Journal of *Marine Science and Engineering*

MDPI

Article

A Study on Enhancement of Fish Recognition Using Cumulative Mean of YOLO Network in Underwater Video Images

Jin-Hyun Park [1] and Changgu Kang [2,*]

[1] Department of Mechatronics Engineering, Gyeongnam National University of Science and Technology, Jinju-si 52795, 33 Dongjin-ro, Gyeongsangnam-do, Korea; uabut@gntech.ac.kr

[2] School of Computer Engineering, Gyeongnam National University of Science and Technology, Jinju-si 52795, 33 Dongjin-ro, Gyeongsangnam-do, Korea

* Correspondence: ckang@gntech.ac.kr; Tel.: +82-55-751-3321

Received: 15 October 2020; Accepted: 19 November 2020; Published: 22 November 2020

Abstract: In the underwater environment, in order to preserve rare and endangered objects or to eliminate the exotic invasive species that can destroy the ecosystems, it is essential to classify objects and estimate their number. It is very difficult to classify objects and estimate their number. While YOLO shows excellent performance in object recognition, it recognizes objects by processing the images of each frame independently of each other. By accumulating the object classification results from the past frames to the current frame, we propose a method to accurately classify objects, and count their number in sequential video images. This has a high classification probability of 93.94% and 97.06% in the test videos of Bluegill and Largemouth bass, respectively. The proposed method shows very good classification performance in video images taken of the underwater environment.

Keywords: exotic invasive species; object classification; video image; YOLO

1. Introduction

Techniques for classifying and estimating populations in aquatic ecosystems are important and essential for conserving rare and endangered populations and eliminating exotic species that destroy ecosystems. In general, for small individuals, the number is estimated either by direct counting or by using the cross-line method [1,2] or the mark collection method [3,4]. In the case of large numbers of them, we generally use a camera, and must make efforts to count individuals directly from camera images or video images [5,6]. It is very difficult to classify populations and estimate the number of individuals in any method.

Convolutional Neural Network (CNN) is widely used for object recognition and classification, and shows very good results. Many methods have been proposed based on the principles of CNN, and their performance has been demonstrated in various fields [7–10]. However, some studies in the field have been conducted using CNN [11–14]. The issue of image classification began in AlexNet [8] and further research has been carried out in GoogLeNet and VGGNet [15,16]. ResNet, which appeared in 2015, outperformed human judgment [17]. Based on these studies, research has focused not only on the image classification problem, but also on the image detection problem that classifies various objects of the image into specific classes, and predicts the location of the specific objects [10]. R-CNN [18,19], which shows good performance in image detection problems, creates potential bounding boxes on an image, and then runs a classifier on the proposed bounding boxes. After the classification, post-processing is used to refine the bounding boxes, eliminate duplicate detection, and calculate classification scores based on other objects [20]. In contrast, You Only Look Once (YOLO) [20–22] is the fastest system to

detect and classify various objects. YOLO is a simple structure with a single convolutional network that simultaneously predicts bounding boxes and classification probabilities. The much faster operating time allows real-time processing, and plays a role in filtering the background image by reasoning globally. Furthermore, a general CNN cannot classify multiple objects in one image, but the YOLO network can classify multiple objects using a bounding box. This is useful for recognizing different objects in a single image and counting the number of those that are recognized. In particular, it can be used very effectively for classifying populations and estimating the number of individuals in video images.

However, despite the advantages of YOLO, it is difficult to obtain accurate results in every frame for low illumination or unfocused images, such as video images in an underwater environment [11,23]. To solve this problem, a data collection method has been proposed and has become a very useful alternative [11]. This method can improve the classification performance of images, but it is difficult to classify multiple objects in a single image or perform real-time processing for counting the number of objects. The human visual system can classify objects by constantly looking at them. In contrast, YOLO does not use sequential image but it processes images in each frame individually. YOLO can process video images in real-time, but independently classifies object's locations and classes using only one frame at a time. This means that the classification results of the previous frame image do not affect those of the current frame image.

Therefore, we propose the method to accurately classify objects and count the number of objects in a video image by accumulating the classified results from the past frame to the current frame. The proposed method may degrade the classification performance of some frames depending on the underwater environment, but this disadvantage is compensated by applying the cumulative average. This is a heuristic method that mimics human experience and learning.

In this study, we use YOLOv2 [21] for object recognition in video images taken in the underwater environment, and we apply the human heuristic approach by accumulating the mean of classification results of past frames to increase object classification results and count the number of objects accurately. We verified that the proposed method improves the classification and counting of objects in video images.

2. YOLO and Learning Data

2.1. YOLO

There are many studies on how to apply CNN to classify objects in unedited real-time video images [22,24–27]. In order to apply CNN, it is necessary to crop an image to fit the input size of CNN [24–26]. Recent studies have used a saliency map [26,27] to select the region to crop an image. However, when using a saliency map, processing time and performance vary depending on the number of filters. This is the most important factor in real-time processing. In the case of YOLO, there is no need to crop the input image for object recognition. Additionally, it has a structure and processing time that are suitable for real-time processing. YOLO handles bounding boxes and class probabilities at more than 45 fps over the entire image, making it very fast. Furthermore, if there are no objects or they are not subject to classification in the image, it is less likely to detect the wrong object [20–22]. However, from the viewpoint of real-time processing, it is difficult to derive the accurate result in every frame from the video images with unfocused or low illumination. If YOLO outputs accurate classification results in every frame, the proposed method may not be necessary.

