

# PHOTOMETRIC AND GEOMETRIC DATA AUGMENTATIONS FOR SMALL OBJECT DETECTION

RongChuan Zhang <sup>1</sup>

<sup>1</sup>Universiti Kebangsaan Malaysia  
A173165@siswa.ukm.edu.my

**Abstract:** With the development of science and technology, more and more organizations use drones to do surveillance, rescue, or some mission at sea. However, due to the changing climate on the ocean and the small size of the ship photographed from the top of the UAV, it is challenging to detect the target. Although many methods have been proposed at present, such as modifying neural network model or using mixed dataset training method, and achieved good results, there is still a lot of room for improvement. But one of the reasons why the small object detection didn't achieve the better results was that there was no image of different environments and it was hard to have a rich different position and shape. This study aims to solve this problem by using geometric and photometric transformation to do data augmentation, because they could be used to simulate the impact in various special environments or the boat in different location, even the different shapes of the boats. So, geometric and photometric data augmentation is proposed in this report. The Seagull Dataset is chosen for this project because it is taken by a top-view UAV and contains image data in various weather conditions. At the same time, this project will use Yolov5 as an object detection model, experiment with geometric and photometric transformations, and propose a cumulative Threshold combination strategy for screening and combining geometric and photometric transformations with doing data augmentation. To improve the detection performance of Yolov5 on small objects. After many trials and comparisons, Yolov5 was trained with 6900 images and tested with 5500 images, resulting in mAP@0.5=0.7. After data augmentation, the effect was improved by 15%. The project used fewer images and improved performance by 3% compared to other related efforts. At the end of this project, A Web-based detector with trained models will be implemented using Flask, which could output the bounding box and the position coordinates of the object in the image.

**Keywords:** Data augmentation, Yolov5, Tiny object detection

## 1. INTRODUCTION

With the development of scientific level and technological progress. The drone is playing an increasingly important role in people's lives. In many cases, individuals or the army try to use the drone to do surveillance, rescue, or some mission at sea. So, top-view object detection is vital for UAVs to perform these tasks, especially for tiny target detection. For two-stage models, such as R-CNN[7], FAST R-CNN[5] and FASTER R-CNN[6]. R-CNN[7] follows the traditional idea of object detection, and also uses extraction frames to detect objects in four steps: feature extraction for each frame, image classification, and non-maximum suppression. However, in the step of feature extraction, the traditional feature is replaced by the feature extracted by deep convolutional network. This feature vector is fed into a multi-class SVM classifier, which predicts the probability values of

the objects in the candidate region belonging to each class. Fast RCNN[5] is a substantial improvement over RCNN, using two different fully connected layers and using the SoftMax classifier instead of the SVM classifier. For Fast-RCNN selective search to find all candidate boxes and time-consuming problems, Faster R-CNN[6] designs a network RPN to extract candidate regions instead of time-consuming selective search, and merges candidate box extraction into the deep network, which greatly improves the detection speed. But they are all relatively slow, because they need to extract object first and then judge object. The one-stage model is very suitable for this project because of its fast detection speed. Compared with other one-stage models, Yolo[3][4][14] satisfy the demand of both fast and accurate. At the same time, Yolov5 also shows better performance in small object detection than other versions of Yolo[22]. So, Yolov5 was used as a detector for experiments. The Seagull dataset includes multi-camera and multi-spectrum image sequences for research on sea monitoring and surveillance[9]. It records the boat, sea, and other objects from the view of the UAV which shown in Figure 1.1.

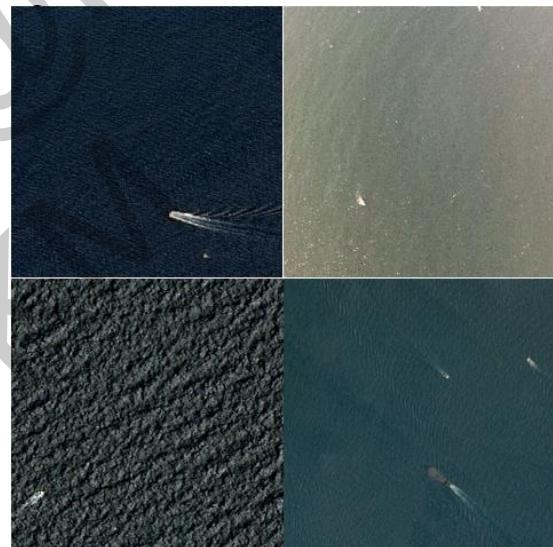


Figure 1.1 example of seagull dataset which contains the boats in different Lighting conditions and environmental conditions.

