BIRD IDENTIFICATION FROM MINIMAL SAMPLE

Vinod Jatav
Department of CSE
IIIT Una
Jatavvinod1997@gmail.com

Keshav Jangid
Department of CSE
JECRC Foundation
Keshav.jangid60@gmail.com

Amit Sharma
Department of CSE
Galgotias University
amitsharma641999@gmail.com

Love Kumar Yadav
Department of CSE
Galgotias University
lovekumaryadav642@gmail.com

Sridhar Swaminathan
Department of CSE
Bennett University
sridhar.swaminathan@bennett.edu.in

***Abstract*—There are thousands of categories of birds, it is very difficult to identify the birds by human and computer because different variation of birds to control and analysis. We need to identify Birds that weigh over 1.8 pounds (0.816 kg) because these birds strike can cause lots of damage to the aircraft. Bird strikes occurs mostly during LANDING, APPROACH, INITIAL ASCENT, TAKE OFF. In this research we have used Caltech-UCSD Birds-200-2011 Dataset Caltech-UCSD Birds-200-2011 (CUB-200-2011) which is the upgraded version of the CUB-200 dataset. In this dataset there are 200 bird species categories, in this research project we create handled device (android application). In this handled device we use transfer learning to train this model. We try different model to get better accuracy like VGG16, VGG19, MobileNet, ResNet50 and Inception etc.**

***Index Terms:* bird species, Caltech-UCSD Birds-200-2011, Transfer learning, VGG16, VGG19, MobileNet, ResNet, Inception.**

I. INTRODUCTION

Bird strikes by aircraft need to be identified quickly for safety of aircraft. Identification of the bird is very difficult to identify its size/weight and to determine if more testing for damage is required and what precaution to take. Currently, birds can be classified by DNA and feather classification, but the process take a lot of time (days) including delivery of samples back and forth. A faster method for bird identification and classification can be developed using deep learning so that we can identify and classify birds in very less tine (seconds).

1. The accident between bird and aircraft cost millions of dollars per year and it is significant threads to plan safety and caused hundreds of human casualties.

2. This term is also used for bird deaths resulting from various strikes with infrastructure such as power cables, tall buildings and windmills.

3. Bird strikes are a weighty hazard to aircraft safety, and have instigated a lot of accidents with human life threat. There are over 11,500 bird strikes per year.

4. The bird category "Geese" have been graded as the third most dangerous bird category. Mosthappenings occur when birds hit with the windscreen or is imbibed into the blowers of airplane.

5. These strikes caused millions of dollars sometimes whole engine needs to be change.

6. Bird identification is really a difficult task because there is lots of different categories of birds who look similar.

Therefore, we built an android application which identify a bird in a certain distance to avoid bird strike and we can find out that which category bird belong to so that next time we avoid collision.

II. Related Work

Jaderberg et al. claim that they can achieve 78.3% with the Inception-V2 architecture and Krause et al. claim that they can get 80.4% accuracy with the Inception-V3 architecture [2]. IN this research we have used "Caltech-UCSD Birds (200-2011) dataset". "Caltech-UCSD Birds (200-2011)" is an upgraded version of the CUB-200 dataset. In this dataset there are 200 Bird Species CATEGORIES. In this research project we create handled device (Android Application) to identify the images of the Birds. In this handled device we use transfer learning to train this model. We try different model to get better accuracy like VGG16, VGG19, MobileNet, ResNet50 and Inception etc. Through this model we get the 98.33% accuracy.

We basically work on different approach like: -

a) Landing
b) Initial Ascent
c) Take off of Aeroplan

We also work on classification of dataset such as it automatically detects-

a) Birds or Not Bird
b) If Bird then Check Weight; 1.8 Pounds

At airport we use these following things-

a) Speed- LR
b) Drones
c) 360* Cameras

There are number of ways to classify bird using audio data rather images. For this Purpose, Feature Extraction from audio data have some advantages like bird category have characteristic calls and no line of view is need for detection. There are some disadvantages of this method, because some birds may not produce any sound at all for a particular time and we also not able to count the number of birds precisely.

