applied sciences

MDPI

Article

A New Instrument Monitoring Method Based on Few-Shot Learning

Beini Zhang [1], Liping Li [2], Yetao Lyu [3], Shuguang Chen [1], Lin Xu [1] and Guanhua Chen [1,2,*]

[1] Department of Chemistry, The University of Hong Kong, Pokfulam, Hong Kong 999077, China; bennyzh@hku.hk (B.Z.)
[2] The Hong Kong Quantum AI Lab (HKQAI), New Territories, Hong Kong 999077, China
[3] Department of Electronic and Computer Engineering, Hong Kong University of Science and Technology, Kowloon, Hong Kong 999077, China
* Correspondence: ghc@yangtze.hku.hk

Abstract: As an important part of the industrialization process, fully automated instrument monitoring and identification are experiencing an increasingly wide range of applications in industrial production, autonomous driving, and medical experimentation. However, digital instruments usually have multi-digit features, meaning that the numeric information on the screen is usually a multi-digit number greater than 10. Therefore, the accuracy of recognition with traditional algorithms such as threshold segmentation and template matching is low, and thus instrument monitoring still relies heavily on human labor at present. However, manual monitoring is costly and not suitable for risky experimental environments such as those involving radiation and contamination. The development of deep neural networks has opened up new possibilities for fully automated instrument monitoring; however, neural networks generally require large training datasets, costly data collection, and annotation. To solve the above problems, this paper proposes a new instrument monitoring method based on few-shot learning (FLIMM). FLIMM improves the average accuracy (ACC) of the model to 99% with only 16 original images via effective data augmentation method. Meanwhile, due to the controllability of simulated image generation, FLIMM can automatically generate annotation information for simulated numbers, which greatly reduces the cost of data collection and annotation.

Keywords: instrument monitoring; few-shot learning

Citation: Zhang, B.; Li, L.; Lyu, Y.; Chen, S.; Xu, L.; Chen, G. A New Instrument Monitoring Method Based on Few-Shot Learning. *Appl. Sci.* **2023**, *13*, 5185. https://doi.org/10.3390/app13085185

Academic Editor:Byung-Gyu Kim

Received: 6 February 2023
Revised: 31 March 2023
Accepted: 11 April 2023
Published: 21 April 2023

1. Introduction

Automated instrument monitoring has become important in many fields such as automated new drug development, industrial production involving robotic arms, automatic driving, biochemical experiments, medical devices, and electric power facilities [1–4]. In high temperature, radiation, and pollution environments, the stability of equipment is extremely important, but the use of human monitoring is risky and costly [5–7]. In this paper, we mainly hope to achieve fully automated experiments for new drug and material development by using robotic arms with vision monitoring algorithms. In order to automatically monitor and operate instruments, identification is one of the most critical aspects. The existing instrument recognition techniques have two categories: traditional image-processing-based and machine-learning-based methods.

P. Xu, W. Zeng, Y. Shi, and Y. Zhang [8] proposed a method using color features that are transformed from Red, Green, and Blue (RGB) space into Hue, Saturation, and Value (HSV) to detect meter, while the accuracy achived in this paper is not high since the color information is easily influenced by the change of light. S. Shizhou, S. Kai, and L. Hui [9] proposed the corner matching algorithm, but it is easily affected by deformation and other factors, thus making it unreliable. The template matching method combined with threshold segmentation is the traditional algorithm that can achieve relative high accuracy in fully automatic numeric or alphabetic recognition, but it is limited by the

difficulty of threshold selection and is also prone to error detection, and omission [10]. Machine learning combined with morphology methods was also largely applied in the field of vision recognition [11–15]. L. Lei, H. Zhang, and X. Li [16] used Mask Regional Convolutional Neural Network (Mask-RCNN) to obtain the binary mask for each number and then applied a Back Propagation neural network algorithm to fuse invariant moment information. D. Li, J. Hou, and W. Gao [17] proposed image pre-processing and capsules network model for digitalization in Industrial Internet of Things. Y. Lin, Q. Zhong, and H. Sun [18] used histogram normalization transform to optimize the brightness and enhance the contrast of images, and then You Only Look Once version 3 (YOLOv3) was applied to detect and capture the panel area, and Convolutional Neural Networks (CNNs) were then used to read and predict the characteristic images.

In summary, traditional algorithms are vulnerable to environmental interference, such as uneven lightening and noise, so the accuracy is low [15,19]. When operating the instrumentation, it is important to set the correct parameters. In the experimental scenario presented in this paper, when the centrifuge is operated by the robot arm and the wrong number of revolutions is selected, there is a high risk of overload, downtime, or even machine scrapping. Therefore, the key steps of safety monitoring and operation identification are still largely dependent on manual work. In our experimental scenario, if there is no recognition algorithm with an accuracy rate of more than 98%, we will require a safety officer to monitor whether the parameters are set correctly. With the development of deep learning in the field of machine vision, fully automated instrumentation monitoring has become possible. However, traditional neural networks usually require large datasets for training [20–22]. Furthermore, the diverse data distribution patterns of various multi-digit dashboards lead to high data collection and labeling costs. To solve the above problems, this paper proposes a new Few-shot Learning Instrument Monitoring Method (FLIMM) based on a novel Fully Automated Digital Generation and Annotation Method (FADGAM) and Improved YOLOv5, which can obtain 99% average accuracy (ACC) with only 16 raw data.FADGAM can generate a large number of simulation images with similar distribution patterns to real data and automatically generate corresponding annotation files. Thus, our method effectively improves accuracy and greatly reduces the cost of data collection and annotation. When the application scenario is changed, FLIMM can achieve algorithm iteration with low data cost, making it possible for fully automated instrumentation monitoring to have a wider range of applications in the future.