So the seagull dataset is used to cope this problem. There are lots of previous works on tiny object detection. Some works mentioned that for images containing small targets, multiple small targets were copied and pasted to increase the frequency of small targets in the image. In this way, positive samples of small targets would also increase during training, making the model more focused on learning small targets[16]. In addition to that, there are a lot of people working on the data augmentation section. Geometric and photometric transformation to do data augmentation have been found to improve CNN detection performance significantly in some crack detection applications[12]. The author firstly tested the effects of various data enhancement methods in

crack detection and analyzed them. After that, various data enhancement techniques are combined by greedy algorithms to achieve better results with very little data. Another paper proposed geometric like flip and rotation, and photometric transformation like histogram equalization and some filters to do data augmentation to improve the performance on detection of blood cells[21]. At the same time, Some people try to improve the performance of small targets by modifying the network model[11]. Based on Yolov3 network, the residual module[10] is improved by establishing dense connection and adding Carbon Border Adjustment Mechanism (CBAM) to enhance the use of feature and global information. At the same time, the feature pyramid network (FPN)[1] is replaced by the second-level recursive feature pyramid network to reduce the influence of complex environment. Or, a new water surface small target detection algorithm based on convolutional neural network, self-attention feature fusion detection algorithm is proposed[20]. Geometric and photometric data augmentation should also bring nice improvements to other small-target detection. For the geometric transformation, it changes the spatial relationships between pixels in an image. This data augmentation allows the model to learn more features about the boat in different location, even the different shapes of the boats. For the photometric transformation, it realizes image transformation by using filter, changing the color space of the image or changing the bright contrast of the image. These data augmentation methods are used to simulate the impact in various special environments, enhance the robustness of the model, and improve the ability of the model to detect various noises or under different conditions. So, to overcome the lack of datasets in quantity and diversity, the geometric and photometric transformation are used to do data augmentation. In this case, this project use cumulative threshold combinations strategy to select and combine those data augmentation techniques. After many experiment, the results showed that the results of  $mAP@0.5 = 0.7$  were obtained with 5,500 images. The effect of other papers improved 3% performance[2]. At the end of the experiment, the boat detector which could be used to detect the boat of the top-view sea monitoring will be implemented using Flask.

## 2. METHODS

### 2.1. Methodology

In this project, one of the most important tasks is article review, the project can learn more ideas from others' work. And then, the second step is to determine the dataset and model required by the project. Dataset selection based on objective pairs will have a great influence on the final result. After selected the dataset and the model, label and clean the dataset and fine tune the hyper-parameter is the next step. In this case, the core of the project will be testing and combining data augmentation techniques in order to find combinations that best fit the selected model and

dataset. The data augmentation strategy is used from two stage which is network training and model inference process[12]. In the part of network training, model will be trained without data augmentation and with only one of the photometric and geometric data augmentation techniques added. So, the effect of each technique will be evaluated. In the second stage, to improve the mean average precision of the boat detection, this stage focus on the combination of the data augmentation based on the cumulative threshold combination strategy. After that, the project has been obtained the best combination of the data augmentation. So, evaluate and compare with other works is necessary in this part. Since the whole project needs to do a lot of experiments, therefore, in order to save time cost. With the exception of final evaluation, all experiments were trained with 1200 images and validated with 1000 images. For the final evaluation part of the project, we trained the entire dataset with 6,900 images, tested it with 5,500 images and got the final result. At the end of the project, the web-based detector will be developed based on the trained model. The flowchart for Methodology is shown in Figure 2.1.1.

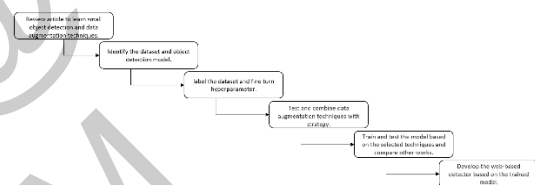


Figure 2.1.1 The flowchart of Methodology

### 2.2. Label the dataset

We all know that images and annotations are the compulsory components of the object detection model. However, the seagull dataset only supports images and masked images. Generally, the images are labeled manually using labelling or some other software. But the seagull dataset has ten thousand images so it will take a lot of time to label. In this case, I proposed a method to automatically generate the labels of the images based on the masked images. This method automatically labels the position of the target by code with the help of the distinctive black and white in the masked image. The masked images show in Figure 2.2.1

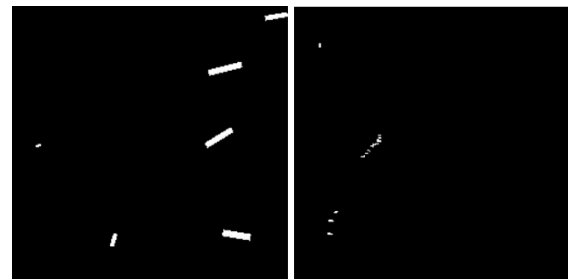


Figure 2.2.1 The example of masked images

In this picture, you can clearly see that the boat object is represented in white, while the

background is represented in black. So, the method of labeling is to judge and label with the help of this black and white color difference. The code of labeling the images using OpenCV. After generated the label, the result shown in Figure 2.2.2 The green one represents the bounding box generated. In these results, there were some misjudgements but most of the labeling is accurate.

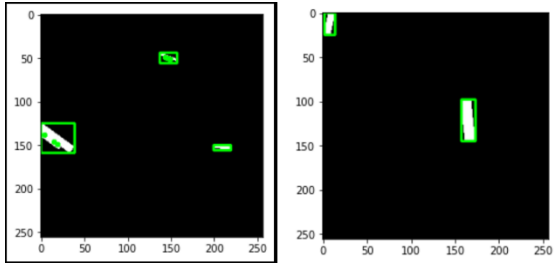


Figure 2.2.2 After generating the bounding box, the green bounding box is a visualization of the generated label.