To solve the problem of bird strike we must identify the number of birds approaching to airdrome so that we can delay the flights to avoid bird strike.

IV. Methodology

The very first step for training a model is to collect data. We used “Caltech-UCSD Birds-200-2011” dataset [2] for training our model.

The Caltech-UCSD Birds-200-2011 dataset has 200 classes of birds which include 11,788 total images around 65-75 images per category of birds.

As transfer learning is the most used methodology in Deep learning. We used MobileNet V2 architecture for training our model. MobileNet V2 is most suitable for mobile and embedded devices-based vision applications where there is a lack of computational power. The MobileNet V2 is high speed, good performing and low maintenance deep learning architecture.

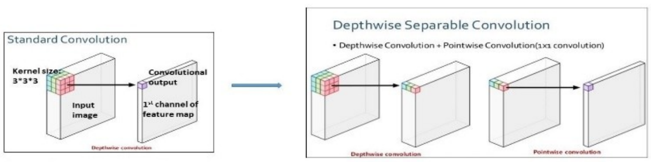


Fig. 1. Standard and Depthwise Separable Convolution [1]

stage	output	ResNet-50	**ResNeXt-50 (32×4d)**
conv1	112×112	7×7, 64, stride 2	7×7, 64, stride 2
conv2	56×56	3×3 max pool, stride 2 [1×1, 64 3×3, 64 1×1, 256] ×3	3×3 max pool, stride 2 [1×1, 128 3×3, 128, *C*=32 1×1, 256] ×3
conv3	28×28	[1×1, 128 3×3, 128 1×1, 512] ×4	[1×1, 256 3×3, 256, *C*=32 1×1, 512] ×4
conv4	14×14	[1×1, 256 3×3, 256 1×1, 1024] ×6	[1×1, 512 3×3, 512, *C*=32 1×1, 1024] ×6
conv5	7×7	[1×1, 512 3×3, 512 1×1, 2048] ×3	[1×1, 1024 3×3, 1024, *C*=32 1×1, 2048] ×3
	1×1	global average pool 1000-d fc, softmax	global average pool 1000-d fc, softmax
# params.		$\mathbf{25.5}\times 10^6$	$\mathbf{25.0}\times 10^6$
FLOPs		$\mathbf{4.1}\times 10^9$	$\mathbf{4.2}\times 10^9$

Table 1. (**Left**) ResNet-50. (**Right**) ResNeXt-50 with a 32×4d template (using the reformulation in Fig. 3(c)). Inside the brackets are the shape of a residual block, and outside the brackets is the number of stacked blocks on a stage. “*C*=32” suggests grouped convolutions [24] with 32 groups. *The numbers of parameters and FLOPs are similar between these two models.*

Fig. 2. ResNet50 Architecture [1]

Implementation of ResNet50:

Prepare the base model from the ResNet50 model developed by Google, and pre-trained on the ImageNet dataset, which is a large dataset including 1.4 million images and 1000 modules of online pictures.

First, pick middle most layer of ResNet50 will be used for extracting features. A public practice is to use the yield of the very last layer before we apply the flatten action, the so-called "bottleneck layer". The is for the following fully-connected layers will be too specific to the task the system was trained on, and thus the structures learned by these layers won't be very useful for a new training model. The ResNet50 model in-built with weights which are trained on ImageNet model. By declaring the ‘include top=False’ argument, we import a network that doesnot include the classification layers at the top, which is important for extracting features [1]

```
[ ] IMG_SHAPE = (IMAGE_SIZE, IMAGE_SIZE, 3)

    # Create the base model from the pre-trained model MobileNet V2
    base_model = tf.keras.applications.MobileNetV2(input_shape=IMG_SHAPE,
                                                   include_top=False,
                                                   weights='imagenet')
```

Fig. 3. Using ImageNet

After training our model with the help of processed dataset we saved the model to a HDF5 file, for further processing we used Tensor-Flow Lite which is an free deep learning interface for mobile device . And we converted our saved model to tflite (compressed flat buffer model) with the help of with the TensorFlow Lite Converter.