2. Methods and Dataset

The overall flow chart of FLIMM is shown in Figure 1. FLIMM is composed of the following main steps: 1. Fuse the screen background image and number figures by the seamlessClone function which is built in OpenCV [23] and equal interval random position selection function to simulate different number permutation pattern. 2. Automatically calculate the label information and generate an annotation file from the randomly pasted image number size, type, center position, and original image size. 3. Pre-train the Improved YOLOv5 model with 1000 simulation images and automatically generate annotation files for corresponding images. 4. By randomly replacing one digit in 16 real screen digital data, 450 simulated data are generated, the pre-trained model is fine-tuned, and the final model is obtained.

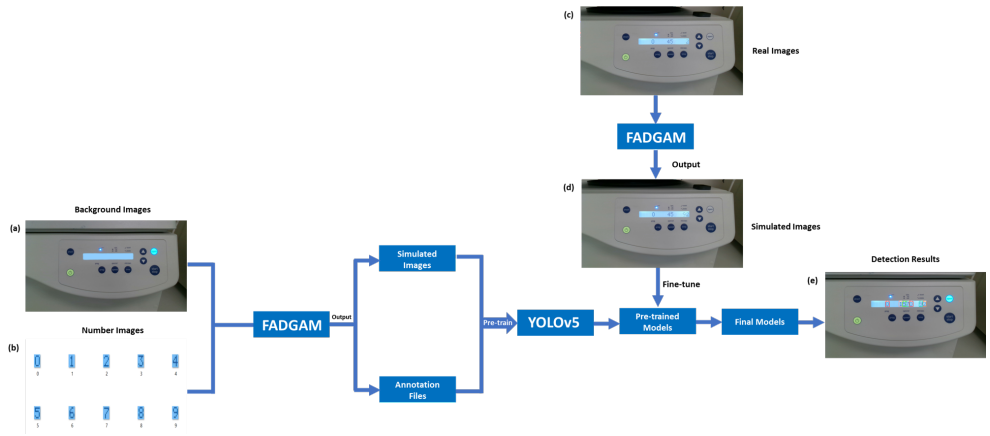

Figure 1. A flow chart of FLIMM. (**a**) Background Images. (**b**) Number Images. (**c**) Real images. (**d**) Simulated Images. (**e**) Detection results.

2.1. Dataset

In this paper, the dataset used for FLIMM consists of the training dataset, validation dataset, and test dataset. Among them, the training dataset contains 1015 images (1920 × 1080 pixels), the validation dataset contains 435 images (1920 × 1080 pixels), and the test dataset contains 390 images (1920 × 1080 pixels). All images in the training dataset and validation dataset are simulation data that are generated by the FADGAM function of FLIMM, while the test dataset consists of 390 real experimental images which are shown in Table 1. In detail, among the 1015 training images of FLIMM, 1000 images were generated by the multi-digit simulation mode of FADGAM based on one image with blank screen background (1920 × 1080 pixels, Figure 2a) and 10 digital figures (30 × 50 pixels, Figure 2b) for pre-training, and the other 450 were generated by single-digit mode of FADGAM based on 16 real images augmented by the simulation mode for fine-tuning (Figure 2c).

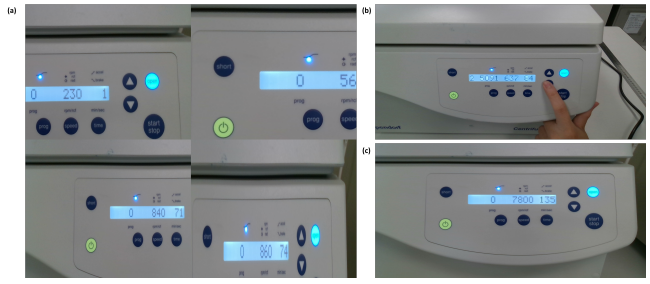

Figure 2. The augmentation results comparison (**a**) Mosaic data augmentation. (**b**) FADGAM with multi-digit augmentation mode. (**c**) FADGAM with single-digit augmentation mode.

Table 1. Dataset Summary.