### 2.3. Mosaic data augmentations

The mosaic data augmentation is another way of loading data. This method stitch the four images together. Each image has its corresponding target's bounding box, and after stitching the four images together, a new image is obtained, which is randomly crop out the same size as the input image, but contains the information of the four images, and also obtains the corresponding target's bounding box of this image. The Figure 2.3.1 shows the process of mosaic. And Figure 2.3.2 shows the example of mosaic in seagull dataset.



Figure 2.3.1 The process of mosaic data augmentation

Then we pass this new image into the neural network to learn, which is equivalent to passing in four images at once for learning. This method greatly enriches the background of the detected objects. It is also possible to compute four images at the same time during the normalized BN computation.

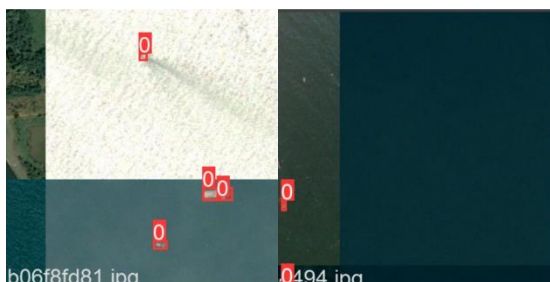


Figure 2.3.2 The example of mosaic data augmentation

### 2.4. Geometric transformations

In this chapter, we'll discuss geometric transformations that are used in this projects. Geometric transformations have been shown to work well in experimental reports on crack detection such as Rotation, Flip[12]. In addition, the project also incorporates techniques such as Shear, Scale, Perspective, and so on. We'll go through those methods one by one, and then use Figure 2.4.1 as an example to test each geometric transformation and show its effect.



Figure 2.4.1 the example of original image

1. **Scale:** It is the operation of zooming in and out of a picture, which can be specified in a certain direction, such as horizontal (x-axis) and vertical (y-axis), and it is possible to zoom in and out of 3-dimensional graphics by z-axis. The Figure 2.4.2 shows the example of scale.

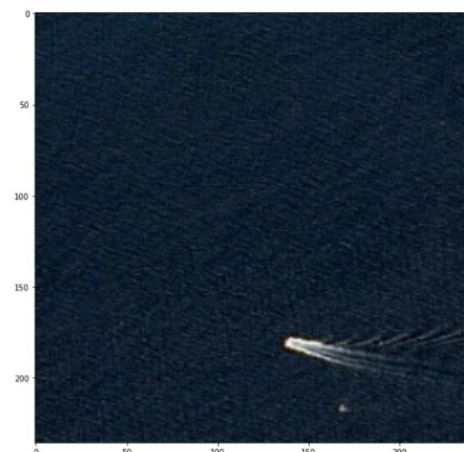


Figure 2.4.2 The result of the scale operation on the original image

2. **Translation:** Move the position where the picture is located and get different pictures by changing the relative position of the target in the image, for example, the target is in the middle, top left, bottom right, etc. of the picture. The example of translation shown in Figure 2.4.3. For example, an



image moves right 10 pixels and moves down 30 pixels, The transformation matrix is going to be:

$$M = \begin{bmatrix} 1 & 0 & 10 \\ 0 & 1 & 30 \end{bmatrix} \quad (2.1)$$

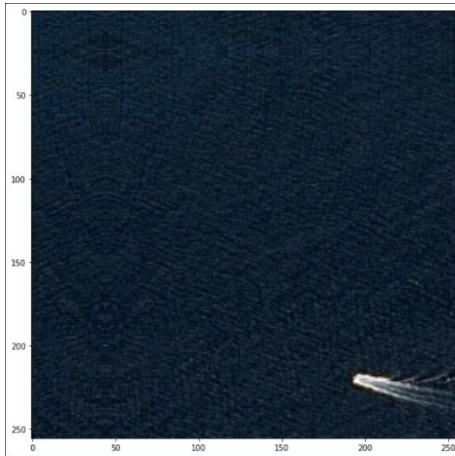


Figure 2.4.3 The result of the translation operation on the original image

$$(x, y) \rightarrow (x', y') \quad (2.5)$$

$$x' = x \cos \alpha - y \sin \alpha \quad (2.6)$$

$$y' = x \sin \alpha + y \cos \alpha \quad (2.7)$$

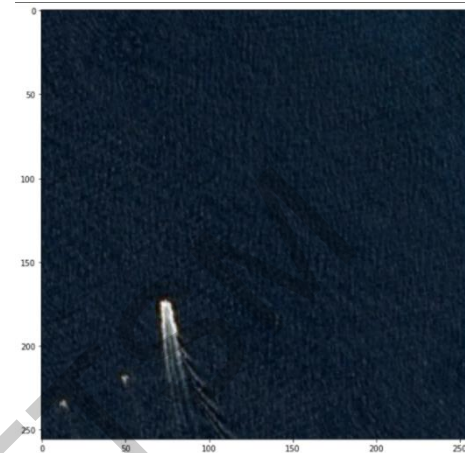


Figure 2.4.5 The result of the rotation operation on the original image

3. **Flip:** Flip the image horizontally to get a mirror image of the original image or vertically or flip the image both horizontally and vertically. The example of flip shows in Figure 2.4.4. The formular show below.