We developed an Android app using TensorFlow Lite which can classify the category of bird using camera of mobile device in Real-time. Our android application takes less than 2 seconds to detect a bird and show the result with confidence score and index of bird category.

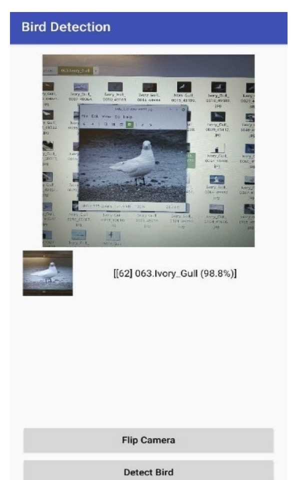


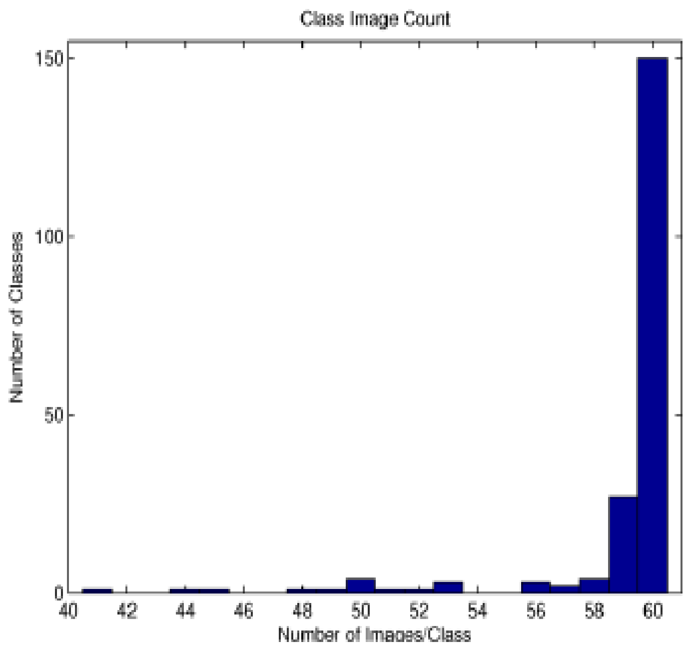
Fig. 4. Android App Demo

V. Experimental Results

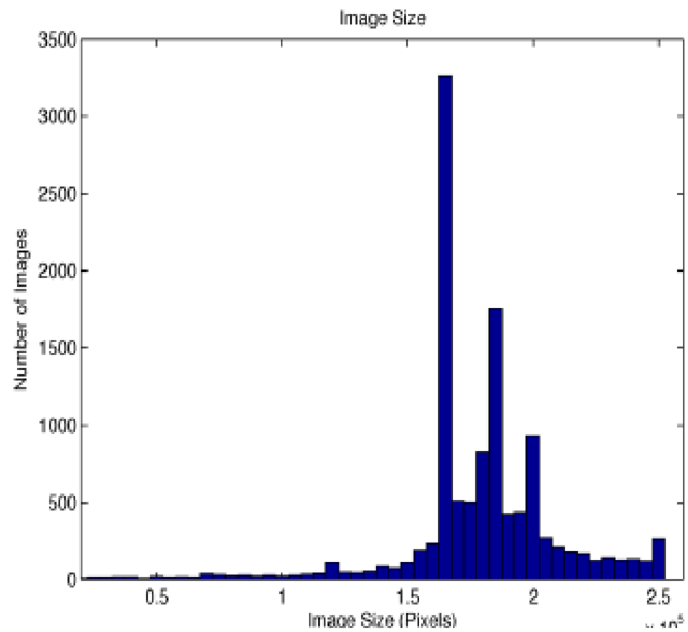
"Caltech-UCSD Birds-200-2011 (CUB-200-2011)" is an upgraded version of the "CUB-200 dataset", with nearly double the number of images per category.
"Caltech-UCSD Birds-200-2011" dataset has 200 classes of birds and 11,788 images.