Method	Train	Validation	Test
Mask R-CNN	280 (all real)	120 (all real)	390 (all real)
Pure YOLOv5	280 (all real)	120 (all real)	390 (all real)
FLIMM	1015 (all simulation)	435 (all simulation)	390 (all real)

The dataset used for Pure YOLOv5 and Mask R-CNN contains 280 real images for the training dataset, 120 real images for the validation dataset, and 390 real images for the test dataset. In addition, all comparison methods (human detection, template matching, Pure YOLOv5, Mask R-CNN and FLIMM) use the exact same test dataset, and the test dataset contains all the same real data images without any augmentation that are directly captured from the actual experiments.

2.2. Data Augmentation

Deep neural networks have higher accuracy and better sensitivity than traditional algorithms for complex backgrounds, noisy environments, and random targets. However, traditional neural networks usually require a large amount of training data, at least hundreds for a single class. Because the actual environment is often high temperature, high radiation, and high pollution, data collection for specific scenarios may be expensive and dangerous. In addition, instrument panels often consist of multiple digits that need to be manually labeled bit by bit, which also makes data annotation costly. In order to minimize data collection costs and improve data annotation efficiency, we propose a Fully Automated Digital Generation and Annotation Method. In the design of the data augmentation method, we refer to the self-attention mechanism, and consider that the background information is less useful for on-screen digital recognition, although it accounts for a relatively large part of the whole image, and the real useful information comes from the digital distinction on the screen. Therefore, in the simulation, we focused on the simulation of the arrangement and combination pattern of different numbers on the screen, so as to achieve higher accuracy with less raw data for the recognition in specific scenes.

In our simulation method we also have multi-digit and single-digit, where multi-digit means that multiple freely arranged digits are simulated by the FLIMM method and single-digit is simulated as a single digit. The multi-digit is mainly used with a blank screen background to simulate as many digit combinations as possible, while the single-digit is mainly used with a screen image with digits to rewrite specific digits to enhance the recognition of confusing digit pairs (e.g., 6 and 9). For the multi-digit augmentation mode of FADGAM, it calculates the starting coordinates of the first screen digits number (N1) as the initial input parameter. Since the screen remains horizontal, y_1 remains unchanged for all digits, and x_1 decreases with equal spacing, showing the characteristics of arithmetic sequence numerically. The x-coordinates of other digits are calculated as the Formula (1) shows:

$$x_n = x_1 - k * I_w \qquad k\left\{0, 1, ..., \frac{S_w}{I_w}\right\}, P_{k=i} = \frac{I_w}{S_w} \tag{1}$$

$$y_n = y_1 \tag{2}$$

in which the x_1 is the x-coordinate of N1, I_w is the interval value between adjacent digit centers, S_w is the width of the screen, and k is a set of the array with positive integers ≥ 0. Since k is the core parameter that controls the number of digits generated by FADGAM, in order to prevent the situation where two digits repeatedly select the same digit, in the actual algorithm, FADGAM uses the combination of random.sample() function and range() function of python by random.sample(range(n_s,n_e),n_t) to construct a sequence K with n_t number, and n_t is a positive integer. For the above function, range() is used to build a sequence of numbers spaced 1 apart starting with n_s and ending with n_e, random.sample() is used to pick n_t numbers from the sequence randomly. For example, our experiment set the $n_s = 0$, $n_e = 15$, and $n_t = 10$, which means randomly picking 10 numbers from the array [0,1,2,....,15] with non-repetitive ordering, such as [3,1,4,5,6,8,9,0,2,11]. Furthermore, the above-selected arrays are used as K values to calculate the x-coordinates (x_n) of the numbers 0–9 in turn, thus realizing the simulation of the random position arrangement of numbers. After calculating the center position of each generated figure, the FADGAM core performs edge region fusion through the seamlessClone function that is provided by OpenCV during image fusion. The seamlessClone function used in this paper requires five parameters: SRC, DST, MASK, POINT, FLAGS. FADGAM uses the number image as shown in Figure 3 as SRC, the background image in Figure 3 as DST, the MASK is a pure white image of the same size as DST, the POINT is the center of the fused position of number image coordinates (x_n, y_n), and FLAGS is the fusion method using cv2.MIXED_CLONE. For the single-digit augmentation mode of FADGAM, the rest of the calculation principle is

exactly the same as that of the multi-digit mode, only the k value is fixed to 0, and the N1 coordinate is fixed to the coordinate of the desired augmentation position.

YOLOv5 itself provides Mosaic data augmentation, as shown in Figure 2a, and will randomly intercept, stitch and scale any four training data. Figure 2b shows the multi-digit generation mode of FADGAM, whose screen digits are all randomly generated by FADGAM, and the images generated by this mode are used for pre-training. Figure 2c shows the single-digit generation mode of FADGAM, where only one digit of the screen is generated by simulation (third from right to left, 1), and the rest is real data, and the images generated by this mode are used for fine-tuning.

