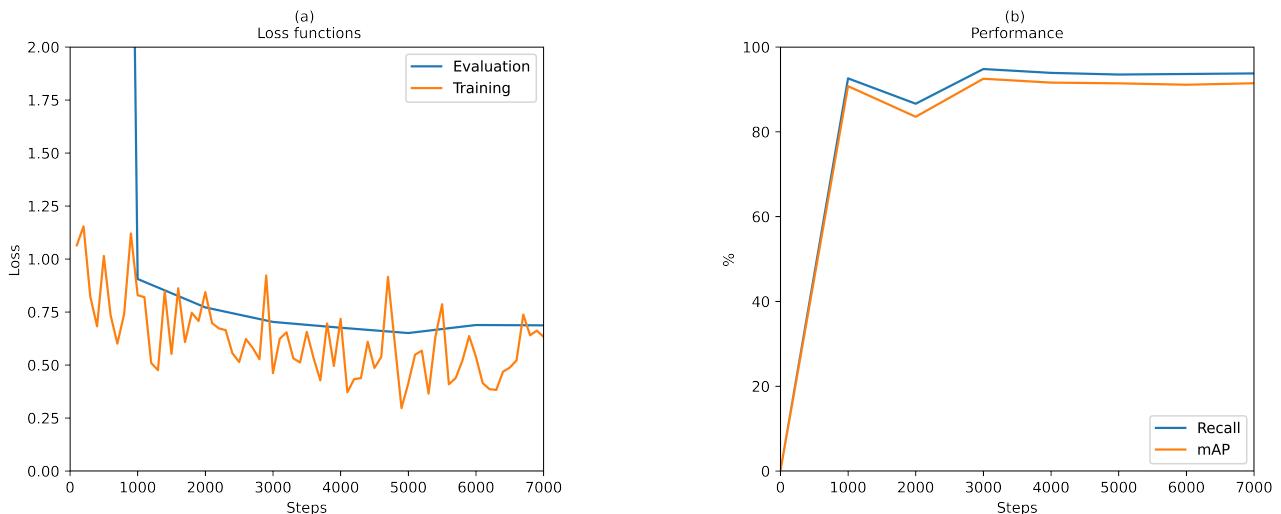


Figure 4. CenterNet HourGlass104 model Training. (a) Loss functions, (b) performance graphs.

2.2. Dial Segmentation and Center Localization

After gauge detection, the next step is to segment the dial region and determine its center. Therefore, two methods were analyzed for this purpose: the Hough transform and the region growing. Some relevant results of these tests are shown in Figure 6, which indicates that the region growing is sensitive to interferences such as reflections, shadows or even characters in the dial. On the contrary, the Hough transform performed better, especially in images with high interference, so this was the method applied in the algorithm to segment the dial, which also determines the center of the dial. To implement the Hough transform, the ROI needs to be pre-processed to increase its quality; therefore, the Contrast Limited Adaptive Histogram Equalization (CLAHE) algorithm was used to enhance the

illumination. Then, to eliminate some of the noise, the bilateral filter was applied, which also facilitates the circle detection of by preserving object edges.

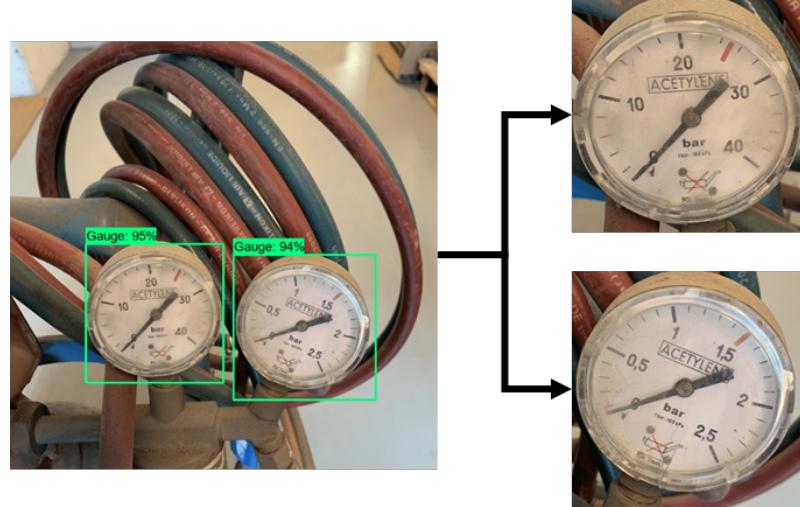


Figure 5. Gauge detection implementation.

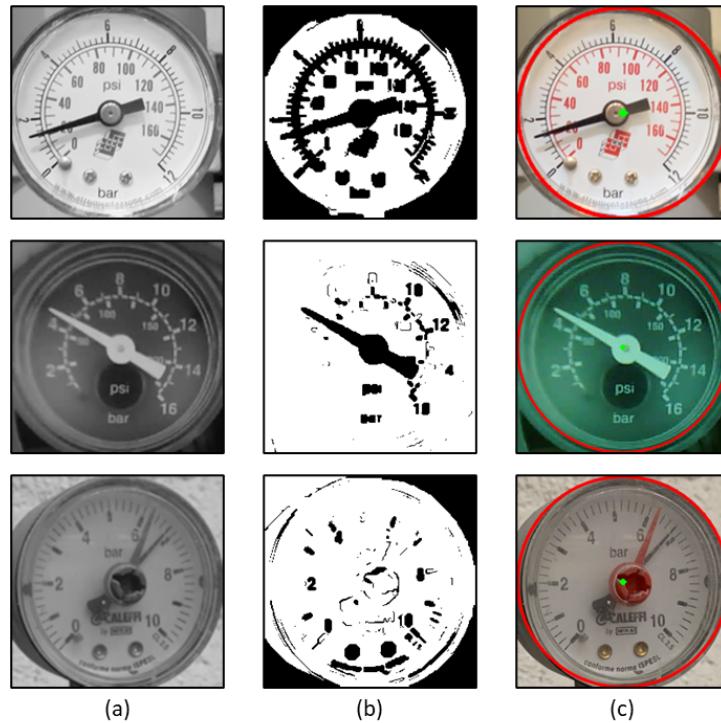


Figure 6. Dial gauge segmentation. (a) grayscale, RoI (b) region growing, (c) Hough transformation.