2.2. Learning Data and Video Image

For the learning data and the performance evaluation of this study, we needed image data for learning and video images taken in the underwater environment. It was also difficult to construct a lot of image data for the same kind of object for learning, and to obtain video images taken in the underwater environment. Although this study can be applied to various kinds of object classification, we selected fish that are easy to shoot in the underwater environment. Therefore, in this study, video images of fish

were taken directly in the laboratory environment, and learning data images of fish species were made based on the captured video images [28]. The fish species to be classified include Largemouth bass, Bluegill, Common carp, Crucian carp, Catfish, Mandarin fish, and Skin carp. The seven fish images for learning have a similar streamline shape, and the image environment of the objects to be classified shows very different characteristics depending on the underwater environment [28]. We used 5000 [image/fish species] as basic learning data for YOLO. Figure 1 shows the labeled fish in fish images.

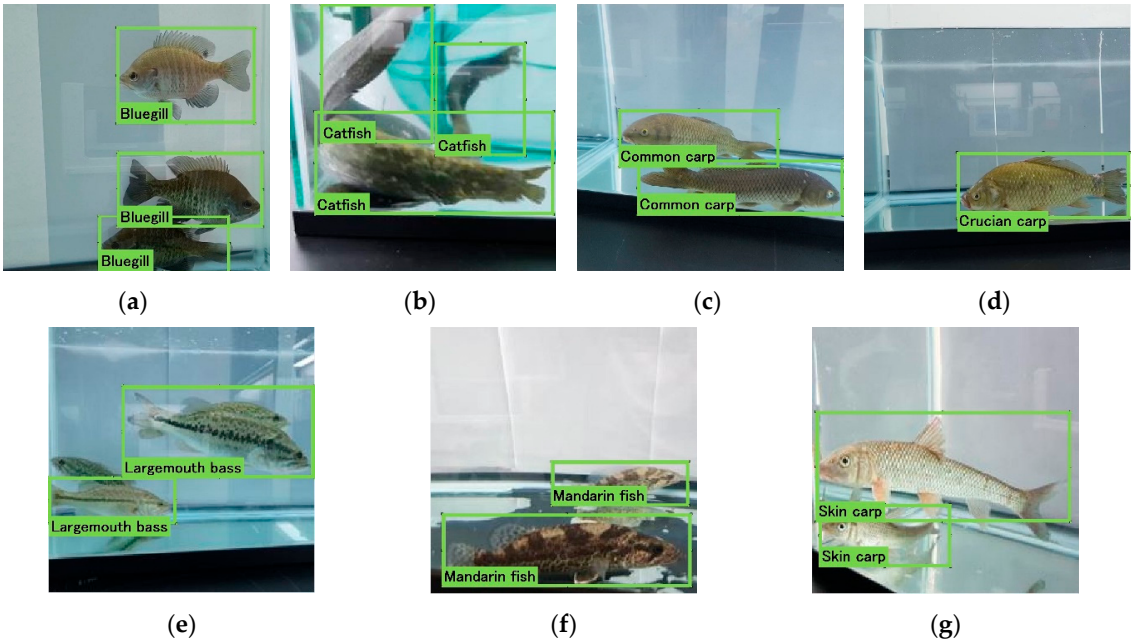


Figure 1. The labeled fish—(**a**) Bluegill, (**b**) Catfish, (**c**) Common carp, (**d**) Crucian carp, (**e**) Largemouth bass, (**f**) Mandarin fish and (**g**) Skin carp.

Using anchor boxes to help predict the position and the size of objects in an image in deep learning systems increases the speed and efficiency of object detection. Even in YOLO networks, anchor boxes are a set of bounding boxes with defined heights and widths. These boxes are defined to capture the magnification and aspect ratio of the specific object classes that are to be detected, and are typically selected according to the size of the objects of the learning data set, as shown in Figure 1. In this study, the average Intersection of Union (IoU) was calculated by k—mean clustering of various bounding box sizes, and k = 4 with an average IoU of more than 0.74 was selected.

YOLO was learned using YOLOv2 provided by MATLAB. The optimization method for YOLOv2 used Stochastic Gradient Descent with Momentum (SGDM), the initial learning rate was set to $1.0 \times e^{-4}$, and the size of the mini-batch was set to 256. For the hardware devices, CPU (Intel i9-7900 3.30 GHz) and four GPUs (NVIDIA GeForce GTX1080Ti) were used. Figure 2 shows the learning results of YOLOv2. In the case of catfish, the average precision is 82%. Six species of fish, except catfish, were learned with more than 93% precision.

After learning YOLO, a heuristic method was applied to classify objects from video images taken in the underwater environment. Figure 3 shows the installed underwater photography system for test video images. Since there are many floats in the aquatic environment, the video image changes according to the change of sun and external light. Our underwater photography system was equipped with wireless communication, and transmitted classified fish images and classification probabilities. Therefore, we needed a method that can accurately classify fish and count the number of classified fish in video images.

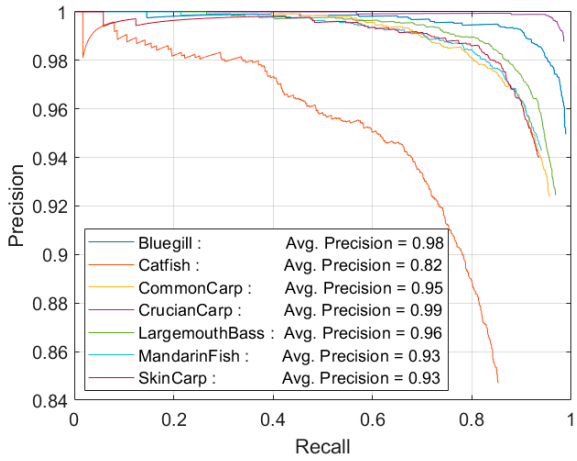


Figure 2. Learning results.

