ACCV2022 #471

AirBirds: A Large-scale Challenging Dataset for Bird Strike Prevention in Real-world Airports

Anonymous ACCV 2022 submission

Paper ID 471

1 Comparisons

In Tab. 1, extra comparisons between AirBirds and other relevant datasets are presented. AirBirds has 3 appealing merits compared to existing ones. (i) It can be used for detection (ii) Images in our dataset are in chronological order (iii) The data volume of AirBirds is much bigger than others.

Table 1: Comparisons of AirBirds and relevant datasets. In the Task column, DES means the database mainly describes occurred strike accidents, CLS means the dataset is intended for classification, and DET means the dataset can be used for detection. Sequence means whether the images are in chronological order

Dataset	**Format**	**Images**	**Instances**	**Task**	**Sequence**	**Duration**	**Year**
FAA Database	text	-	227,005	DES	✗	30 years	1990
CUB-200-2010 [1]	image	6,033	6,033	CLS	✗	-	2010
CUB-200-2011 [2]	image	11,788	11,788	CLS	✗	-	2011
Birdsnap [3]	image	49,829	49,829	CLS	✗	-	2014
NABirds [4]	image	48,562	48,562	CLS	✗	-	2015
Wind Farm [5]	image	16,200	32,000	CLS	✓	3 days	2015
VB100 [6]	video	-	1,416	CLS	✓	-	2016
AirBirds	image	118,312	409,967	DET	✓	1 year	2021

2 Configurations

We evaluate a wide range of detectors on the created dataset for flying bird discovering. To reproduce the results, it is necessary to clarify the specific settings for each detector and we conclude them in Tab. 2.

3 Analysis

Various kinds of detectors perform consistently below expectations according to Table 2 in the main paper, even with carefully customized configurations.

Table 2: **Version** indicates the specific model. **Period** refers to the total hours of the model training. LR: initial learning rate

Model	**Version**	**Optimizer**	**LR**	**Batch Size**	**Epoch**	**Period**
FCOS[7]	ResNeXt101	SGD[8]	1.0e-2	6	50	35h
EffiDet[9]	EffiDet-D2	AdamW[10]	1.0e-3	4	20	50h
YOLOv3[11]	DarkNet53	SGD[8]	1.0e-3	2	56	60h
YOLOv5[12]	Medium-size	SGD[13]	1.0e-2	4	20	48h
Faster[14]	ResNet50+FPN	SGD[8]	1.5e-2	4	20	20h
Cascade[15]	ResNet50+FPN	SGD[8]	1.5e-2	4	20	20h
DETR[16]	Vanilla	AdamW[10]	2.0e-4	2	100	336h
Deform[17]	Vanilla	SGD[8]	2.0e-4	2	32	110h
FPN[18]	ResNet50	SGD[8]	1.5e-2	2	20	20h
NASFPN[19]	ResNet50	SGD[8]	8.0e-2	10	50	51h
RepPoints[20]	ResNet50+FPN	SGD[8]	1.0e-2	2	24	23h
CornerNet[21]	Hourglass104	Adam[13]	5.0e-4	3	42	68h
FreeAnchor[22]	ResNet101	SGD[8]	1.0e-2	3	20	18h
HRNet[23]	HRNet	SGD[8]	1.5e-2	1	24	49h
DCN[24]	ResNet50	SGD[8]	2.0e-2	2	24	27h
DCNv2[25]	ResNet50	SGD[8]	2.0e-2	2	24	31h

Therefore, a natural question is raised, what are possible reasons that degrade their performances on AirBirds in a dramatic way? We summarize three reasons that cannot be overlooked.