2.3. Image Uniformization

In order to make the algorithm more robust, a method to deal with the gauges with tell-tale pointers and black dials was applied. First, if the gauge has a tell-tale pointer, a segmentation method is applied by taking advantage of the color red or orange of these pointers. Therefore, a mask for these colors is created using the image in the Hue, Saturation, Value (HSV) color space. This segmentation, as shown in Figure 7, yields good results because these colors do not typically appear in the rest of the gauge. To uniformize these gauges images, the segmented tell-tale pointer is extracted from the original image to emulate a gauge with only the main pointer.

Since the following steps in the algorithm use the binarized ROI, the problem of black dial gauges can be solved by using this binarization process applied to the original ROI and its inverted colors image, as shown in Figure 8. This process results in two binarized images, but only one is correct, so the number of white pixels inside the dial region is calculated for both images, and the one with the highest number is considered the correct binarization and used for the next steps in the algorithm.

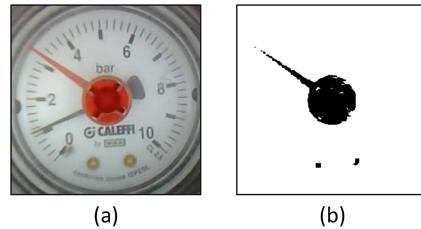


Figure 7. Tell-tale pointer segmentation. (a) original image, (b) tell-tale pointer segmented.

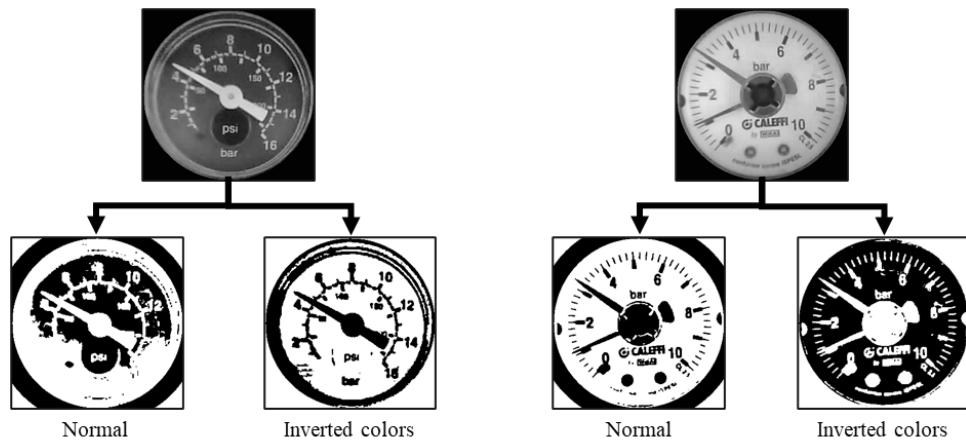


Figure 8. Binarization of the ROI and the inverted colors ROI.

2.4. Pointer Detection

In order to extract the pointer information, the gauge image often needs to be binarized. Experiments were performed by binarizing some dataset images with challenging conditions such as gauges with reflections and shadows in the dial, and even with a multi-color dial. One global and one local threshold method were performed, the Otsu method and the adaptive threshold method, respectively. These tests were performed after the images were transformed to grayscale and a median filter was applied. As shown in Figure 9, the Otsu method lost a lot of critical information about the image and was affected by uneven lighting and colors in the dial; on the contrary, the adaptive threshold was able to binarize the image the best in all analyzed situations. Therefore, the method applied was the adaptive threshold, even with the disadvantage of being more computationally expensive.

To compare the performance, three commonly used methods to detect pointers were applied. The first method was the region growing algorithm to segment the pointer; this method can be used before thinning and/or line fitting to determine the pointer direction. To test this method, the seed point was set as the center of the gauge. As shown in Figure 10, the algorithm applied was unreliable with a light color center gauge and uneven lighting, which makes the region expand beyond the pointer if its contrast is low compared to the dial's background.

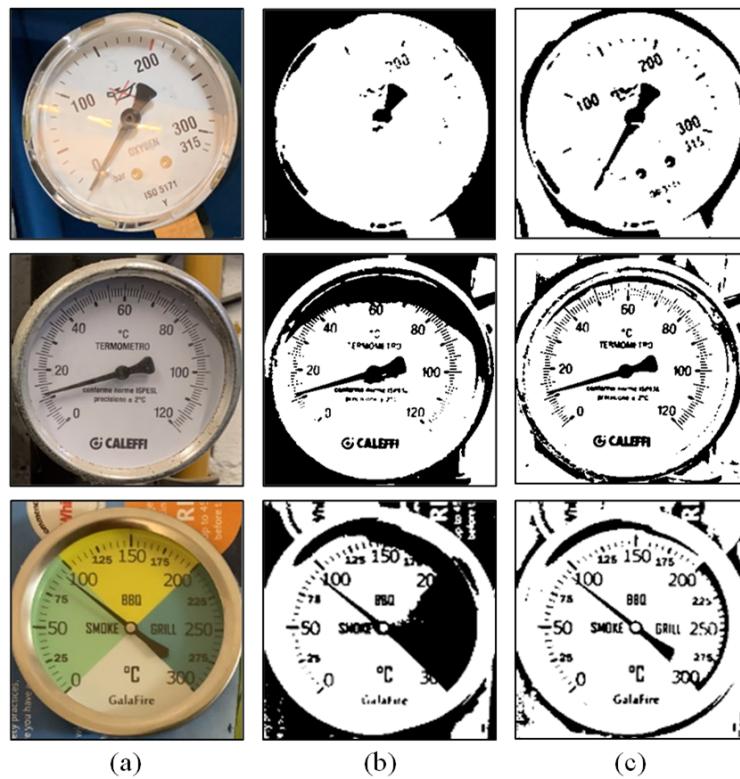


Figure 9. Application of different Binarization techniques. (a) Original image, (b) Otsu method, (c) adaptive threshold.