$$A = (a_{ij})_{m \times n} \quad (2.2)$$

$$B = (b_{ij})_{m \times n} \quad (2.3)$$

$$b_{ij} = a_{n+1-j} \quad (2.4)$$

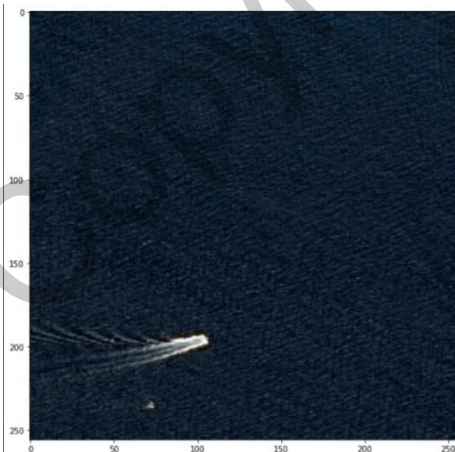


Figure 2.4.4 The result of the flip operation on the original image

5. **Shear:** The operations are divided into horizontal and vertical misalignment, and the operations performed, such as horizontal misalignment, are linear transformations of the horizontal coordinates of each point on the image while keeping the vertical coordinates unchanged. This results in a non-vertical projection of the image in the plane. The example of shear shows in Figure 2.4.6. If shear in x direction. The transformation matrix is going to be:

$$M = \begin{bmatrix} 1 & \tan \alpha & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (2.8)$$

If shear in y direction. The transformation matrix is going to be:

$$M = \begin{bmatrix} 1 & 0 & 1 \\ \tan \alpha & 1 & 0 \end{bmatrix} \quad (2.9)$$

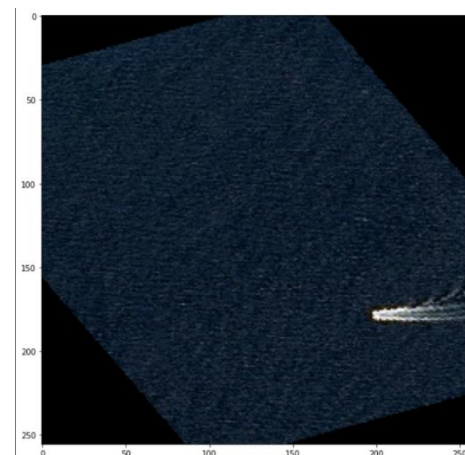


Figure 2.4.6 The result of the shear operation on the original image

4. **Rotation:** In practice, the detected target may have a certain angle of tilt in the image, so it can be rotated to make the target have a different tilt angle in the image and improve the generalization ability of the model. The example of rotation shows in Figure 2.4.5. The formular show below.

6. **Perspective:** Perspective Transformation is the projection of a plane onto a specified plane by means of a projection matrix. The example of perspective shows in Figure 2.4.7.

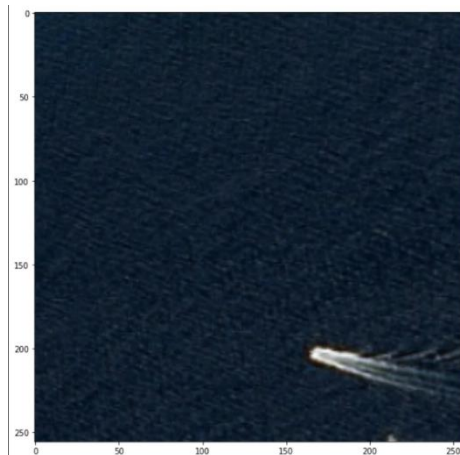


Figure 2.4.7 The result of the perspective operation on the original image

## 2.5. Photometric transformations

The photometric transformation realizes image transformation by using filter, changing the color space of the image or changing the bright contrast of the image. Those methods are used to simulate the impact in different special environments, and various noises under different conditions. In some biological detection experiments on blood cells[21], we can find that some photometric transformations, such as histogram equalization and some filters, perform very well. Based on the situation of the project, some other techniques, such as Gaussian noisy, to Gray, Gamma Transformation, etc., are also added. We'll go through those methods one by one, and then use Figure 2.4.1 as an example to test each photometric transformation and show its effect.

1. **Brightness contrast:** This method enhances the data by adjusting the brightness and contrast of the image to improve the robustness of the model. The example of brightness contrast shows in Figure 2.5.1.