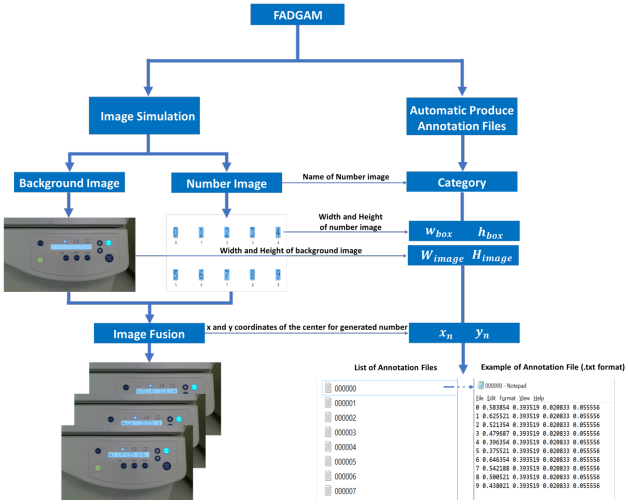

Figure 3. The flow chart of FADGAM.

2.3. Data Annotation Methods

In addition to generating instrument images with different permutation patterns for the simulation, FADGAM can automatically calculate and generate annotation files (.txt format) for each simulation image. Each annotation file is named the same as the simulation image name, where each row has the information of one annotation box, in the order by column are: category, x_c, y_c, w, and h. Since the x_n and y_n parameters of each generated number are pre-known parameters during the image simulation. Furthermore, the category is the class of the generated number, which should be equal to the image name, all parameters in the annotation file can be calculated automatically as Formulas (3)–(6):

$$x_c = x_n / W_{image} \tag{3}$$

$$y_c = y_n / H_{image} \tag{4}$$

$$w = w_{box} / W_{image} \tag{5}$$

$$h = h_{box} / H_{image} \tag{6}$$

in which, the W_{image} represents the width of the whole image, the H_{image} represents the height of the whole image, the w_{box} represents the width of the annotation box, the h_{box} represents the height of the annotation box, x_c represents the relative scale of the x-center of annotation box in the image, y_c represents the relative scale of the y-center of annotation box in the image, y_n represents the y-coordinate of the center of each annotation box, and x_n represents the x-coordinate of the center of each annotation box. The simulation images generated by FADGAM and their corresponding annotation files will be input together

as the pre-training files of Improved YOLOv5, which can greatly reduce the reliance on human labor since no manual annotation is required in this process.

A comparison of the annotation time between FADGAM and human labeling is shown in Figure 4. Since the FADGAM is designed to automatically generate labels based on simulated images, the FADGAM takes an average of only 0.03 s per image to label. For the human labeling time or human detection time, we invited four human experts to conduct the test and calculated an average labeling time of 22.925 s per image. Compared with pure human labeling, FADGAM effectively improves labeling efficiency by more than 99.87%, which is calculated by $(Time_{Human} - Time_{FADGAM})/Time_{Human}$ [24].

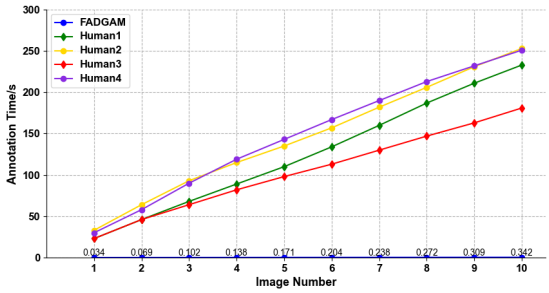

Figure 4. A comparison of the annotation time.

2.4. Improved YOLOv5 Structure

Improved YOLOv5 is a model based on YOLOv5 proposed in this paper, which can be structurally divided into three parts: input, backbone, and head, as shown in Figure 5. The input side mainly performs three operations: mosaic data augmentation, adaptive anchor frame calculation, and resize. The principle of mosaic data enhancement is to randomly select four images from the training dataset, perform random cropping, scaling, and rotation operations on each image, and then stitch them together into a single image. Mosaic can enrich the amount of information while increasing the number of detection targets in a single input data, thus improving the accuracy of target detection. The resize operation is designed for the original images with different widths and heights. Resize adopted a uniform scaling process to make these images with a standard size, for the part that needs to be filled after scaling is uniformly filled with gray pixel values (114,114,114). The operation of adaptive anchor frame calculation is based on the initial anchor frame by comparing the input prediction frame with the real frame, iteratively calculating the gap, and then reverse update so as to finally calculate the most suitable anchor frame value.