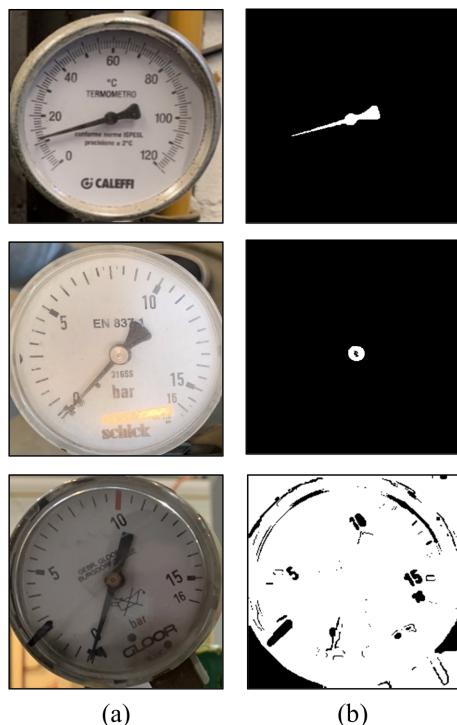


Figure 10. Application of the region growing. (a) Original image, (b) region growing segmentation.

An alternative is the line Hough transform, which takes advantage of the linear edges of the gauge pointer. First, some dataset images were binarized to apply the Canny edge detector, and then the line Hough transform was performed. The results, as shown in Figure 11, indicate that the edge detection is influenced by shadows and uneven lighting, and it has difficulty distinguishing the pointer lines and other lines, such as the ones from characters and scale lines.

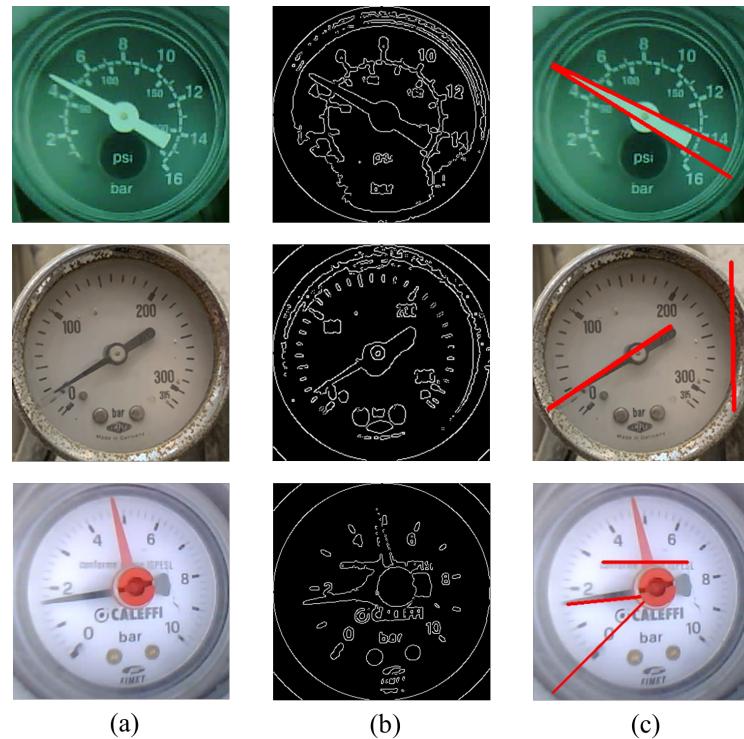


Figure 11. Application of the Hough line transform. (a) Original image, (b) Canny edge detection, (c) Hough line transform.

The third test was based on the characteristic that the pointer has an elongated tip going out from the gauge center. Therefore, a pixel projection was performed after the polar transformation of the binarized image. This method showed to be less affected under bad lighting but was still affected by interference in the dial, such as characters, scale marks, and even chromed bezels. This problem was minimized by removing some interference with image cropping by removing 10 % of the outer and inner circles. The angle of the maximum white pixels projection was defined to be the angle of the pointer. Some results of this method are shown in Figure 12.

After analyzing the methods tested, pixel projection achieved the best performance; therefore, this method was the one implemented after the adaptive threshold binarization, as illustrated in Figure 13. Moreover, this method was also applied to the tell-tale pointers using its segmented region, as represented in Figure 14. In order to have a more robust algorithm, a value that indicates the certainty of the pointer detection was developed, which indicates the percentage of the image height that the maximum projection occupies; this is especially important for gauges with a tell-tale pointer because they can cover the main pointer, making it impossible to read the gauge. A threshold of 70% for the certainty value was applied to detect the pointer, so only pointers above this value were considered correctly detected.