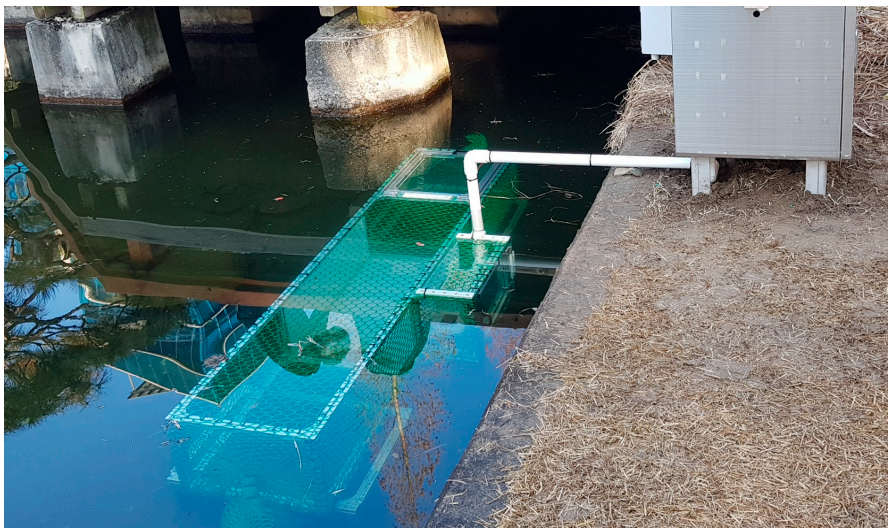


Figure 3. Underwater photography system.

3. Proposed Method

3.1. Heuristic Method

Humans can classify objects by looking only once at some objects or objects in the video image, but in general, humans recognize objects in succession, classify objects, and accurately classify objects with sequential images using information from their experience and learning. For example, if an animal suddenly appears in a dark forest, few people identify it precisely from the beginning. At first, they are not sure if it is a particular animal. However, if the animal appears in close proximity, most people will recognize it as a particular animal. In this way, the proposed method sequentially applies real-time images to YOLO. It computes the classification probability of an object in the heuristic method of computing the cumulative mean by accumulating the outputs of YOLO. This guarantees higher classification performance by classifying objects using the cumulative mean of sequential object classification values, than by their using CNN or YOLO from a single image containing them.

In YOLO, the classification result for each object is represented by probability values. Assuming these classification results follow a normal distribution, as the number of samples for the same object increases by the central limit theorem, it is known that the mean of the classified sample means is equal

to the mean of the population, and the standard error of sample means decreases with the number of samples, as shown in Equation (1) [29,30]:

$$s_x = \frac{\sigma}{\sqrt{n}} \tag{1}$$

where, s_x is the standard error of the sample means, σ is the standard deviation of the population, and n is the number of samples.

Therefore, the more classified samples of the same object in the video images, the higher the confidence level for the classified object. In general, the time for recognizing and disappearing objects in video images is not constant, but assuming that at least 1 s or more is measured, a sample means of 30 frames or more may be detected. In the case of using the sample means of 30 frames or more, the standard error is reduced to $s_x < \left(\sigma / \sqrt{30} = 0.1825\sigma\right)$.

By applying the heuristic method to YOLO, our method can maintain higher accuracy than CNN or YOLO classification results, which use one frame for the object classification of video images. This is because our method has low standard error depending on the number of frames, as shown in Equation (1). The mean of the sample means was calculated as the cumulative mean for each object using the classification results for successive images. The mean of the sample means was calculated as the cumulative mean for each object using the classification results for successive images, as shown in Equation (2).

$$Avg_i(k) \;=\; \frac{(i-1)}{i}\, Avg_{i-1}(k) \;+\; \frac{p_i(k)}{i}, \tag{2}$$

where, i denotes the number of frames, and k denotes a classification object, so $Avg_i(k)$ is the cumulative mean for i and k, and $p_i(k)$ is the probability of classification for i and k.

3.2. Cumulative Mean of The YOLO Network

We describe how to enhance recognition using the cumulative mean of the YOLO network. The proposed method uses YOLO for object recognition, and uses the heuristic approach to improve object classification results. Figure 4 shows the overall flow for the proposed method. Our method uses YOLO to recognize fish from all frame images of the video. Furthermore, it calculates the cumulative mean using the heuristic method when the fish were recognized. Next, the number of fish is counted. The method for counting the number of fish is to increase the number of fish each time the fish disappears after the fish is recognized for a certain period of time within the capture area of the image. If fish have not been recognized in the capture region of the image for a certain period of time, they are not classified, as they are considered less reliable.

Figure 5 shows the capture region, and the capture lines are adaptively set according to the size of the detected object, as shown in Equation (3). If the object size is large, the bounding box of the recognized object is large, and the recognition probability is also high. Therefore, the capture region is set to narrow, so that the object can be classified when the center of the object is only a little away from the center of the image. If the object is small, the capture region is set to wide, so that the object can be classified when the center of the object is far from the center of the image. When YOLO did not recognize fish over 20 frames after the fish was recognized in the capture region, it is assumed that the fish disappeared in the other direction, and we classified the fish and calculated the number of fish:

$$c_l \;=\; A/(w_l \times w_h),\; c_s \;\leqq c_l \;\leqq c_w \tag{3}$$

where, A is any constant, w_l is the width of the bounding box of the object, w_h is the height of the bounding box, c_l is the width of the capture region, c_s is the width of the minimum capture region, and c_w is the width of the maximum capture region.