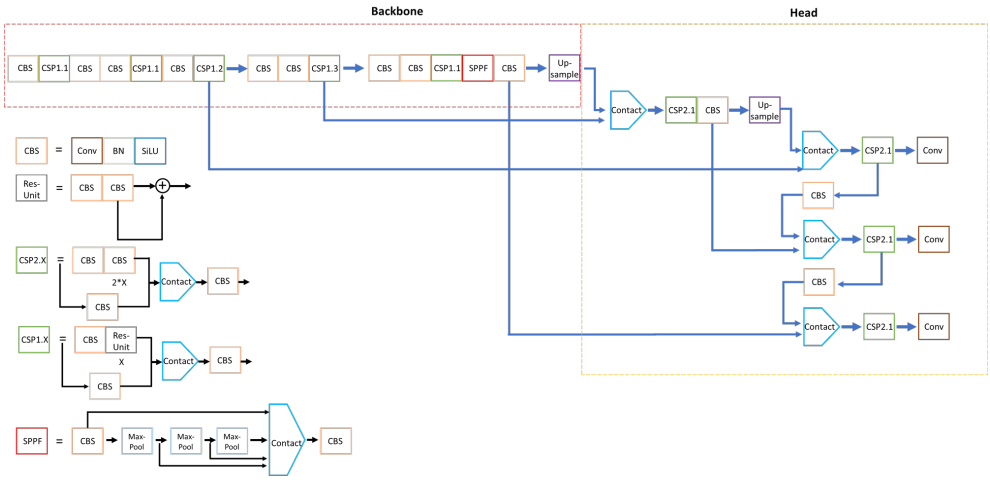

Figure 5. The structure of Improved YOLOv5.

The backbone of Improved YOLOv5 is mainly composed of CBS and CSP blocks, where the structure of CBS is convolutional layer + batch normalization + silu activation

function. In addition, the structure of CSP in a backbone has a shortcut, which is different from the CSP in the head, so it is distinguished by 1.x and 2.x, respectively. With the addition of the shortcut mechanism, in the back-propagation, not only the gradient but also the gradient before the derivation is passed between every two blocks, which is equivalent to artificially enhancing the size of the gradient passed forward structurally, and is thus able to reduce the possibility of gradient dispersion. The role of the SSPF module is to stitch the results of CBS and three pooling together and then extract features by CBS. Although the feature map is pooled three times, the feature map size does not change, and the main role of SPPF is to extract and fuse the high-level features. The head part of Improved YOLOV5 gives three different scales of feature maps for prediction by FPN and PAN structure, which is basically consistent with YOLOv4.

2.5. Training

FLIMM adopts a pre-training plus fine-tuning mechanism in training, and pre-training is performed by a large amount of simulated data, which has high randomness in the distribution position and permutation of numbers but has a certain gap with the real data in terms of brightness and appearance pattern of numbers. Therefore, we later adopted an active learning mechanism to select 16 real data and perform secondary amplification on them using FADGAM. In the secondary augmentation process, we no longer arrange and combine the numbers for the full display screen but change the numbers at a specific location, thus augmenting the 16 original data into simulated images with a more similar distribution pattern to the real data for model fine-tuning and thus obtaining a higher accuracy. Data annotation software is labeled, and the model training and testing equipment for all methods are DGX A100 with 2 TB memory.

2.6. Comparison Method

This paper uses human detection results as the gold standard. As for the traditional panel recognition method, we choose the most widely used method, template matching that is combined with the automatic threshold segmentation, as the comparison scheme. The principle of the template matching method is to compare the recognition target with the template image in turn, and by calculating the error between the template and the target image, the template image with the smallest error will be selected as the final prediction result. In the actual experiments of digital recognition, in order to reduce the interference of the background area to the digital target, the template match first manually selects the screen region of the number to be measured. Threshold segmentation is then performed for the selected area, which can decrease the influence of the background.

In addition, we used 280 real datasets (1920 $\times$ 1080 pixels) to train Pure YOLOv5 and Mask R-CNN. The benefit of mosaic data augmentation of Pure YOLOv5 is the ability to augment the richness of the background images, but the realism of the digital target permutations of mosaic data augmentation are much lower compared to the augmentation method of FADGAM in the equipment monitoring scenario.

3. Results and Discussion

For the digital instrument monitoring, the measurement results of different methods are shown in Table 2, where ACC represents the average accuracy. The calculation method of ACC is $ACC = (TN + TP)/(TN + TP + FN + FP)$, in which the TP represents True Positive, FP represents False Positive, FN represents False Negative, TN represents the True Negative. For our task, the TN is not suitable so is set to be 0 for all methods. Furthermore, the Human measurements are selected as the gold standard in the form of ignoring random errors and are considered to be 100%, and all other results will be measured against human results. Although the manual measurement is the most accurate method, it also has the slowest detection speed. For a single image with eight digits, the average time to detect and record each digital value is 22.925 s. Template Matching is limited by the difficulty of automatic threshold selection during threshold segmentation step and is susceptible to

factors such as noise, light, and uneven screen color, resulting in a large difference between the segmentation result and the template. The above reasons lead to a large number of erroneous and missing results of template matching, with the lowest accuracy of 49.2% and a relatively slow detection speed.

Table 2. Comparison of the results of different methods for the test dataset.