- First, objects with small areas in a high-resolution image are usually fuzzy in appearance and outline. For example, an object in a small area could be a bird, a kite or a drone. Furthermore, small areas are more likely to be confused with the background then discarded by detectors, regardless of what object it is.
- Second, the backbone networks used in current object detection models extract and aggregate features by a series of convolution and pooling layers with downsampling operation which even worsens the problem. For example, a backbone of 50-layer ResNet [26] will reduce an area of 8x8 pixels of a tiny bird in an input image to a 2x2 feature map only after the first convolution and pooling layer.
- Third, the predictions for small-size objects are prone to interesting with ground-truth boxes with lower IoUs than those for medium or large objects. Fig. 1 illustrates this case, there are large(100x100), medium(40x40) and small(8x8) types of objects. Green boxes are groundtruth, red boxes are predictions given by a detector and all of them offset 4 pixels from the groundtruth. As observed from left to right, their IoUs are 85.5%, 68.1% and 14.3%, respectively. It means an offset of a few pixels caused by the detector will dramatically degrade the IoU score for small objects. However, the lower

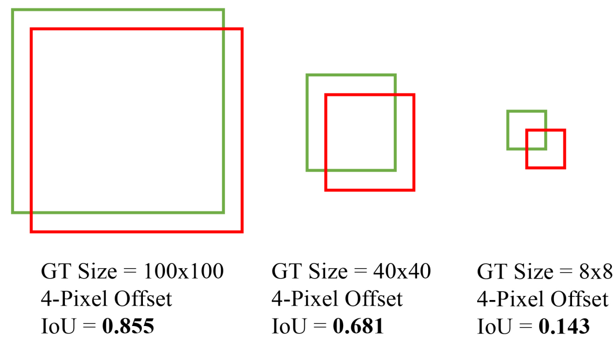


Fig. 1: Influence of the object size on IoU. Red boxes are predictions and green boxes are ground truth. Under the same offset, the IoUs between predictions and ground truth of large(100x100), medium(40x40) and small(8x8) objects are 0.855, 0.681 and 0.143, respectively

bound of IoU in current elevation metrics of COCO is 50%, increasing to 95% with a stride of 5%. It is non-trivial for the IoUs between predictions and annotations in AirBirds to touch that lower bound, thus previous decent detectors struggle.

We believe the above reasons illustrate low AP$_s$ scores for most detectors and argue that it is unfair to emphasize IoU when evaluating small object detection, specifically for flying birds discovering in real-world airports.

4 Visualization

Visualization is an essential tool to explore a dataset. In this section, we firstly investigate what the detectors learned from AirBirds by visualizing a series of feature maps. Secondly, common examples are presented to give readers an intuitive idea about the developed dataset.

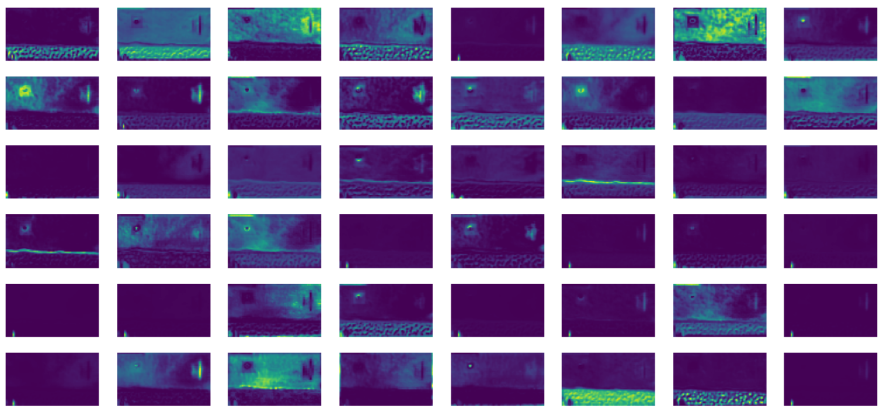
4.1 Feature Maps

In order to figure out what those models have learned from AirBirds, we take YOLOv5 as an example since it outperforms others according to Table 2 in the main paper. Multiple feature maps of the convolution layers in YOLOv5 are visualized, referring to Fig. 2c. By comparing the input image in Fig. 2a with these feature maps, a clear message is that the model has recognized a borderline that segments the sky and the ground. And small light patches in feature maps are all above that borderline and they probably convert to final predictions. For example, in Fig. 2b a bird discovered by YOLOv5 corresponds to a light patch in the feature maps of the second row.