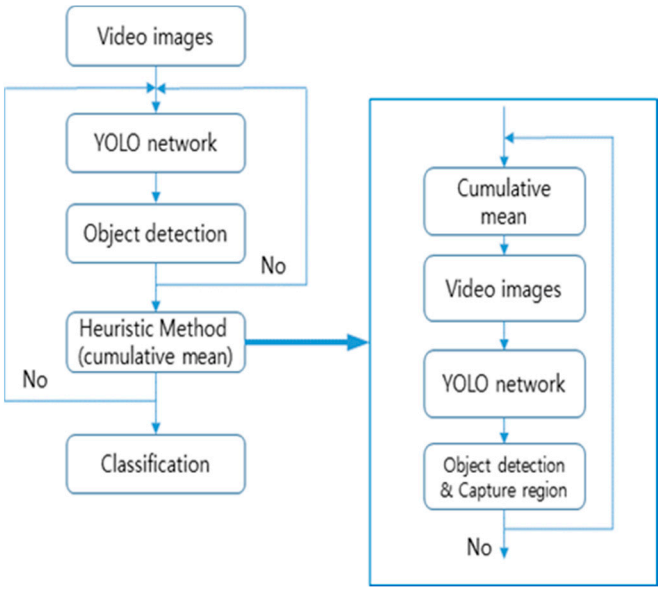


Figure 4. The flow for cumulative mean of the YOLO network.

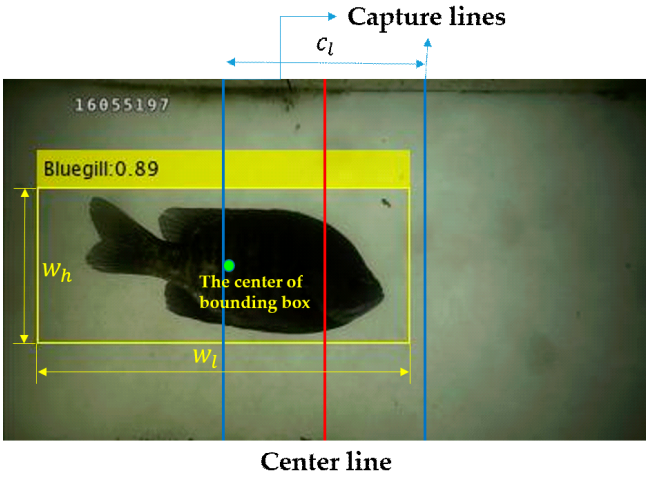


Figure 5. Capture lines.

4. Experiments

The Largemouth bass and the Bluegill video images taken in the pond were used for the performance evaluation. In general CNN, classification of objects is conducted by one frame, so the classification performance and recognition rate differ, depending on learning. In particular, as the video images are underwater, it is sensitive to changes in sunlight or external lighting, and thus a secondary method of recognizing fish in a single frame is required [20–22]. In addition, if an object other than the object to be classified in the video image is captured and input to CNN, CNN has the disadvantage of forcibly classifying it as a fish species. YOLO also classifies objects for a single image, which can degrade classification performance depending on the learning; and it is difficult to obtain accurate classification results in every frame.

Firstly, an evaluation of the proposed method was performed on Largemouth bass. In the video images of 34 Largemouth bass, the proposed method classified 33 Largemouth bass (97.06%), with one classified as an object. Figure 6 shows the classification probabilities of 33 Largemouth bass, and recognized with a value of 60% or greater.

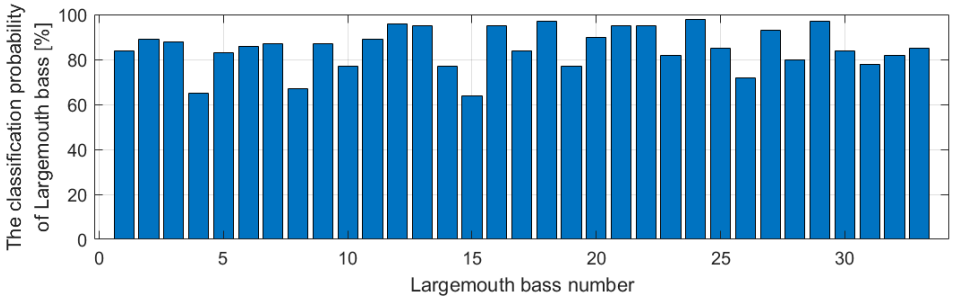


Figure 6. The probability of Largemouth bass.

Figure 7 shows the classification results and frame images of YOLOv2 when the proposed method finally classifies Largemouth bass as 0.83. It took 322 frames until the Largemouth bass appeared on the right edge and disappeared to the left edge, and the frame image was recognized as a Mandarin fish for frame 1 to frame 22 but was correctly recognized as Largemouth bass in frames after frame 23. In the proposed method, the classification performance is represented by the cumulative average of the classification performance up to the last frame, even if the classification is made wrong up to frame 22. Therefore, the proposed method is less likely to yield incorrect classification results. In particular, it has a high classification probability for very slow-moving fish. Figure 7i,j indicate the classification probability of YOLOv2 and the proposed method, respectively, for each frame. It can be seen that the proposed method accurately recognizes a Largemouth bass after frame 37.