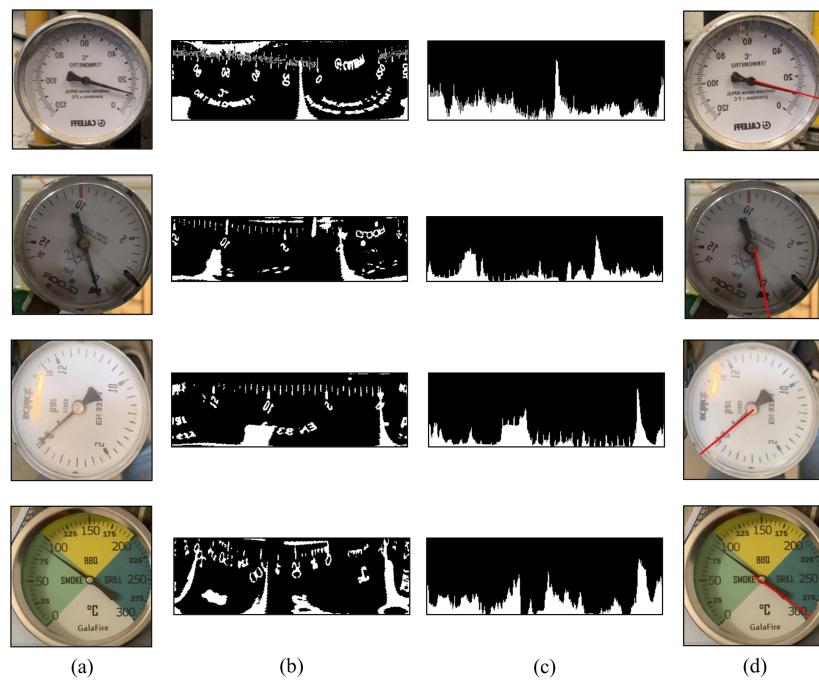


Figure 12. Application of the pointer detection using pixel projection. (a) ROI, (b) polar transformation, (c) pixel projection, (d) pointer detection.

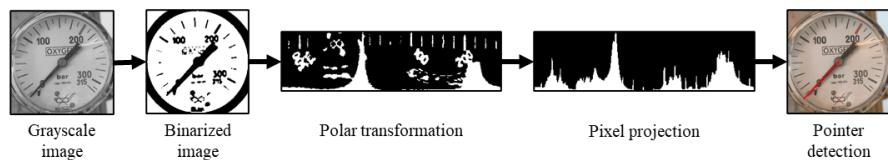


Figure 13. Flowchart of the pointer detection implementation.

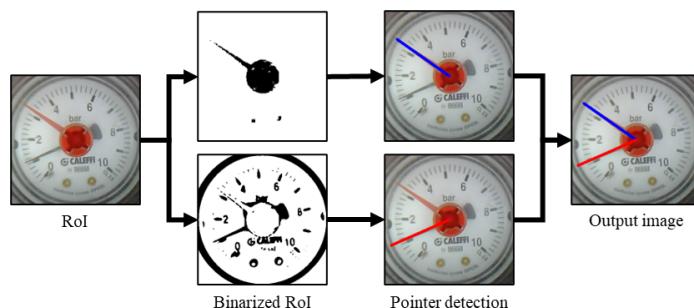


Figure 14. Flowchart of the detection of the pointer and tell-tale.

2.5. Determination of the Indicating Value of the Gauge

In this project, additional information was provided to the algorithm to read the gauge value. The information was related to the gauge scale (angle, value and linearity) and whether or not the gauge had a tell-tale pointer. Since these are fixed and specific to each individual gauge, the information was linked to a QR code placed close to the gauge.

Finally, the gauge indicating value was calculated using the angle method because it is simple to apply and it works for most gauges in the market. With the information about the scale and pointer angle, the final value is determined using Equation (1).

$$\text{IndicatingValue} = R \times \frac{\beta}{\alpha}, \quad (1)$$

where R is the range of values of the scale, β is the angle between the pointer and the minimum scale mark, and α is the angle between the minimum and maximum scale marks.

2.6. Proposed Final Solution

With all the methods explored, a final solution was proposed for automatic gauge reading, as shown in Figure 15. The algorithm begins with a trained CNN object detection to localize different gauges in an image. These RoIs are cropped, and a specific QR code is identified to read the needed information. The next step is the dial segmentation, which begins with the grayscale transformation and the bilateral filter application. Then, the circle Hough transform is applied. Since there are a lot of variations among gauges, the following steps help to uniformize gauge images. First the tell-tale pointer, if existent, is segmented using a mask in HSV color space with the red and orange colors. Then, to correct black dials, the ROI and its inverted colors image are binarized using the adaptive threshold, but only the image with the highest number of white pixels inside the dial is used. With this binarized image, the pointer angle is determined by first performing a polar coordinate transformation, followed by a pixel projection. In the end, the gauge indicating value is calculated using the angle method.