ACCV2022 #471

(a) input image to YOLOv5

(b) prediction(s) of YOLOv5



(c) multiple feature maps of the convolution layers of YOLOv5

Fig. 2: (a) A test image for YOLOv5 (b) Prediction(s) output by YOLOv5 (c) Visualization of the feature maps of the layers in stage16 of YOLOv5

4.2 Gallery

We select some examples of AirBirds to open a window of exploring this challenging and diverse dataset intuitively. Birds in the images are highlighted by red bounding boxes, referring to Fig. 3.

In the first row, a bird flies to the left. Similarly, another bird appears at the top of the monitoring view in the second row, gliding from right to left. Both images in the first and second row are sampled from August 2021, but the difference in color tone is noticeable. It is a struggle for readers to distinguish tiny birds in the images of remaining rows while they appear. As stated in the paper, 88% of all instances are smaller than 10 pixels and the rest of 12% are mainly in the interval [10, 50). These examples provide you with an intuitive impression. Images from rows between 3 and 6 are taken in spring, summer, fall and winter, respectively, where birds appear in dark blue sky in silent spring, soar around clouds in vibrant summer, sprint towards sunset in golden autumn and fly in flocks in cold winter. The penultimate row shows complex background, where clouds move rapidly and light conditions change dramatically, making discover-

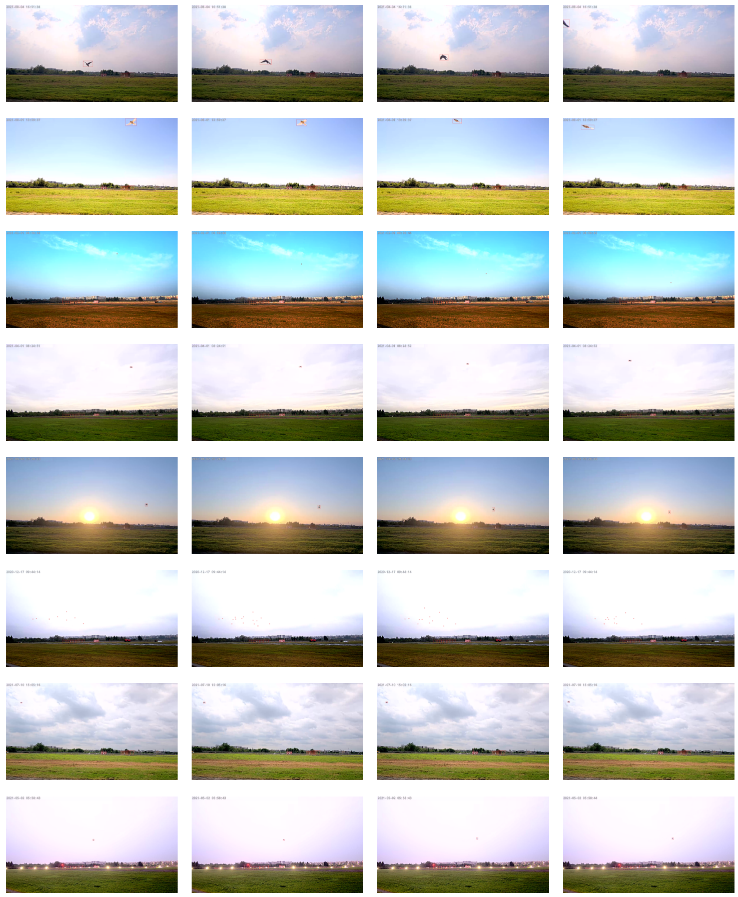


Fig. 3: Examples in AirBirds.

ing flying birds harder. Scenarios in the last row are captured near nightfall, and birds are flying under the airport lights.

In short, samples exhibited here are only a small portion of 118,312 images and we hope they open a window of exploring the scenario-diverse dataset for the research of bird strike prevention. This dataset also reveals the challenges of flying bird discovery in real-world airports and tiny object detection in other scenes, which deserve further investigation.

5 Discussion

To the best of our knowledge, AirBirds is the first large-scale challenging image dataset for bird strike prevention in real airports. Due to taking directly from real-world datasets, we should be aware of its possible limitations. AirBirds does not provide species names of birds because they are often too small to distinguish the specific species, even for domain experts. Unlike previous work CUB series [2, 1], Birdsnap [3] and NABirds [4], they are dedicated for fine-grained classification through the collection of tailor-made images. Meanwhile, we think species information can be supplemented by other sources, such as seasons, literate in the domain and existing databases. As we know, birds usually migrate with seasons and extensive literature documents desirable contents. Another typical source is FAA wildlife strike database, the bird species is one of the columns in the returned sheet after a user enters the airport name to start a query[1].