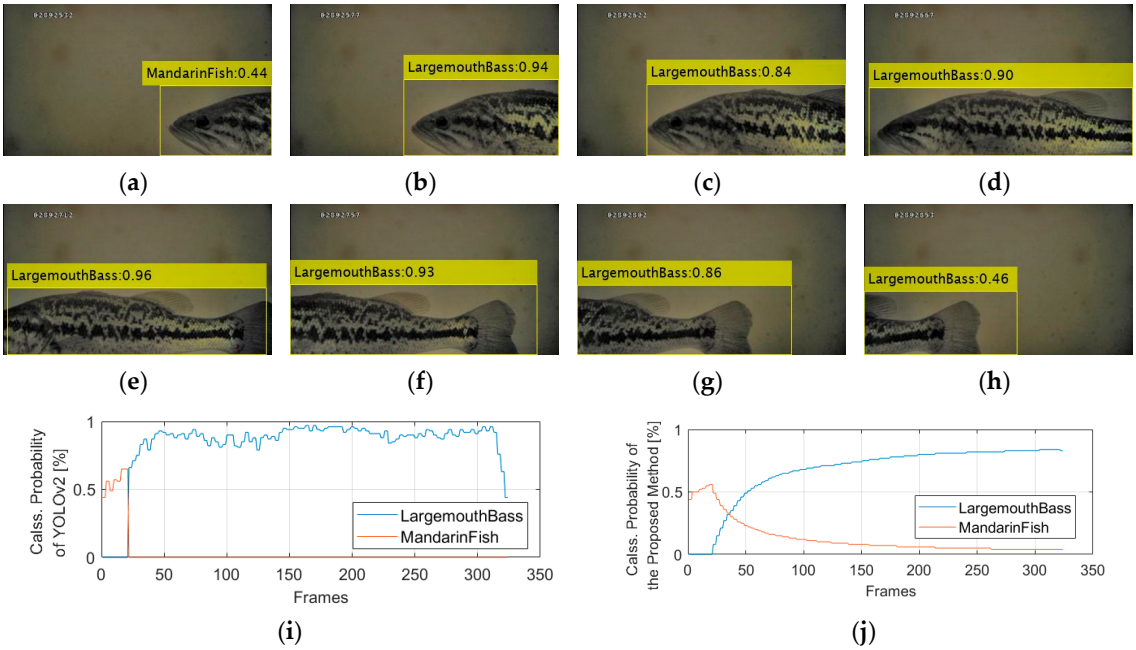
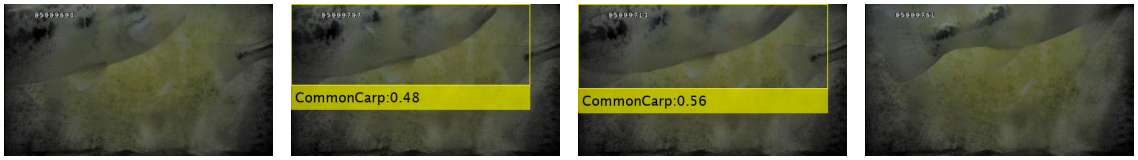


Figure 7. Classification performances for video frames: Largemouth bass (83%). (**a**) Frame 1, (**b**) Frame 46, (**c**) Frame 91, (**d**) Frame 136, (**e**) Frame 181, (**f**) Frame 226, (**g**) Frame 271, (**h**) Frame 322, (**i**) YOLOv2, and (**j**) the proposed method.

Figure 8 shows the YOLOv2 classification results of each frame for one fish whose proposed method does not recognize Largemouth bass. A Largemouth bass appears at the top left of the camera, comes very close, and disappears to the top right. YOLOv2 misclassified it as Common carp for frame 9 to frame 15, after which it did not recognize any fish. YOLOv2 did not classify correctly, because each frame image did not show the overall outline of the fish, but instead only one part. In the case of such video images, CNN and YOLOv2 show the wrong classification results and count of the number

of fish species. The proposed method may also not classify fish species. However, it does not count individuals for misclassification results.



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

Figure 8. Classification of YOLOv2 for Largemouth bass video: Non detection. (**a**) Frame 1, (**b**) Frame 9, (**c**) Frame 15, and (**d**) Frame 63.

Second, we evaluated the proposed method for Bluegill. The proposed method recognizes 62 (which is 93.94%) of a total of 66 fish as Bluegill, and did not classify 4 Bluegills. Most of the 62 Bluegills were recognized with classification probabilities of more than 60%, as shown in Figure 9. The proposed method using the heuristic method shows a very high recognition rate for the detection of fish, and can accurately count the population of fish.

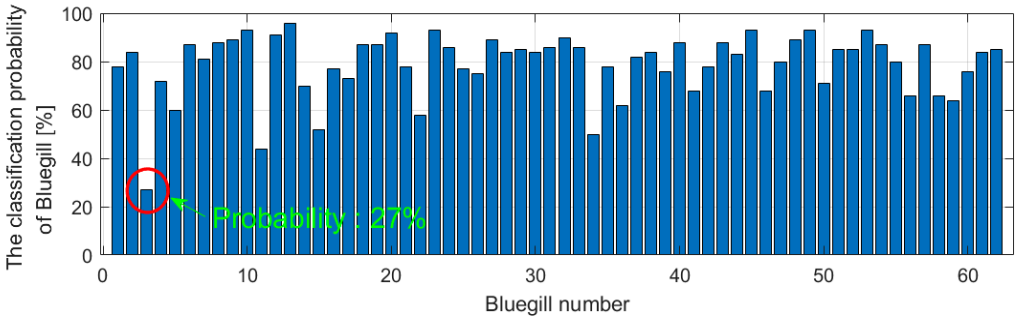


Figure 9. The probability of bluegill.