Dataset Statistics:

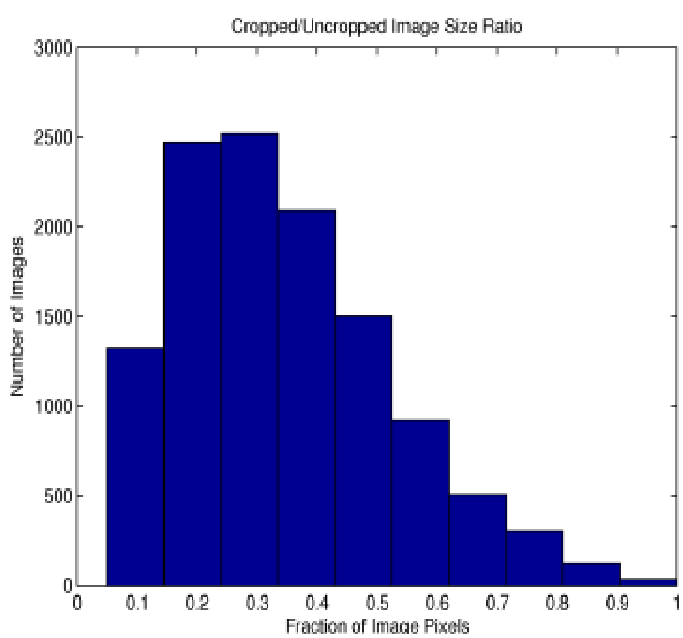
(a) Variations of the number of images per class (most modules have 50 images).
(b) Variation of the size of mostly images in pixels (most images are almost 500X500 pixels).



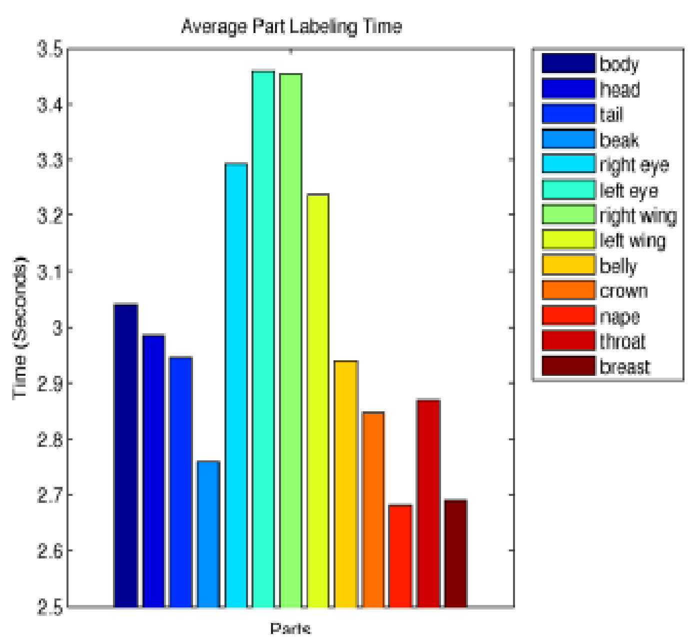
(a) Classes of images



(b) Image Resolution



(c) Cropped Image Size Distribution



(d) Labeling Time average

Fig. 5. Dataset Statistics [2]

(c) The average total of time it took by MTurkers to label each part of dataset.
(d) Variation of the ratio of the area of the bird's hover box to the area of the complete image.

In this MobileNet model we resized images to 224 by 224 and then used for training and validation.