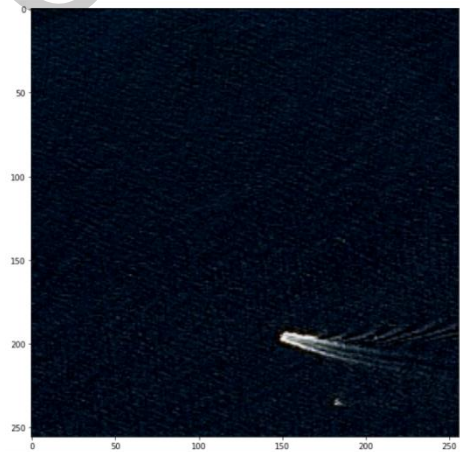


Figure 2.5.1 The result of the brightness contrast operation on the original image

2. **To Gary:** In this method, the Gray dynamic range of the original image is linearly extended to a specified range or the whole dynamic range. The example of to gray shows in Figure 2.5.2.

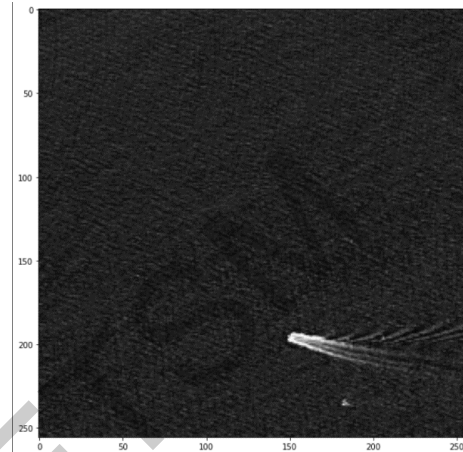


Figure 2.5.2 The result of the to gray operation on the original image

3. **Histogram equalization:** Histogram enhancement technology is a kind of image enhancement technology by changing all or part of the image contrast, one of the typical processing methods is histogram equalization. The idea of histogram equalization is to map the gray scale of pixels in the original image to make the probability density of the Gray scale of the transformed image evenly distributed, that is, the Gray level of the transformed image is uniform, which means that the dynamic range of the Gray scale of the image is increased, and the contrast of the image is improved. The example of histogram equalization shows in Figure 2.5.3.

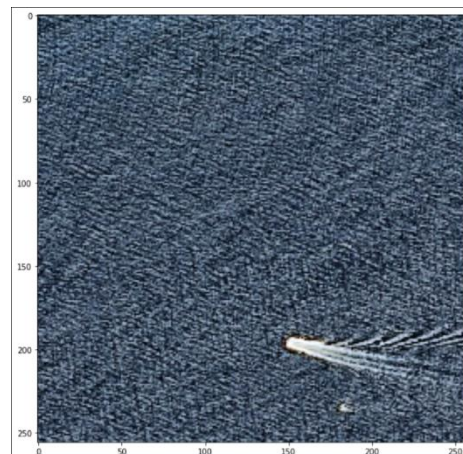


Figure 2.5.3 The result of the histogram equalization operation on the original image

4. **Gaussian blur:** Image blur, is used to eliminate noise, reduce the level of detail, fuzzy image of a processing technology. In the image, the closer the distance is to the object to be processed, the greater its influence on the pixel is. Therefore, the

weighted average method is usually adopted when the graph is smooth. Gaussian blur is also a template of weighted average method, but it uses normal distribution as a template for pixel mapping. The example of Gaussian blur shows in Figure 2.5.4.

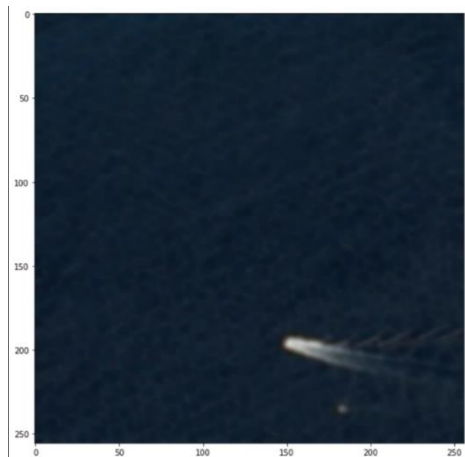


Figure 2.5.4 The result of the Gaussian blur operation on the original image

5. **Gaussian noisy:** Noise is a random overlay of noise on top of the original image. At the same time, the network can learn to avoid noise by adding noise, improve performance and avoid the occurrence of over-fitting. The example of Gaussian noisy shows in Figure 2.5.5.

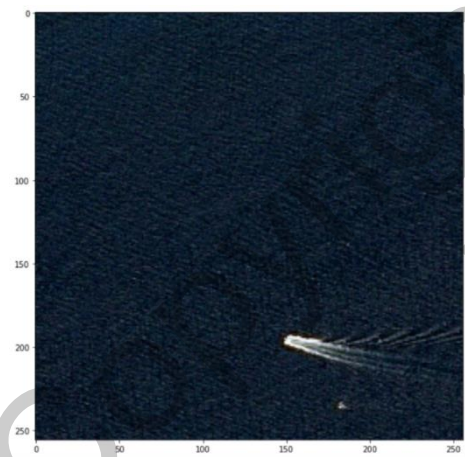


Figure 2.5.5 The result of the Gaussian noisy operation on the original image

6. **Gamma transformation:** The Gamma transform is used to enhance the image. It enhances the dark details. In short, it changes the image from a linear response to exposure intensity to a response more similar to that experienced by the human eye through nonlinear transformation. The value of  $S$  ranges from 0 to 1, so we need to normalize first and then take the exponent. The example of gamma transformation shows in Figure 2.5.6.