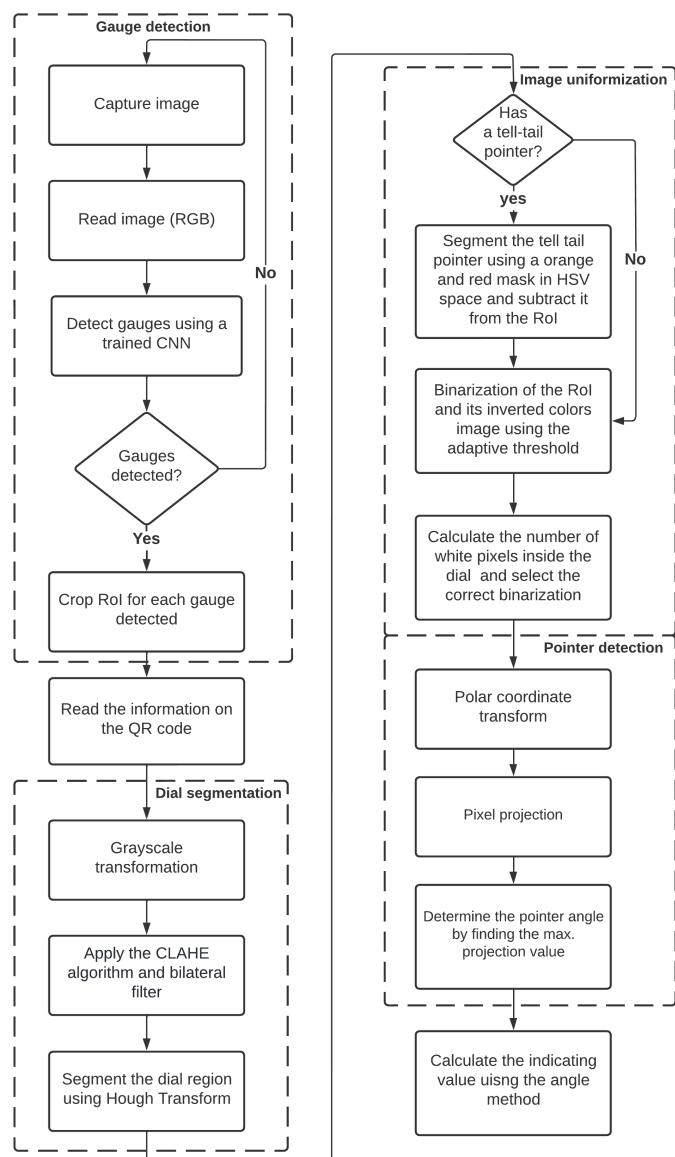


Figure 15. Proposed solution for automatic gauge reading.

3. Experimental Tests and Results Analysis

To evaluate the algorithm, first of all, some images were chosen from the dataset already created combined with extra images captured using an ESP32-CAM with a OV2640 camera and a FOV 160° lens to obtain images in different conditions, such as harsh lighting, reflections, and angled shots. As a result, these images were grouped, making it possible to compare the results and analyze the influence of the different shooting environments. The first group was named the ideal conditions, which only has images with good lighting and front-facing gauges, making it easier to read the indicating value. The other two groups were composed of images with bad lighting conditions and different shooting angles. Since some of the gauges present scale takeup (the portion of the scale between the position where the pointer is stopped and its true zero pressure position), the definition of the angle for the first scale mark was set in the true zero pressure position to avoid reading errors.

Furthermore, to make the tests consistent, only gauges with the same accuracy class were used, as these gauges followed the standard EN 837-1 [23]. Since most of the gauges in the images had an accuracy of 2.5%, these were the ones tested. For each gauge, the evaluation metric used was the relative error, E_i , between the read indicated value and the value determined by the algorithm, and then the average relative error, \bar{E} , was determined for all images and each group of shooting environment. These metrics are calculated using Equations (2) and (3).

$$E_i = \frac{|r_i - R_i|}{S} \times 100\%, \quad (2)$$

$$\bar{E} = \frac{\sum_{i=0}^n E_i}{n}, \quad (3)$$

where r_i is the reading value from the algorithm, R_i is the reading value determined visually, S is the range of the values, and n is the number of images.

Out of the 20 images tested, only one was read incorrectly, which demonstrates the capability of the algorithm even in difficult conditions. However, for one image, the algorithm did not perform correctly because the Hough transform detected the dial circle as slightly deviated, which affected the polar transform and consequently the pixel projection; therefore, the maximum projection value was incorrect at the tale of the pointer. The accuracy was generally high, with an average relative error of 0.67% in ideal conditions. In challenging conditions, the algorithm almost doubles the error, which indicates that the bad lighting and angled gauges negatively affect the algorithm; nevertheless, the error calculated was still relatively low, with a value of 0.95%, as shown in Table 2. Therefore, the accuracy, especially in ideal conditions, is suitable for less critical commercial uses, which cover a wide range of current analog gauges.

Table 2. Average relative error for different shooting environments.

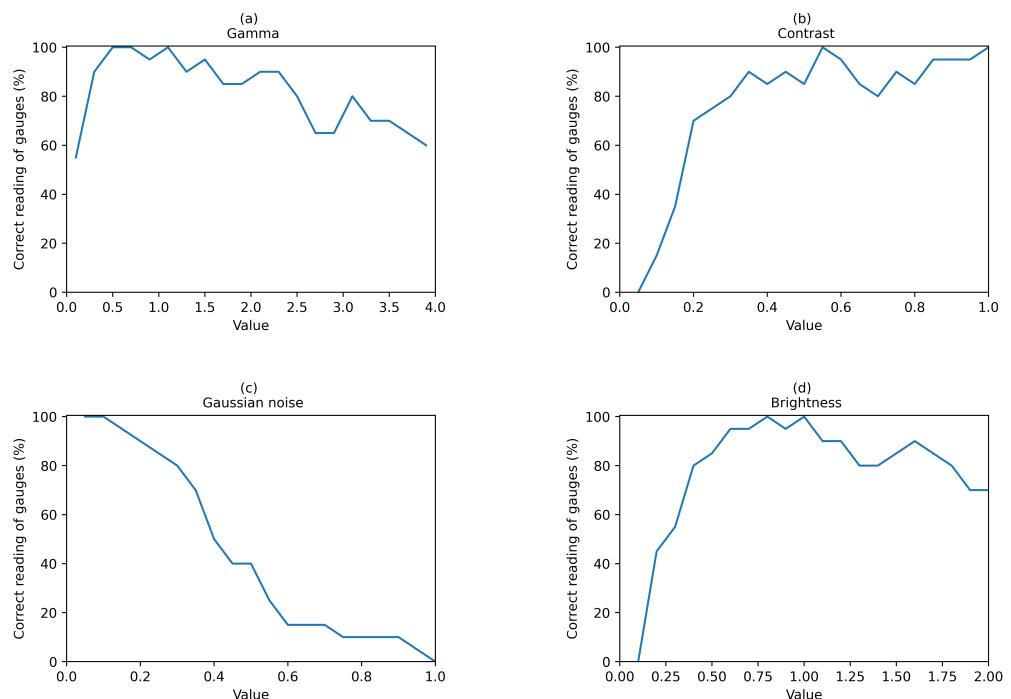
Shooting Environment	Average Relative Error (%)
Ideal conditions	0.67
Bad lighting conditions	1.28
Different shooting angles	1.23
All images	0.95

Furthermore, comparing the proposed solution with some state-of-the-art algorithms, the proposed algorithm performs relatively well, having a slightly better performance than most methods analyzed in Table 3. However, the Zuo et al. [18] solution performs the best out of all, which achieved high accuracy in difficult conditions of lighting and tilting gauges, but the tests were performed for only one gauge, which could result in low generalization and algorithm overfitting. This can explain the much better results in comparison with the other methods.