AirBirds is free of use for researchers and the license is Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0)[2].

References

1. Welinder, P., Branson, S., Mita, T., Wah, C., Schroff, F., Belongie, S., Perona, P.: Caltech-UCSD Birds 200. Technical Report CNS-TR-2010-001, California Institute of Technology (2010)
2. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology (2011)
3. Berg, T., Liu, J., Lee, S.W., Alexander, M.L., Jacobs, D.W., Belhumeur, P.N.: Birdsnap: Large-scale fine-grained visual categorization of birds. In: Proc. Conf. Computer Vision and Pattern Recognition (CVPR). (2014)
4. Van Horn, G., Branson, S., Farrell, R., Haber, S., Barry, J., Ipeirotis, P., Perona, P., Belongie, S.: Building a bird recognition app and large scale dataset with citizen scientists: The fine print in fine-grained dataset collection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2015)
5. Yoshihashi, R., Kawakami, R., Iida, M., Naemura, T.: Construction of a bird image dataset for ecological investigations. In: 2015 IEEE International Conference on Image Processing (ICIP). (2015) 4248–4252
6. Ge, Z., McCool, C., Sanderson, C., Wang, P., Liu, L., Reid, I., Corke, P.: Exploiting temporal information for DCNN-based fine-grained object classification. In: International Conference on Digital Image Computing: Techniques and Applications. (2016)
7. Tian, Z., Shen, C., Chen, H., He, T.: Fcos: Fully convolutional one-stage object detection. arXiv preprint arXiv:1904.01355 (2019)
8. Ruder, S.: An overview of gradient descent optimization algorithms. arXiv preprint arXiv:1609.04747 (2016)
9. Tan, M., Pang, R., Le, Q.V.: Efficientdet: Scalable and efficient object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). (2020)
10. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: International Conference on Learning Representations. (2019)
11. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement (2018)

[1] https://wildlife.faa.gov/search
[2] https://creativecommons.org/licenses/by-nc-sa/4.0

12. Ultralytics: Yolov5. https://github.com/ultralytics/yolov5 (2021)
13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR). (2015)
14. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **39** (2017) 1137–1149
15. Cai, Z., Vasconcelos, N.: Cascade r-cnn: Delving into high quality object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2018)
16. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In Vedaldi, A., Bischof, H., Brox, T., Frahm, J.M., eds.: Computer Vision – ECCV 2020, Cham, Springer International Publishing (2020) 213–229
17. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable {detr}: Deformable transformers for end-to-end object detection. In: International Conference on Learning Representations. (2021)
18. Lin, T.Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S.: Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017)
19. Ghiasi, G., Lin, T.Y., Le, Q.V.: Nas-fpn: Learning scalable feature pyramid architecture for object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). (2019)
20. Yang, Z., Liu, S., Hu, H., Wang, L., Lin, S.: Reppoints: Point set representation for object detection. In: The IEEE International Conference on Computer Vision (ICCV). (2019)
21. Law, H., Deng, J.: Cornernet: Detecting objects as paired keypoints. In: 15th European Conference on Computer Vision, ECCV 2018, Springer Verlag (2018) 765–781
22. Zhang, X., Wan, F., Liu, C., Ji, R., Ye, Q.: FreeAnchor: Learning to match anchors for visual object detection. In: Neural Information Processing Systems. (2019)
23. Sun, K., Xiao, B., Liu, D., Wang, J.: Deep high-resolution representation learning for human pose estimation. In: CVPR. (2019)
24. Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y.: Deformable convolutional networks. In: Proceedings of the IEEE international conference on computer vision. (2017)
25. Zhu, X., Hu, H., Lin, S., Dai, J.: Deformable convnets v2: More deformable, better results. arXiv preprint arXiv:1811.11168 (2018)
26. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2016) 770–778