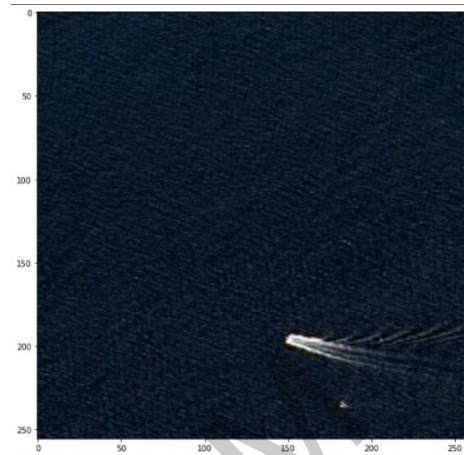


Figure 2.5.6 The result of the gamma transformation on the original image

7. **HSV transformation:** HSV (Hue, Saturation, Value) is a color space created by A. R. Smith in 1978 based on the intuitive properties of color, also known as the Hexcone Model. In the HSV model, color is composed of Hue, Saturation, and Value. This method is transformed the RGB image to HSV image to achieve the effect of data augmentation. The example of hsv transformation shows in Figure 2.5.7.

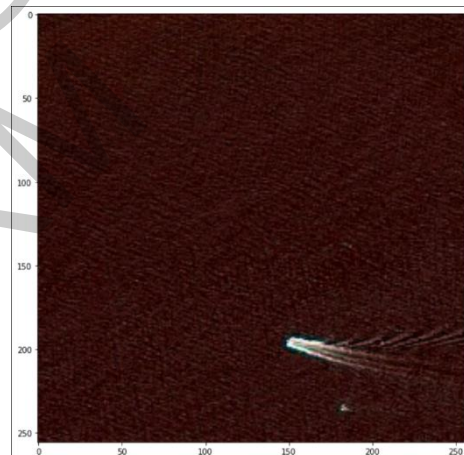


Figure 2.5.7 The result of the HSV transformation on the original image

## 2.6. Strategy on combination of data augmentation

At the beginning of this experiment, the two different upload image methods are compared: the regular Upload Image method and the Mosaic data augmentation uploading method. The suitability of Mosaic for these data augmentation methods and seagull dataset was tested by training all data augmentation methods individually. The result show that mosaic perform well on this model. So the strategy will be used based on the experiment of mosaic. The result show on the table 2.6.1 and table 2.6.2.



Table 2.6.1 the result of photometric transformation + mosaic on Yolov5

| Method                                 | mAP@0.5 |
|--|---------|
| Yolov5 without data augmentation       | 0.410   |
| Yolov5 + Mosaic + Gamma transformation | 0.529   |
| Yolov5 + Mosaic + HSV Transformation   | 0.529   |
| Yolov5 + Mosaic + Gaussian Noisy       | 0.517   |
| Yolov5 + Mosaic + Histogram Equation   | 0.517   |
| Yolov5 + Mosaic + Brightness Contrast  | 0.511   |
| Yolov5 + Mosaic                        | 0.507   |
| Yolov5 + Mosaic + Gaussian Blur        | 0.506   |
| Yolov5 + Mosaic + ToGray               | 0.497   |

Table 2.6.2 the result of geometric transformation + mosaic on Yolov5

| Method                           | mAP@0.5 |
|----------------------------------|---------|
| Yolov5 without data augmentation | 0.410   |
| Yolov5 + Mosaic + Scale          | 0.520   |
| Yolov5 + Mosaic + Perspective    | 0.518   |
| Yolov5 + Mosaic + Translation    | 0.517   |
| Yolov5 + Mosaic + Flip           | 0.513   |
| Yolov5 + Mosaic                  | 0.507   |
| Yolov5 + Mosaic + Rotation       | 0.494   |
| Yolov5 + Mosaic + Shear          | 0.461   |

So, our next step was based on Mosaic's image loading method. The first thing is to sort the results of the previous experiment according to the mAP obtained by the validation. Then, based on the table obtained after sort, I set thresholds to 0.52, 0.515, and 0.51 respectively and divided all data augmentation techniques into three parts based on the threshold I set. After those three combined data augmentation methods will be used to train and validate in order to see the best combination of those methods. The result shows in Table 2.6.3.

Table 2.6.3 the result of cumulative thresholds combination on Yolov5

| mAP@0.5 threshold value    | Methods  | mAP@0.5      | mAP@0.5 -0.95 |
|----------------------------|--|--------------|---------------|
| <b>&gt;=0.520 (strong)</b> | <b>Yolov5 +Mosaic +Gamma transformation +HSV transformation +Scale</b> | <b>0.552</b> | <b>0.307</b>  |
| >=0.515 (strong + mid)     | Yolov5 +Mosaic +Gamma transformation +HSV transformation +Scale        | 0.531        | 0.285         |

|                               | +Gaussian Noisy<br>+Histogram<br>Equation<br>+Translation<br>+Perspective  |       |       |
|-------------------------------|--|-------|-------|
| >=0.510 (strong + mid + weak) | Yolov5 +Mosaic +Gamma transformation +HSV transformation +Scale +Gaussian Noisy +Histogram Equation +Translation +Perspective +Flip +Brightness contrast | 0.549 | 0.294 |

Therefore, it is easy to find that Yolov5 + Mosaic + gamma transformation + HSV transformation + Scale is the best combination of those techniques.