Method	TP	FP	FN	ACC	Detection Time/s
Human	2581	0	0	100%	22.925
Template Matching	1672	815	909	49.2%	15.971
Pure YOLOv5	2170	0	411	84.1%	0.107
Mask R-CNN	2361	53	220	89.6%	4.131
FLIMM	2560	0	21	99.2%	0.092

Compared with the traditional methods, the deep learning algorithms have improved the recognition results significantly, and the accuracies of the Pure YOLOv5 and Mask R-CNN models that are trained with 280 real data are 84.1% and 89.6%, respectively.Both methods are among the most widely used deep learning algorithms for target detection in industry and academia. In general, Pure YOLOv5 has faster detection speed but more False Negative targets, and Mask R-CNN has higher average accuracy but more false positive targets. Pure YOLOv5 adopts the mosaic augmentation method, which can effectively increase the number of targets in the training image but cannot accurately simulate random permutations of numbers, so the overall accuracy is lower than that of FLIMM. A typical example of detection results is shown in Figure 6. Template matching has one False Positive result and two False Negative results, Mask R-CNN has one False Negative result and one False Positive result, Pure YOLOv5 has two False Negative results, and all results of FLIMM are correct. So, the accuracy of Mask R-CNN and Pure YOLOv5 is higher than template matching but lower than FLIMM.

Figure 6. An example of the detection results. (**a**) Original Image. (**b**) Template Matching result. (**c**) Mask R-CNN result. (**d**) Pure YOLOv5 result. (**e**) FLIMM result. The detection results of (**b**) are, from left to right: 0,1,1,0,0,0 1, where the last two digits are missed and a 0 is mistakenly checked for 1 at the same time. The results of mask detection in (**c**) are, from left to right: 0,1,0,0,0,2,2 and 6, where the second bit is missed and the last bit is incorrectly detected by an additional 2. Each test box in (**d**) has the test number on the left and the confidence level on the right, from left to right: 0 0.93, 1 0.94, 1 0.94, 0 0.93, 0 0.92, 0 0.92, the last two digits are missed and the other tests are correct. Figure (**e**) shows the number of tests on the left of each box and the confidence level on the right, from left to right: 0 0.96, 1 0.94, 1 0.94, 0 0.93, 0 0.93, 0 0.93, 2 0.92,6 0.94, the last two digits were missed and the other tests were correct.

FLIMM is pre-trained with a large number of simulation-generated images (Figure 2b), which largely produces the possible alignments in the real situation. The model was then fine-tuned with single-digit simulated images (Figure 2c) generated from 16 real images, effectively achieving an average accuracy of more than 99%. Figure 7 shows the confusion matrix of FLIMM, and it can be clearly found that the detection rate of all kinds of samples is maintained at a high level of over 96%, which proves the effectiveness of the FLIMM

method. At the same time, compared with Pure YOLOv5 and other algorithms that require pure manual annotation, FLIMM can automatically generate annotation files for multiple full simulation data while improving the effectiveness, and for single digit simulation data, only the first source image needs to be annotated manually, and the rest of the augmented images can be automatically added with the annotation information of the simulation target based on the annotation of the source image, which greatly saves the time and cost of manual annotation. The above-mentioned realism, efficiency, fully automated annotation and data collection savings of FLIMM allow for fast migration, as it does not require a lot of time to collect and annotate data when encountering new scenarios. Figure 8 shows an example of a new scene migration, where the detection target is anoven display figures. Using FLIMM, we were able to efficiently amplify only 11 raw data, obtain 1400 training sets, 600 validation sets and, according to the detection results on a test set of 200 real data, an accuracy of over 99%. To generate these 2000 simulated images, FLIMM took only about 120 s, compared to more than 20 h to manually collect and label the 2000 images. This experiment effectively demonstrates the effective migration of the FLIMM method between scenes.

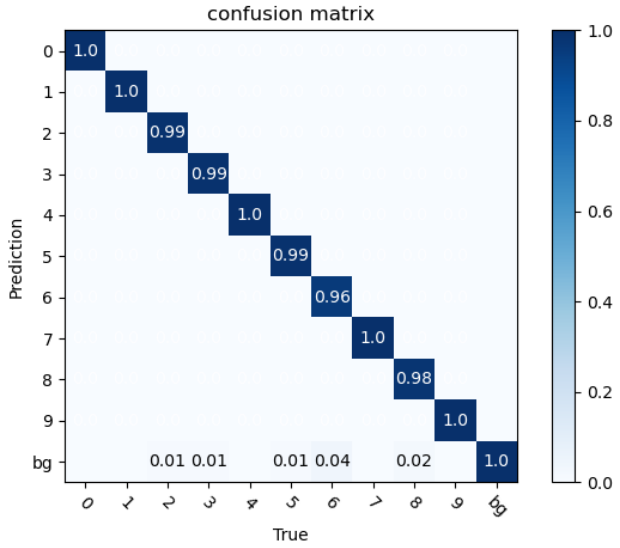

Figure 7. The confusion matrix of FLIMM.