Table 3. Average relative error for state-of-the-art methods.

Methods	Average Relative Error (%)
Zheng et al. (2016) [7]	0.95
Wang et al. (2019) [2]	1.35
Zuo et al. (2020) [18]	0.17
Proposed solution	0.95

In addition, an experiment that was only focused on the robustness was executed. It used the same 20 images from the previous experiment with some defined image adjustments to observe if the proposed solution was capable of reading gauges correctly. Therefore, four different adjustments were made by altering the values of gamma, contrast, Gaussian noise, and brightness. For each, a set of 20 values were uniformly selected, where gamma was varied between 0.1 and 4, contrast and Gaussian noise between 0.05 and 1 and brightness between 0.1 and 2. In total, the algorithm was performed on 1600 images. The results shown in Figure 16 demonstrate that the proposed solution is capable of adapting to different environmental conditions but is most affected by the Gaussian noise. Table 4 shows that only the Gaussian noise had a low proportion of correctly read gauges, with a result of 43.3 %, and the other adjustments all show results above 75 %.

**Figure 16.** Algorithm robustness assessments with image adjustments.**Table 4.** Proportion of gauges correctly read for each image adjustment.

Gamma (%)	Contrast (%)	Gaussian Noise (%)	Brightness (%)
81.5	77.3	43.3	78.5

Implementation of an IoT Solution

Furthermore, an IoT implementation was developed as a proof-of-concept using an ESP32-CAM with a OV2640 camera and a 65° FOV lens to capture images and send them to a local computer that has the processing power to run the algorithm. Then, the information is sent to a web application to display the data as an interface for visualization. The communications are made via Wi-Fi using the Hypertext Transfer Protocol (HTTP), and the architecture is illustrated in Figure 17.

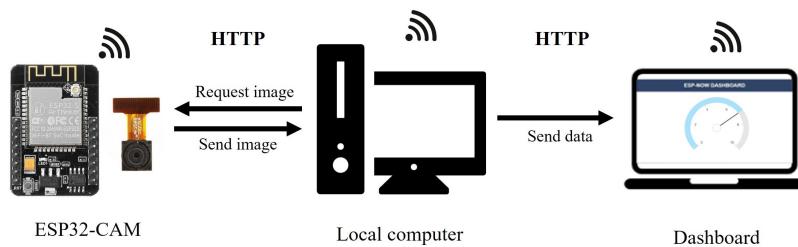


Figure 17. Architecture of the IoT implementation.

The camera was installed inside a cabinet pointing to two gauges of a gas cylinder. One of the gauges was selected to be monitored, so it was set with a QR code, as shown in Figure 18. Finally, a dashboard was developed to indicate the gauge value and tell-tale pointer value, if applied.



Figure 18. Setup of the IoT application.

4. Conclusions

The foremost purpose of this work was to digitize analog gauges, which was achieved by first applying a CNN gauge detection using the available pre-trained models of the TensorFlow API. Some of these models were trained with the target dataset to compare the results and choose the model with the best performance, which achieved a maximum mAP of 92.5% and recall of 94.8%. The following steps were based on conventional computer vision techniques, so after finding the ROI, the gauge dial is segmented using the circle Hough transform, and then the pointer angle is extracted using pixel projection subsequent to the ROI's polar transformation. In the end, the indicating value is determined using the angle method.

After choosing the methodology, some experiments were made to evaluate the algorithm performance, which resulted in an overall average relative error of 0.95%. The results also showed that bad lighting conditions and angled gauges have a negative impact on the algorithm performance since the average error practically doubled in these conditions. However, the solution proved to have sufficient accuracy for most noncritical applications. Moreover, the robustness evaluation demonstrated that, for wide values range of gamma, contrast, and brightness, the solution has high adaptability; the only exception was the Gaussian noise, which had a significant detriment on the image quality and, in turn, in the algorithm performance. Additionally, a proof-of-concept IoT implementation was developed, and it demonstrated the feasibility of this solution in a real-world application.

However, the solution has some limitations, such as the small-sized dataset that can contain personal biases and negatively affect the gauge detection even with data augmentation. Furthermore, the final solution is dependent on external information that could be extracted directly from the gauge image, if robust enough. Nevertheless, these limitations will be explored in future work.

Author Contributions: Conceptualization, J.P., J.S., and R.C. (Ricardo Carvalho); methodology, J.P. and J.S.; software, J.P.; validation, J.S., R.C. (Ricardo Carvalho) and J.M.; investigation, J.P., J.S., and R.C. (Ricardo Carvalho); writing—original draft preparation, J.P.; writing—review and editing, J.S., R.C. (Ricardo Carvalho) and J.M.; supervision, G.S., J.M., and R.C. (Ricardo Cardoso); project administration, A.R. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge the funding of Project POCI-01-0247-FEDER-047091–GRS: Glartek Retrofit Sensors, cofinanced by Programa Operacional Competitividade e Internacionalização (COMPETE 2020), through Fundo Europeu de Desenvolvimento Regional (FEDER).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors are grateful to GlarVision S.A for technical support and equipment.