## 2.7. web-based detector design

For this web-based detector design, the flask was selected as the framework. Flask is a lightweight web application framework written in Python. Based on Werkzeug WSGI toolkit and Jinja2 template engine. The structure of flask project shows in Figure 2.7.1.

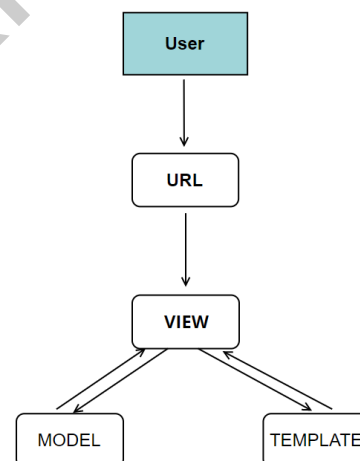


Figure 2.7.1 the structure of flask project

After config the project and starting the server, the user could access the web using the URL. And then, the View will access the Template to get the interface of the project and show it to the user. If the user needs to detect using this system, the View will also access the model and feed the image to the model. At the end of the detection, the model will feedback the detection result to the View. And the View could send the result to another router and show the result to the user. This is how the system

works. It does not take up a lot of memory and at the same time implements all the functions needed for the project. The friendly interface also be designed and show in Figure 2.7.2.

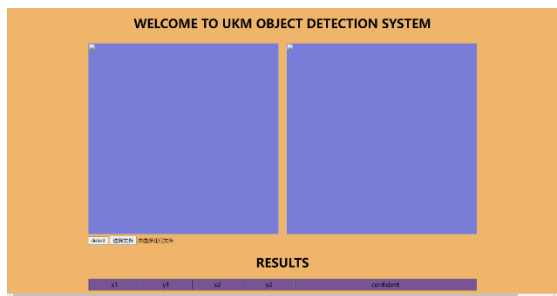


Figure 2.7.2 The interface of the system

### 3. FINDINGS AND ARGUMENT

#### 3.1. Mean average precision

The mAP is a common evaluation metric used in target detection models, and its full name is Mean Average Precision. First we need to know the definition of Precision and Recall, also known as accuracy and completeness.

$$Precision = \frac{TP}{TP+FP} \quad (6.1)$$

$$Recall = \frac{TP}{TP+FN} \quad (6.2)$$

Next, I will introduce the IOU (Intersection over union). The IOU measures the overlap degree of two detection frames for target detection, and the formula is as follows:

$$IOU = \frac{area(predict \cap ground\ truth)}{area(predict \cup ground\ truth)} \quad (6.3)$$

After this, according to the set threshold and the precision and recall we get, we can get a curve about the precision and recall. In this case, the area under the P-R curve is the value of AP. The mAP is the average of all categories of AP. The formula is as follows:

$$AP = \sum_{i=1}^{n-1} (r_{i+1} - r_i) P_{interpolated}(r_i + 1) \quad (6.4)$$

$$mAP = \frac{\sum_{i=1}^k AP_i}{k} \quad (6.5)$$

#### 3.2. Evaluation

At the end of the experiment, the project uses all of datasets, 6900 image training models and 5500 image test models and get the result which is mAP@0.5 = 0.7. The performance was improved by 15% compared with the Yolov5 without data augmentation. The project used fewer images and improved performance by 3% compared to other related efforts[2]. The result show in Table 3.2.1.

Table3.2.1 The final evaluation of the model

| Model                            | mAP@0.5     | mAP@0.5-0.95 |
|----------------------------------|-------------|--------------|
| Yolov5 without data augmentation | 0.55        | 0.321        |
| Yolov5 with data augmentation    | <b>0.70</b> | <b>0.418</b> |
| Other paper result               | 0.67        | 0.357        |

#### 3.3. Detection result

It can be seen from Figure 3.3.1 that the model has a very good effect on small targets and can detect the target to be detected.



Figure 3.3.1 the correct detection results

However, Figure 3.3.2 shows some of the wrong predictions. You can see from the top that the model did not detect any ships, and from the bottom that the detection was wrong.

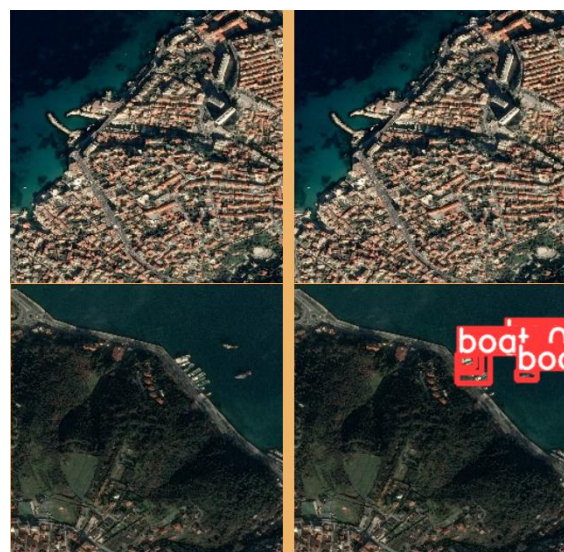


Figure 3.3.2. The incorrect detection results



Through the study and analysis of the detection results, I found that although the accuracy of the current model has been improved, detection errors often occur when detecting both tiny and dense objects.