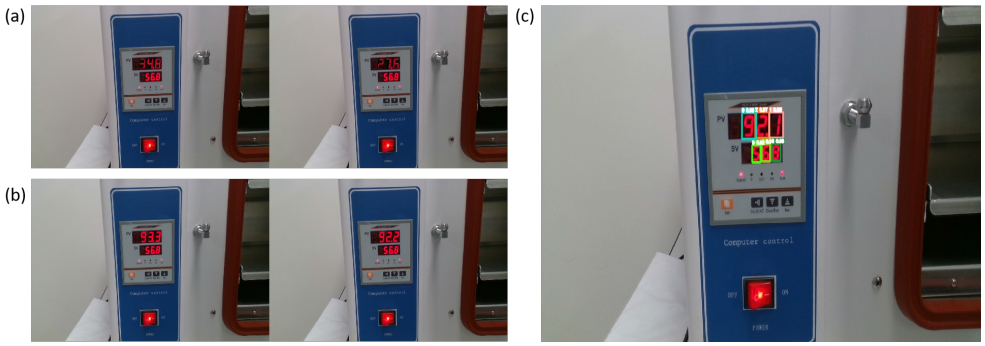

Figure 8. Detection example on migrating new scenarios. (**a**) FLIMM augmentation results. (**b**) Real images examples. (**c**) Improved YOLOv5 detection examples.

In summary, the deep-learning-based approach has greatly improved the error detection rate compared with the traditional algorithm, while FLIMM further reduces the leakage rate based on Improved YOLOv5 by simulating as diverse real data distribution as possible through data augmentation, thus improving the accuracy by 14%. From the perspective of detection time, both traditional methods and deep learning algorithms

have improved the detection rate compared to manual detection, and since Pure YOLOv5 and FLIMM are based on a lightweight architecture with a small number of parameters, YOLOv5s, the detection speed is greatly improved, taking only about 0.1 s per image. Therefore, FLIMM shows the best performance in terms of high-detection accuracy, ease of labeling, and quick detection time. In addition to dashboard monitoring and automated reading, we believe FLIMM has greater potential for exploration in other similar areas, such as letter recognition and license plate recognition.

4. Conclusions

In this paper, we propose the FLIMM, which can automatically perform dashboard image simulation, annotation file production, and image detection with a small amount of raw data. FLIMM augments blank screen images via the multi-digit mode of the FADGAM to automatically generate 1000 simulated images with different distribution patterns and annotation files for model pre-training. A total of 16 real data are augmented by the single-digit mode of FADGAM to automatically generate 450 simulated images for fine-tuning. In summary, FADGAM augments a large number of simulated images with high fidelity using less than 20 raw data, which allows the model to obtain an accuracy of more than 99%. In addition, due to the controllability of the process of generating simulation figures via FLIMM, we have developed a fully automatic labeling function for it, which greatly reduces the workload of traditional deep neural networks that require pure manual labeling for training data and improves labeling efficiency. Compared with template matching, one of the most commonly used traditional fully automated instrument identification methods, FLIMM greatly reduces the error and miss detection rates and improves the accuracy by 50.1%. Compared with pure YOLOv5, which simply uses mosica augmentation to obtain 84.1% accuracy, FLIMM added the FADGAM augmentation method effectively improving the accuracy by 14%. FADGAM reduced the original training dataset from 280 to 16 images, significantly reducing the data annotation while greatly reducing the training data collection cost by more than 99%. We believe that FLIMM's design concept can be more widely used in similar fields such as the Completely Automated Public Turing test to tell Computers and Humans Apart (CAPTCHA) recognition, credit card number recognition, and license plate recognition in the future.

Author Contributions: In this article, G.C. was responsible for the overall control and monitoring of the experiment; B.Z. proposed the idea of the experiment, designed the experimental method and the overall experiment; L.L. was responsible for data labeling and comparing the test results of the method; Y.L. was responsible for providing advice on the improvement and calibration of the algorithm; L.X. and S.C. were responsible for proofreading and providing comments on the article. 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.

Data Availability Statement: Data of this paper can be obtained from the corresponding author with a reasonable request.

Acknowledgments: This work was founded by the AIR@InnoHK Centre of Hong Kong Quantum AI Lab of Hong Kong Government. Special thanks to Jiang Wu, Haoyu Zhu, Yuan Zhuang, Xuejian Leng for their help in data annotation, collection, and comparison.

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

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Average accuracy
CNN	Convolutional neural networks
FLIMM	Few-shot Learning Instrument monitoring method
FADGAM	Few-shot learning digital generation and annotation method
Mask R-CNN	Mask Regional Convolutional neural networks
RGB	Red, green, and blue
YOLOv3	You only look once version 3
YOLOv5	You only look once version 5