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

Appendix A

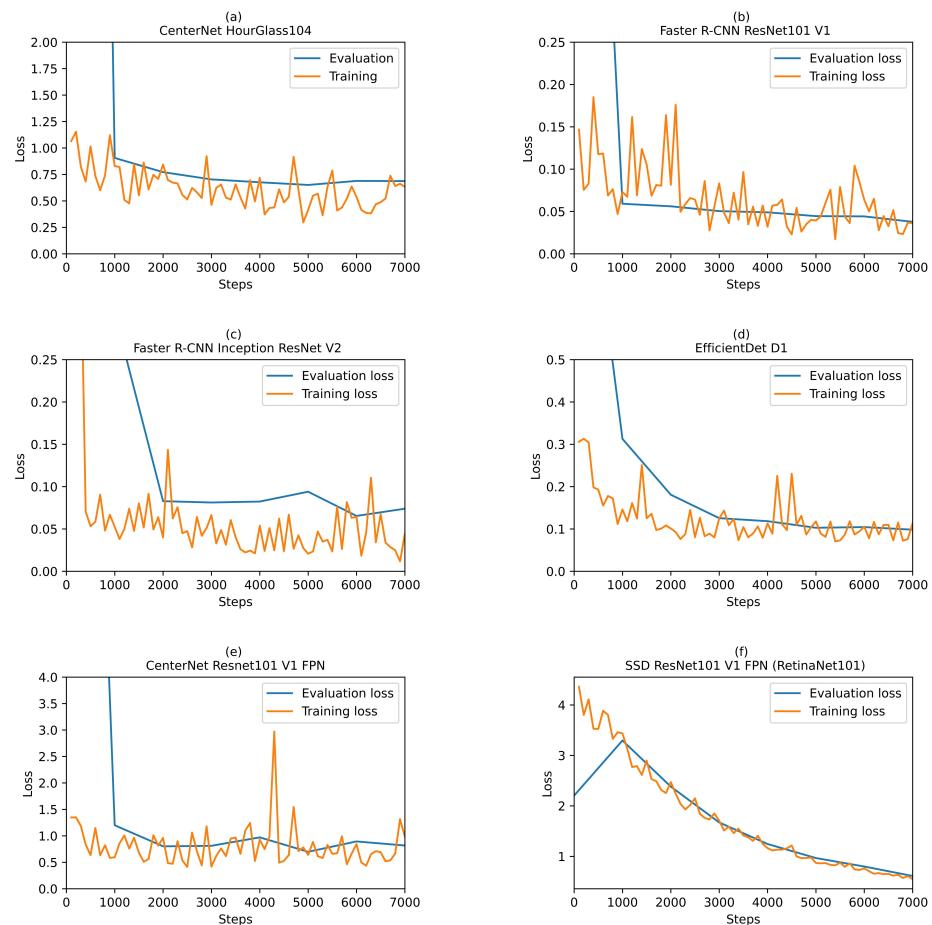


Figure A1. Evaluation and training loss functions of the CNN models trained.

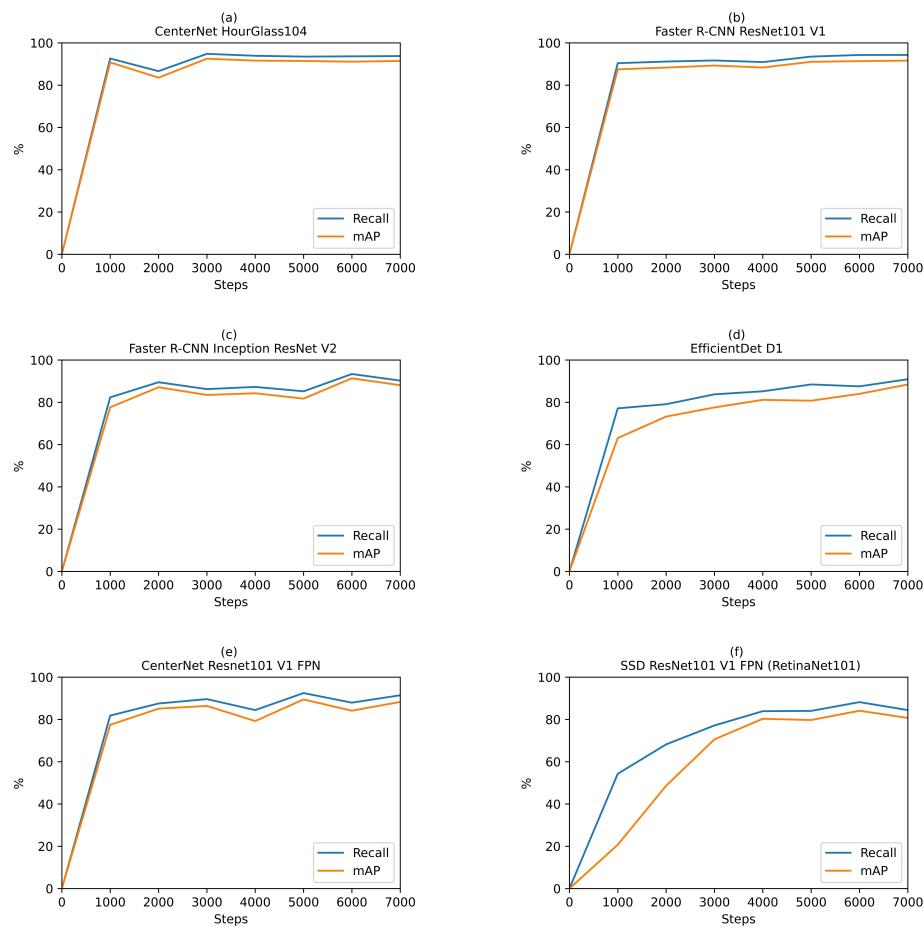


Figure A2. Recall and mAP evaluation graphs of the trained CNN models.