Figure 10 shows the classification results and frame images of YOLOv2, with the lowest classification probability of 27% for Bluegill. The Bluegill appears from the bottom right, disappears quickly down the left, and takes a total of 30 frames. From frame 1 to frame 8 it was recognized as Common carp, but from frame 9 to frame 30, it was recognized as Bluegill, or not as any object. In the case that YOLOv2 does not recognize the fish, the learning of YOLOv2 is not perfect.

Figure 10i,j show the classification performances of YOLOv2 and the proposed method, respectively, for each frame. If the CNN or YOLO network recognizes the images in frame 1 to frame 8 as Common carp, and fails to recognize the images in frames 11, 12, 15, 16, 17, 25, and 26 as fish, the fish species is recognized incorrectly. The proposed method has a low probability after frame 21, but is correctly recognized as Bluegill.

Figure 11 shows one example of four cases where the proposed method did not recognize the Bluegill. It took a total of 23 frames until the Bluegill appeared from the bottom right and disappeared down the left. YOLOv2 recognized the Bluegill in frames 5, 7, 8 and 23, but did not recognize any fish in the other frames. In the proposed method, if fish have not been recognized in the capture region of the image for a certain period of time, they are not classified as objects. The video image used in the experiment is very different from the image of Figure 1a, which trained YOLOv2. Therefore, YOLOv2 is not fully trained, due to the lack of learning images for very fast-moving fish, such as video images. The proposed method is very simple and intuitive, while retaining the advantages of YOLO in video images of underwater environments. The heuristic method has shown excellent performance in classifying and counting objects in video images. Therefore, the proposed method is considered to be useful not only for objects in the underwater environment, but also for other objects.

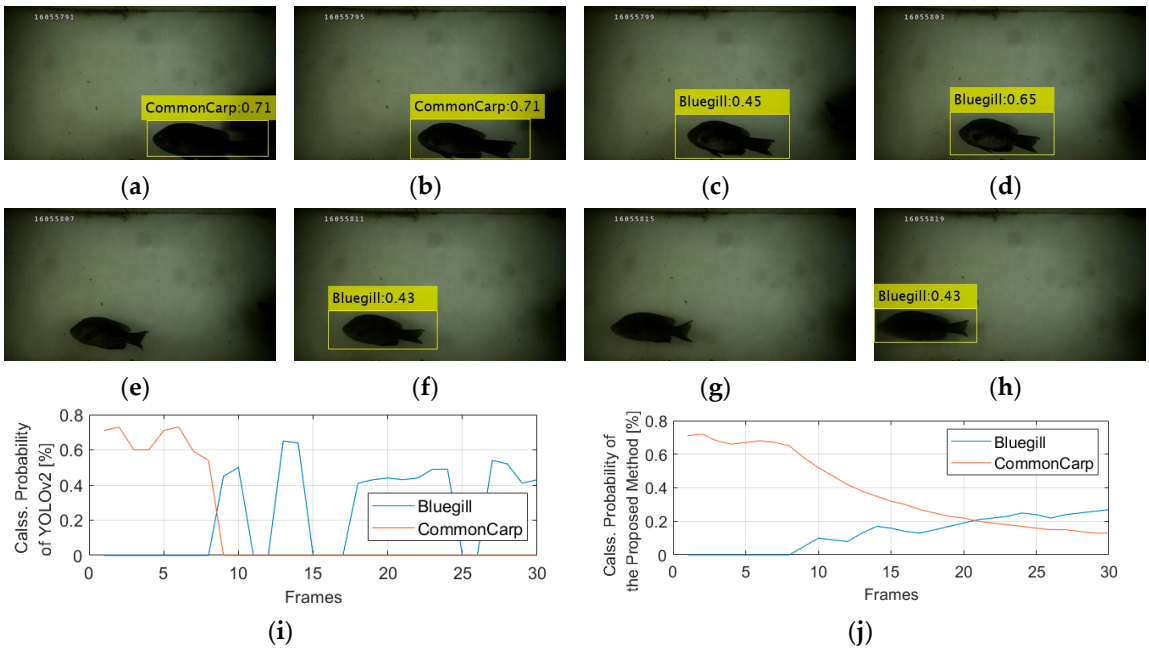


Figure 10. Classification performances for video frames: Bluegill (27%). (**a**) Frame 1, (**b**) Frame 5, (**c**) Frame 9, (**d**) Frame 13, (**e**) Frame 17, (**f**) Frame 21, (**g**) Frame 25, (**h**) Frame 30, (**i**) YOLOv2, and (**j**) the proposed method.

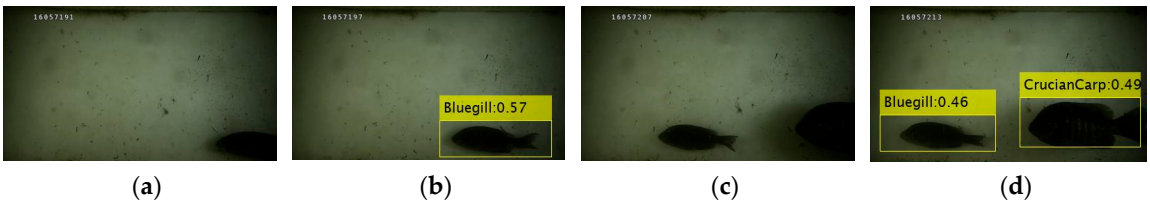


Figure 11. Classification of YOLOv2 for Bluegill video: Non detection. (**a**) Frame 1, (**b**) Frame 7, (**c**) Frame 18, and (**d**) Frame 23.