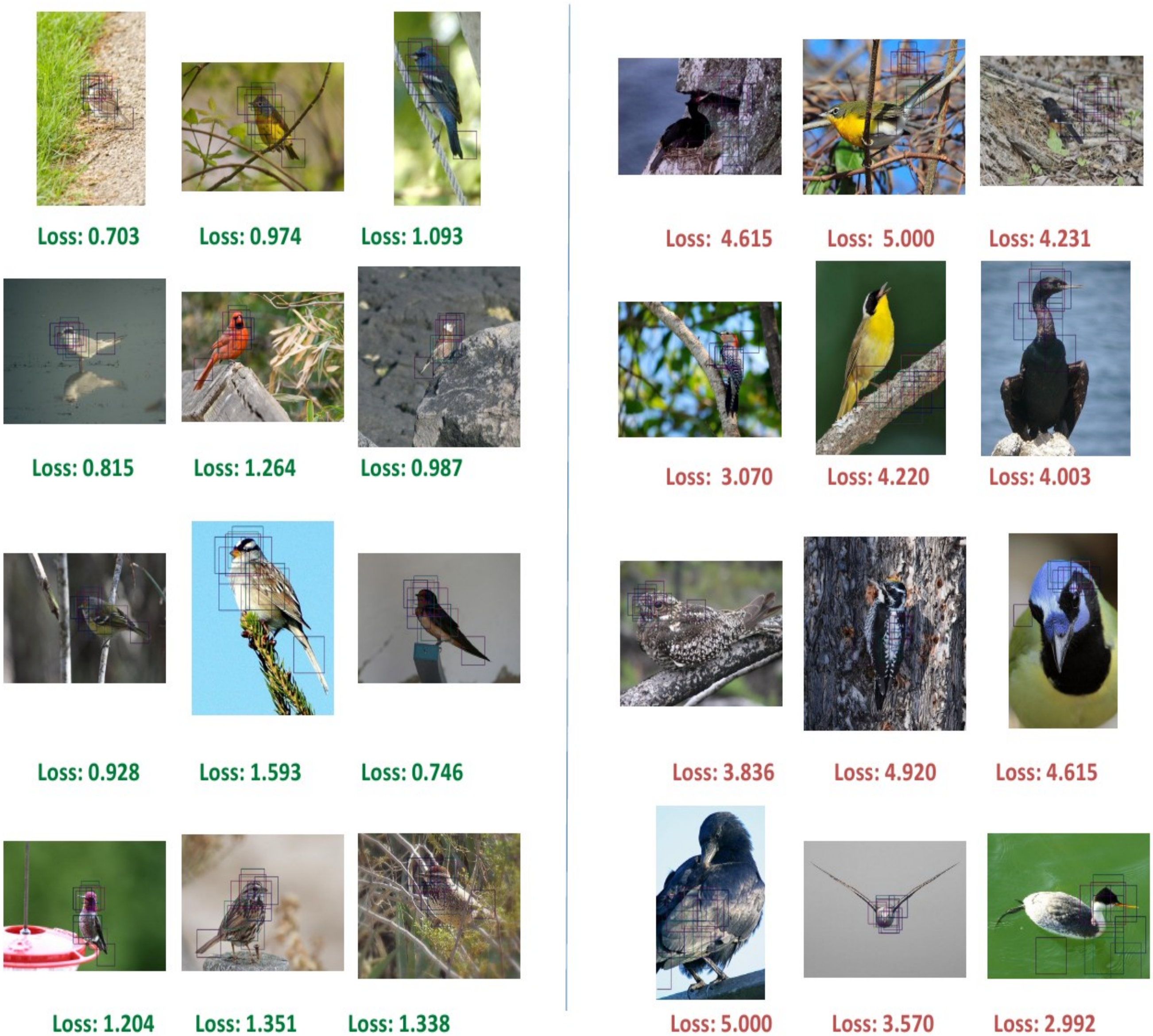


Fig. 6. Sample Part Detection Results of bird categories [3]

Sample Part Detection Results of bird categories, with good recognition samples on the left-hand side and bad recognition samples on the right-hand side which specifies that the foreseen part positions are about quite decent.