#### 4. CONCLUSIONS

To sum up, after multiple experiments, it found that the combination of the four selected data augmentation techniques, including mosaic, gamma transformation, hsv transformation and scale, significantly improved the model, with mAP@0.5 and mAP@0.5-0.95 improving by 0.15 and 0.09 relative to no data augmentation. At the same time, compared with the results of other researchers, fewer data sets were used, but better results were achieved. But the model still doesn't perform well on both small and dense object, so there's plenty of room for improvement.

In my opinion, the project can be further studied from two aspects. First, for the model, the structure of the model can be modified to improve the detection results of both small and dense objects. Try adding Transformer to improve the performance of the model. Second, More data augmentation techniques can be used to test the model, so that the model can learn more different features to improve the performance of the model.

#### 5. REFERENCES

- [1] Lin, T., Dollár, P., Girshick, R.B., He, K., Hariharan, B., & Belongie, S.J. (2017). Feature Pyramid Networks for Object Detection. 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 936-944.
- [2] Vasilopoulos E, Vosinakis G, Krommyda M, Karagiannidis L, Ouzounoglou E, Amditis A. A Comparative Study of Autonomous Object Detection Algorithms in the Maritime Environment Using a UAV Platform. *Computation*. 2022; 10(3):42. <https://doi.org/10.3390/computation10030042>
- [3] Redmon, J., Divvala, S., Girshick, R., and Farhadi, A., "You Only Look Once: Unified, Real-Time Object Detection", <i>arXiv e-prints</i>, 2015.
- [4] Redmon, J. and Farhadi, A., "YOLO9000: Better, Faster, Stronger", <i>arXiv e-prints</i>, 2016.
- [5] R. Girshick, "Fast R-CNN," 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.
- [6] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.
- [7] Girshick, R., Donahue, J., Darrell, T., and Malik, J., "Rich feature hierarchies for accurate object detection and semantic segmentation", <i>arXiv e-prints</i>, 2013.
- [8] Liu, W., "SSD: Single Shot MultiBox Detector", <i>arXiv e-prints</i>, 2015.
- [9] Ricardo, Ribeiro, Seagull dataset, <https://vislab.isr.tecnico.ulisboa.pt/seagull-dataset/>
- [10] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 770-778.
- [11] Zhen, hua, Wang, Xin, yue, Zhang, Jing, Li, Kui, feng, Luan. (2021) Marine UAV data target detection model based on YOLO Pub Date: 2021-11-24, DOI: 10.3390/su132312980.
- [12] Wang, Zirui & Yang, Jingjing & Jiang, Haonan & Fan, Xueling. (2020). CNN Training with Twenty Samples for Crack Detection via Data Augmentation. *Sensors*. 20. 4849. 10.3390/s20174849.
- [13] Loshchilov, I. and Hutter, F., "Decoupled Weight Decay Regularization", <i>arXiv e-prints</i>, 2017.
- [14] Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. *ArXiv*, abs/1804.02767.
- [15] Buslaev, A., Parinov, A., Khvedchenya, E., Iglovikov, V. I., and Kalinin, A. A., "Albumentations: fast and flexible image augmentations", <i>arXiv e-prints</i>, 2018.
- [16] Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., and Cho, K., "Augmentation for small object detection", <i>arXiv e-prints</i>, 2019.
- [17] Sergey Ioffe and Christian Szegedy. 2015. Batch normalization: accelerating deep network training by reducing internal covariate shift. In <i>Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37</i> (<i>ICML'15</i>). JMLR.org, 448-456.
- [18] Shi Junnan, Jin Jiucui, Zhang Jie. Detection and removal of reflective area in unmanned ship surveillance image and experimental verification [J]. *Marine Sciences*,2018,42(01):82-87.
- [19] Ribeiro, R., Cruz, G., Matos, J., & Bernardino, A. (2019). A Data Set for Airborne Maritime Surveillance Environments. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(9), 2720-2732. <https://doi.org/10.1109/TCSVT.2017.2775524>

- [20] Liang, Yue Xiang, Feng, Hui, Xu, Haiyang. Surface small target detection algorithm for intelligent ship. Journal of Dalian University of Technology, 1000-8608(2021)03-0255-10.
- [21] Li Yao, Research on Object Detection Algorithm Based on deep Learning -- taking blood cell detection as an example. Shanxi Normal University, 2019.
- [22] Nepal, U.; Eslamiat, H. Comparing YOLOv3, YOLOv4 and YOLOv5 for Autonomous Landing Spot Detection in Faulty UAVs. Sensors 2022, 22, 464. [https:// doi.org/10.3390/s22020464](https://doi.org/10.3390/s22020464)

Copyright@FTSM  
UKM