Figure 12 shows the results of a comparative experiment on recognition rates with other deep learning-based methods. GoogLeNet, Vgg16, and Vgg19 measured the recognition rate as the point in time when the fish is in the center of the video image. In the case of YOLOv2, the recognition rate was measured for all frames from the point when a fish is recognized in the video image to the moment it leaves. Furthermore, YOLOv2 and the proposed method used the same learned YOLO network. All methods showed a high recognition rate of 0.85 or higher. The proposed method has a recognition rate of 0.95 and other methods have a recognition rate of 0.88 ~ 0.89. In the proposed method, the result of the previous frame affects the recognition result of the current frame. This has a function of canceling the recognition error in a single frame, and there is a performance improvement of about 0.08 compared to other methods.

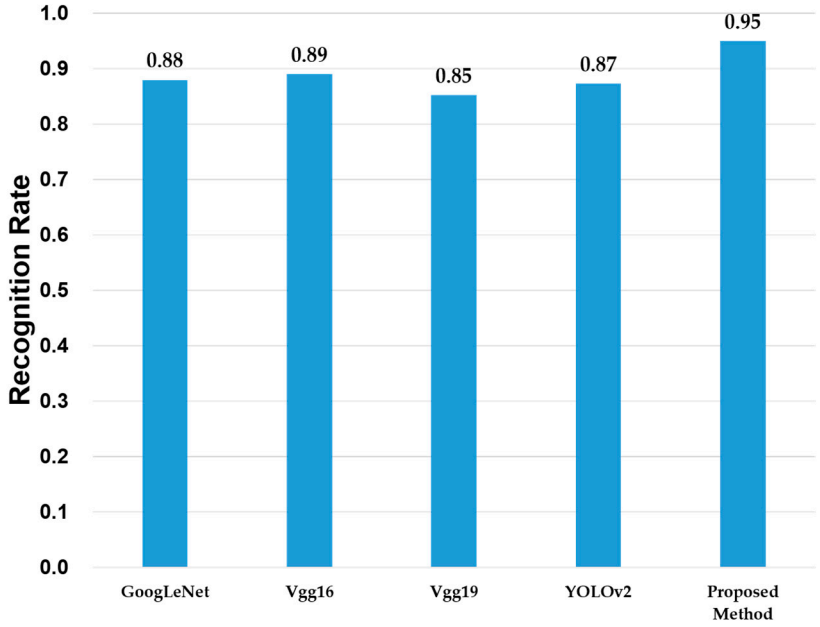


Figure 12. Experimental results compared with other methods (GoogLeNet, Vgg16, Vgg19, and YOLOv2) for recognition rate.

5. Conclusions

YOLO shows excellent performance in object recognition, but the performance varies depending on network learning. It recognizes objects by processing images of each frame independently of each other. This means that the classification results in the previous frame do not affect those of the current frame. By accumulating the object classification results from the past frames to the current frame, we propose a method to accurately classify objects, and count their number in the sequential video images. The proposed method shows very good classification performance in video images taken in underwater environments. It has high classification probabilities of 93.94% and 97.06% in the test videos of Bluegill and of Largemouth bass, respectively. The proposed method is also affected by the performance of YOLO, but its performance was improved by applying the heuristic method that mimics human experience and learning.

Author Contributions: Conceptualization—J.-H.P.; methodology—J.-H.P.; software—J.-H.P.; validation—J.-H.P. and C.K.; writing—original draft preparation—J.-H.P.; writing—review and editing—J.-H.P. and C.K. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by Korea Environment Industry & Technology Institute (KEITI) through the Exotic Invasive Species Management Program, funded by Korea Ministry of Environment (MOE) (2017002270002).

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

References

1. Buckland, S.T.; Turnock, B.J. A Robust Line Transect Method. *Biometrics* **1992**, *48*, 901–909. [CrossRef]
2. Järvinen, O.; Väisänen, R.A. Estimating relative densities of breeding birds by the line transect method. *Oikos* **1975**, *7*, 43–48. [CrossRef]
3. Buckland, S.T.; Garthwaite, P.H. Quantifying Precision of Mark-Recapture Estimates Using the Bootstrap and Related Methods. *Biometrics* **1991**, *47*, 255–268. [CrossRef]
4. Miller, C.R.; Joyce, P.; Waits, L. P. A new method for estimating the size of small populations from genetic mark-recapture data. *Mol. Ecol.* **2005**, *14*, 1991–2005. [CrossRef] [PubMed]
5. Vitkalova, A.V.; Feng, L.; Rybin, A.N.; Gerber, B.D.; Miquelle, D.G.; Wang, T.; Yang, H.; Shevtsova, E.I.; Aramilev, V.V.; Ge, J. Transboundary cooperation improves endangered species monitoring and conservation actions: A case study of the global population of Amur leopards. *Conserv. Lett.* **2018**, *11*, 12574. [CrossRef]