References

- Atieno, N. How Intelligent Automation Is Powering Machine Vision, 2021. Available online: <https://www.section.io/engineering-education/how-intelligent-automation-is-powering-machine-vision> (accessed on 25 June 2022).
- Wang, L.; Wang, P.; Wu, L.; Xu, L.; Huang, P.; Kang, Z. Computer vision based automatic recognition of pointer instruments: Data set optimization and reading. *Entropy* **2021**, *23*, 272. [CrossRef] [PubMed]
- Gellaboina, M.K.; Swaminathan, G.; Venkopalrao, V. Analog dial gauge reader for handheld devices. In Proceedings of the 2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA), Melbourne, VIC, Australia, 19–21 June 2013; pp. 1147–1150. [CrossRef]
- Chi, J.; Liu, L.; Liu, J.; Jiang, Z.; Zhang, G. Machine Vision Based Automatic Detection Method of Indicating Values of a Pointer Gauge. *Math. Probl. Eng.* **2015**, *2015*, 283629. [CrossRef]
- Zhuo, H.B.; Bai, F.Z.; Xu, Y.X. Machine vision detection of pointer features in images of analog meter displays. *Metrol. Meas. Syst.* **2020**, *27*, 589–599. [CrossRef]
- Li, B.; Yang, J.; Zeng, X.; Yue, H.; Xiang, W. Automatic Gauge Detection via Geometric Fitting for Safety Inspection. *IEEE Access* **2019**, *7*, 87042–87048. [CrossRef]
- Zheng, C.; Wang, S.; Zhang, Y.; Zhang, P.; Zhao, Y. A robust and automatic recognition system of analog instruments in power system by using computer vision. *Measurement* **2016**, *92*, 413–420. [CrossRef]
- Lee, D.; Kim, S.; Han, Y.; Lee, S.; Jeon, S.; Seo, D. Automatic Reading Analog gauge with Handheld device. In Proceedings of the 2020 IEEE International Conference on Consumer Electronics (ICCE), Las Vegas, NV, USA, 4–6 January 2020; pp. 1–3. [CrossRef]
- Feng, H.P.; Jun, Z. Application research of computer vision in the auto-calibration of dial gauges. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering, Wuhan, China, 12–14 December 2008; Volume 2, pp. 845–848. [CrossRef]
- Lauridsen, J.; Grassmé, J.; Pedersen, M.; Jensen, D.; Andersen, S.; Moeslund, T. Reading Circular Analogue Gauges using Digital Image Processing. In Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (Visigrapp 2019), Prague, Czech Republic, 25–27 February 2019; Volume 4, pp. 373–382. [CrossRef]

11. Wang, Q.; Fang, Y.; Wang, W.; Wu, M.; Wang, R.; Fang, Y. Research on automatic reading recognition of index instruments based on computer vision. In Proceedings of the 2013 3rd International Conference on Computer Science and Network Technology, Dalian, China, 12–13 October 2013; pp. 10–13. [[CrossRef](#)]
12. Alegria, F.C.; Serra, A.C. Automatic calibration of analog and digital measuring instruments using computer vision. *IEEE Trans. Instrum. Meas.* **2000**, *49*, 94–99. [[CrossRef](#)]
13. Li, D.; Li, W.; Yu, X.; Gao, Q.; Song, Y. Automatic Reading Algorithm of Substation Dial Gauges Based on Coordinate Positioning. *Appl. Sci.* **2021**, *11*, 6059. [[CrossRef](#)]
14. Liu, J.; Liu, Y.; Yu, L. Novel method of Automatic Recognition for Analog Measuring Instruments. In Proceedings of the 2015 6th International Conference on Manufacturing Science and Engineering, Guangzhou, China, 28–29 November 2015; [[CrossRef](#)]
15. Yi, M.; Yang, Z.; Guo, F.; Liu, J. A clustering-based algorithm for automatic detection of automobile dashboard. In Proceedings of the IECON 2017—43rd Annual Conference of the IEEE Industrial Electronics Society, Beijing, China, 29 October–1 November 2017; pp. 3259–3264. [[CrossRef](#)]
16. Liu, Y.; Liu, J.; Ke, Y. A detection and recognition system of pointer meters in substations based on computer vision. *Measurement* **2020**, *152*, 107333. [[CrossRef](#)]
17. Cai, W.; Ma, B.; Zhang, L.; Han, Y. A pointer meter recognition method based on virtual sample generation technology. *Measurement* **2020**, *163*, 107962. [[CrossRef](#)]
18. Zuo, L.; He, P.; Zhang, C.; Zhang, Z. A robust approach to reading recognition of pointer meters based on improved mask-RCNN. *Neurocomputing* **2020**, *388*, 90–101. [[CrossRef](#)]
19. Yu, H.; Chen, C.; Du, X.; Li, Y.; Rashwan, A.; Hou, L.; Jin, P.; Yang, F.; Liu, F.; Kim, J.; et al. TensorFlow Model Garden, 2020. Available online: <https://github.com/tensorflow/models> (accessed on 25 June 2022).
20. Read Analogue Gauges with Computer Vision on AWS, 2020. Available online: <https://aws-computer-vision.jacobcantwell.com> (accessed on 25 June 2022).
21. Howells, B.; Charles, J.; Cipolla, R. Real-time analogue gauge transcription on mobile phone. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Virtual, 19–25 June 2021; pp. 2369–2377. [[CrossRef](#)]
22. Shorten, C.; Khoshgoftaar, T.M. A survey on Image Data Augmentation for Deep Learning. *J. Big Data* **2019**, *6*, 60. [[CrossRef](#)]
23. Pressure Gauges. BS EN 837-1. 1998. Available online: <http://bqw.csstgc.com.cn/userfiles/2c9b92cd31ce403ca56b87348189cf0f/files/teckSolution/2019/09/BS%20EN%20837-1%20Pressure%20Gauges.pdf> (accessed on 25 June 2022).