Using only the unprocessed birds' images, which allot each image to one of 200 bird species. Since the images are uncropped, we antedate that the problem cannot be resolved with high precision without gaining some grade of localization

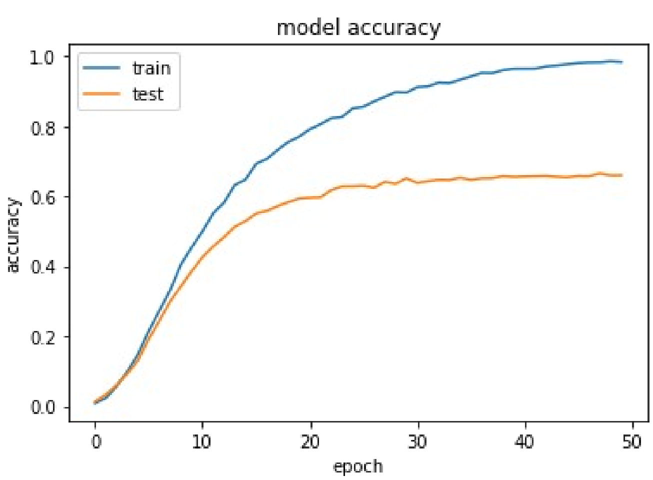
Fig. 7. 200 Categories of Birds [2]

On this dataset we use transfer learning and try with different model like VGG16, VGG19, ResNet50, and MobileNet etc.
As we do not have much data for individual species of bird so it is very difficult to get better validation accuracy after experimenting with different models, we found that with MobileNet V2 we had some better results for training accuracy and validation accuracy.

Fig. 8.Train and Validation Split [4]

Highest accuracy achieved 98.33% with ResNet50 after fine tuning our model.

Learning Curves while freezing the base layers:



Epochs

Fig. 9. Learning Curves

VI. Conclusion

Caltech-UCSD Birds-200-2011 dataset has large number of categories make it more interesting. With the help of Caltech-UCSD Birds-200-2011 we train a ResNet50 Model using transfer learning and save that model in a model.h5 file and convert it into tflite file and with the help of tflite file we develop an android application that can predict image category with the probability and with the help of this mobile application we can easily find out the category of the dead bird (bird strike with aircraft) so that we can easily get the information of the bird that which category birds are can for crash aircraft engine so that next time we can reduce the damage at the time of bird strike.

References

[1] Z. Ge, A. Bewley, C. McCool, P. Cork e, B. Upcroft, and C. Sanderson, “Fine-grained classification via mixture of deep convolutional neural networks,” in Proc. IEEE Winter Conf. Appl. Comput. Vis., 2016, pp. 1–6.

[2] Welinder P., Branson S., Mita T., Wah C., Schroff F., Belongie S., Perona, P. “Caltech-UCSD Birds 200”. California Institute of Technology. CNS-TR-2010-001. 2010

[3] T. Berg, J. Liu, S. W. Lee, M. L. Alexander, D. W. Jacobs, and P. N. Belhumeur, “Birdsnap: Large-scale fine-grained visual categorization of birds,” in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit., Jun. 2014, pp. 2019–2026.

[4] A. Marini, J. Facon, and A. L. Koerich, “Bird species classification based on color features,” in IEEE Int’l Conf Sys Man Cybernetics, Oct 2013, pp. 4336–4341.

[5] Branson S., Wah C., Babenko B., Schroff F., Welinder P., Perona P., Belongie S., “Visual Recognition with Humans in the Loop”, European Conference on Computer Vision (ECCV), Heraklion, Crete, Sept., 2010.

[6] Welinder, P., Perona, P. Online crowdsourcing: rating annotators and obtaining cost-effective labels. Workshop on Advancing Computer Vision with Humans in the Loop at CVPR. 2010.

[7] Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: ICCV. (2009)

[8] Holub, A., Perona, P., Burl, M.: Entropy-based active learning for object recognition. In: Workshop on Online Learning for Classification (OLC). (2008)

[9] P. Perona. Vision of a Visipedia. Proceedings of the IEEE, 98(8):1526 –1534, 2010

[10] M-E. Nilsback and A. Zisserman. Automated flower classification over a large number of classes. In Proceedings of the Indian Conference on Computer Vision, Graphics and Image Processing, Dec 2008