References

1. Manabu, O.; Fujioka, T.; Hashimoto, N.; Shimizu, H. The application of RTK-GPS and steer-by-wire technology to the automatic driving of vehicles and an evaluation of driver behavior. *IATSS Res.* **2006**, *30*, 29–38.
2. Khan, A.K. Monitoring power for the future. *Power Eng. J.* **2001**, *15*, 81–85. [CrossRef]
3. Alfa, M.J. Medical instrument reprocessing: Current issues with cleaning and cleaning monitoring. *Am. J. Infect. Control* **2019**, *47*, A10–A16. [CrossRef] [PubMed]
4. Prabhu, G.R.D.; Yang, T.H.; Hsu, C.Y.; Shih, C.P.; Chang, C.M.; Liao, P.H.; Ni, H.T.; Urban, P.L. Facilitating chemical and biochemical experiments with electronic microcontrollers and single-board computers. *Nat. Protoc.* **2020**, *15*, 925–990. [CrossRef] [PubMed]
5. Lollino, G.; Manconi, A.; Giordan, D.; Allasia, P.; Baldo, M. Infrastructure in geohazard contexts: The importance of automatic and near-real-time monitoring. In *Environmental Security of the European Cross-Border Energy Supply Infrastructure*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 73–89.
6. Angeli, M.G.; Pasuto, A.; Silvano, S. A critical review of landslide monitoring experiences. *Eng. Geol.* **2000**, *55*, 133–147. [CrossRef]
7. Arora, M.; Jain, A.; Rustagi, S.; Yadav, T. Automatic number plate recognition system using optical character recognition. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.* **2019**, *5*, 986–992. [CrossRef]
8. Xu, P.; Zeng, W.; Shi, Y.; Zhang, Y. A Reading Recognition Algorithm of Pointer Type Oil—level Meter. *Comput. Technol. Dev.* **2018**, *28*, 189–193.
9. Shizhou, S.; Kai, S.; Hui, L. A robust method for automatic identification of electricity pointer meter. *Comput. Technol. Dev.* **2018**, *28*, 192–195.
10. Puranic, A.; Deepak, K.; Umadevi, V. Vehicle number plate recognition system: A literature review and implementation using template matching. *Int. J. Comput. Appl.* **2016**, *134*, 12–16. [CrossRef]
11. Alegria, E.C.; Serra, A.C. Automatic calibration of analog and digital measuring instruments using computer vision. *IEEE Trans. Instrum. Meas.* **2000**, *49*, 94–99. [CrossRef]
12. Huang, Z.; Wang, C. Reading Recognition of Digital Display Instrument Based on BP Neutral Network. In Proceedings of the 2008 International Conference on Computer Science and Software Engineering, Washington, DC, USA, 12–14 December 2008; IEEE: Piscataway, NJ, USA, 2008; Volume 1, pp. 106–109.
13. Gong, R.; Nian, S.; Chen, L.; Zhang, G.; Tian, Y. Real-time reading recognition of digital display instrument based on BP neural network. In Proceedings of the IEEE ICCA 2010, Xiamen, China, 9–11 June 2010; IEEE: Piscataway, NJ, USA, 2010; pp. 1233–1238.
14. Mariappan, M.; Ramu, V.; Ganesan, T.; Khoo, B.; Vellian, K. Virtual Medical Instrument for OTOROB based on LabView for acquiring multiple medical instrument LCD reading using optical charcater recognition. In Proceedings of the International Conference on Biomedical Engineering and Technology (IPCBEE), Kuala Lumpur, Malaysia, 4–5 June 2011; Volume 11, pp. 70–74.
15. Wang, Y.; Xu, C.B.; Wang, J.; Gao, P.; Gao, J.P.; Xin, M.Y.; Zheng, J.L.; Shi, X.L. Automatic Recognition of Indoor Digital Instrument Reading for Inspection Robot of Power Substation. In Proceedings of the 2017 International Conference on Wireless Communications, Networking and Applications, Shenzhen, China, 20–22 October 2017; pp. 251–255.
16. Lei, L.; Zhang, H.; Li, X. Research and Application of Instrument Reading Recognition Algorithm Based on Deep Learning. In Proceedings of the 2021 3rd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM), Manchester, UK, 23–25 October 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 535–539.
17. Li, D.; Hou, J.; Gao, W. Instrument reading recognition by deep learning of capsules network model for digitalization in Industrial Internet of Things. *Eng. Rep.* **2022**, *4*, e12547. [CrossRef]
18. Lin, Y.; Zhong, Q.; Sun, H. A pointer type instrument intelligent reading system design based on convolutional neural networks. *Front. Phys.* **2020**, *8*, 618917. [CrossRef]
19. Lei, B.; Wang, N.; Xu, P.; Song, G. New crack detection method for bridge inspection using UAV incorporating image processing. *J. Aerosp. Eng.* **2018**, *31*, 04018058. [CrossRef]
20. Kavzoglu, T. Increasing the accuracy of neural network classification using refined training data. *Environ. Model. Softw.* **2009**, *24*, 850–858. [CrossRef]
21. Beini, Z.; Xuee, C.; Bo, L.; Weijia, W. A new few-shot learning method of digital PCR image detection. *IEEE Access* **2021**, *9*, 74446–74453. [CrossRef]

22. Li, S.; Zhao, X. Image-based concrete crack detection using convolutional neural network and exhaustive search technique. *Adv. Civ. Eng.* **2019**, *2019*, 6520620. [CrossRef]
23. Bradski, G.; Kaehler, A. OpenCV. Dr. Dobb's Journal of Software. 2000. Available online: http://roswiki.autolabor.com.cn/attachments/Events(2f)ICRA2010Tutorial/ICRA_2010_OpenCV_Tutorial.pdf (accessed on 10 April 2023).
24. Zhang, B.; Luo, X.; Lyu, Y.; Wu, X.; Wen, W. A defect detection method for topological phononic materials based on few-shot learning. *New J. Phys.* **2022**, *24*, 083012. [CrossRef]