6. Bischof, R.; Brøseth, H.; Gimenez, O. Wildlife in a Politically Divided World: Insularism Inflates Estimates of Brown Bear Abundance. *Conserv. Lett.* **2016**, *9*, 122–130. [CrossRef]
7. Siddiqui, S.A.; Salman, A.; Malik, M.I.; Shafait, F.; Mian, A.; Shortis, M.S.; Harvey, E.S. Automatic fish species classification in underwater videos: Exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES J. Mar. Sci.* **2018**, *75*, 374–389. [CrossRef]
8. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
9. Rekha, B.S.; Srinivasan, G.N.; Reddy, S.K.; Kakwani, D.; Bhattad, N. Fish Detection and Classification Using Convolutional Neural Networks. In Proceedings of the International Conference on Computational Vision and Bio Inspired Computing, Coimbatore, India, 25–26 September 2019; pp. 1221–1231.
10. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations, ICLR 2015-Conference Track Proceedings, San Diego, CA, USA, 7–9 May 2015.
11. Park, J.-H.; Choi, Y.-K. Efficient Data Acquisition and CNN Design for Fish Species Classification in Inland Waters. *J. Inform. Commun. Converg. Eng.* **2020**, *18*, 106–114. [CrossRef]
12. Briseño-Avena, C.; Schmid, M.S.; Swieca, K.; Sponaugle, S.; Brodeur, R.D.; Cowen, R.K. Three-dimensional cross-shelf zooplankton distributions off the Central Oregon Coast during anomalous oceanographic conditions. *Prog. Oceanogr.* **2020**, *188*, 102436. [CrossRef]
13. Swieca, K.; Sponaugle, S.; Briseño-Avena, C.; Schmid, M.S.; Brodeur, R.D.; Cowen, R.K. Changing with the tides: Fine-scale larval fish prey availability and predation pressure near a tidally modulated river plume. *Mar. Ecol. Prog. Ser.* **2020**, *650*, 217–238. [CrossRef]
14. Schmid, M.S.; Cowen, R.K.; Robinson, K.; Luo, Y.J.; Briseño-Avena, C.; Sponaugle, S. Prey and predator overlap at the edge of a mesoscale eddy: Fine-scale, in-situ distributions to inform our understanding of oceanographic processes. *Sci. Rep.* **2020**, *10*, 1–16. [CrossRef] [PubMed]
15. Rezende, E.; Ruppert, G.; Carvalho, T.; Theophilo, A.; Ramos, F.; de Geus, P. Malicious Software Classification Using VGG16 Deep Neural Network's Bottleneck Features. In *Information Technology-New Generations, Proceedings of the Advances in Intelligent Systems and Computing*; Latifi, S., Ed.; Springer: Cham, Switzerland, 2018. [CrossRef]
16. Liu, C.; Cao, Y.; Luo, Y.; Chen, G.; Vokkarane, V.; Ma, Y. Deepfood: Deep learning-based food image recognition for computer-aided dietary assessment. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Proceedings of the Inclusive Smart Cities and Digital Health. ICOST 2016*; Chang, C., Chiari, L., Cao, Y., Jin, H., Mokhtari, M., Aloulou, H., Eds.; Springer: Cham, Switzerland, 2016; pp. 37–48. [CrossRef]
17. Szegedy, C.; Ioffe, S.; Vanhoucke, V.; Alemi, A.A. Inception-v4, inception-ResNet and the impact of residual connections on learning. In Proceedings of the 1st AAAI Conference on Artificial Intelligence, AAAI 2017, San Francisco, CA, USA, 4–9 February 2017; pp. 4278–4284.
18. Girshick, R. Fast R-CNN. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1440–1448. [CrossRef]
19. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask R-CNN. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 386–397. [CrossRef] [PubMed]
20. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 30 June 2016; pp. 779–788. [CrossRef]
21. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017, pp. 7263–7271. [CrossRef]
22. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 6517–6525. [CrossRef]
23. Park, J.-H.; Choi, Y.-K.; Kang, C. Fast Cropping Method for Proper Input Size of Convolutional Neural Networks in Underwater Photography. *J. Soc. Inf. Disp.* **2020**, *28*, 872–881. [CrossRef]

24. Lu, X.; Lin, Z.; Shen, X.; Mech, R.; Wang, J.Z. Deep multi-patch aggregation network for image style, aesthetics, and quality estimation. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 990–998. [CrossRef]
25. Kang, L.; Ye, P.; Li, Y.; Doermann, D. Convolutional neural networks for no-reference image quality assessment. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1733–1740. [CrossRef]
26. Lu, X.; Lin, Z.; Jin, H.; Yang, J.; Wang, J.Z. Rapid: Rating pictorial aesthetics using deep learning. In *Proceedings of the 22nd ACM International Conference on Multimedia (MM'14)*; Association for Computing Machinery: New York, NY, USA, 2014; pp. 457–466. [CrossRef]
27. Ma, S.; Liu, J.; Chen, C.W. A-lamp: Adaptive layout-aware multi-patch deep convolutional neural network for photo aesthetic assessment. In Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, 21–26 July 2017; pp. 722–731. [CrossRef]
28. Park, J.-H.; Hwang, K.-B.; Park, H.-M.; Choi, Y.-K. Application of CNN for Fish Species Classification. *J. Korea Inst. Inf. Commun. Eng.* **2019**, *23*, 39–46.
29. Rosenblatt, M. A central limit theorem and a strong mixing condition. *Proc. Natl. Acad. Sci. USA* **1956**, *42*, 43. [CrossRef] [PubMed]
30. Hoeffding, W.; Robbins, H. The central limit theorem for dependent random variables. *Duke Math. J.* **1948**, *15*, 773–780. [CrossRef]

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

© 2020 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 (http://creativecommons.org/licenses/by/4.0/).