Received: 27 July 2016 | Revised: 10 July 2017 | Accepted: 21 July 2017

DOI: 10.1002/we.2135

WILEY

RESEARCH ARTICLE

Bird detection and species classification with time-lapse images around a wind farm: Dataset construction and evaluation

R. Yoshihashi | R. Kawakami | M. Iida | T. Naemura

School of Engineering, The University of Tokyo, 113C2, Bldg. 2, 7-3-1, Bunkyo-ku, Tokyo, Japan

Correspondence
R. Yoshihashi, 113C2, Bldg. 2, School of Engineering, The University of Tokyo, 7-3-1, Bunkyo-ku, Tokyo, Japan.
Email: yoshi@hc.ic.i.u-tokyo.ac.jp

Funding information
JSPS KAKENHI, Grant/Award Number: JP16K16083 and JP16J04552

Abstract

Collisions of birds, especially endangered species, with wind turbines is a major environmental concern. Automatic bird monitoring can be of aid in resolving the issue, particularly in environmental risk assessments and real-time collision avoidance. For automatic recognition of birds in images, a clean, detailed, and realistic dataset to learn features and classifiers is crucial for any machine-learning-based method. Here, we constructed a bird image dataset that is derived from the actual environment of a wind farm and that is useful for examining realistic challenges in bird recognition in practice. It consists of high-resolution images covering a wide monitoring area around a turbine. The birds captured in these images are at relatively low resolution and are hierarchically labeled by experts for fine-grained species classification. We conducted evaluations of state-of-the-art image recognition methods by using this dataset. The evaluations revealed that a deep-learning-based method and a simpler traditional learning method were almost equally successful at detection, while the former captures more generalized features. The most promising results were provided by the deep-learning-based method in classification. The best methods in our experiments recorded a 0.98 true positive rate for bird detection at a false positive rate of 0.05 and a 0.85 true positive rate for species classification at a false positive rate of 0.1.

KEYWORDS

bird conservation, bird detection, environmental assessment, image recognition, social acceptance

1 | INTRODUCTION

Wind energy has been seen as an environmentally friendly way to generate power and balance the need for protecting the environment with the demand for energy. However, as demand for wind energy grows rapidly around the world, the environmental impact of wind farms themselves has become an issue.[1-3] One of the primary concerns is the increase in bird mortality caused by collisions with blades, loss of nesting and feeding grounds, and interception on migratory routes.[3-6] Hundreds of bird fatalities have been reported annually at several sites.[6] Automatic bird detectors have thus drawn attention in the wind energy industry.[7] The primary reason for the attention is that many countries have regulations for environmental impact assessments during the establishment and operation of wind farms.[8,9] These assessments require operators to collect sufficient data on the surrounding environment and estimate ecological risks posed by the farm.[10] Bird monitoring is an expensive and laborious task when it is carried out manually.[11] Here, automation can lower the cost, enable long-term monitoring, and lead to higher accuracy and reproducibility. In addition, automatic bird detectors can work with systems that decelerate blades or sound an alarm when birds approach.[12,13] Such systems may alleviate the environmental impact, shorten the time needed for the risk assessment survey, and help to facilitate the construction of wind farms.

Performing detection and classification of birds, however, is not a trivial task for machines. Yet, image-based detection remains one of the promising approaches,[11,12,14,15] as the information provided by visual detection is rich and detailed. Image-based detection can be complementary to radar-based detection,[16-20] which is practical for nighttime monitoring.[19,20] Image recognition has flourished in the last decade, driven by the progress in machine learning and the development of larger and larger datasets for training. Datasets, ie, pairs of inputs and desirable outputs, are crucial for building machine-learning algorithms. Furthermore, having access to the same datasets allows researchers to share the same goal and compare methods in the same manner, and it has advanced the fields of handwriting recognition,[21] face and pedestrian detection,[22,23] and

generic image classification.[24,25] In addition, it has produced robust features,[22,23,26] good classifiers,[27,28] and new image structures.[24,29] The biggest advances in recent times have been the development of web-scale general image datasets with tens of millions of images[30] and deep neural networks trained on them,[25] whose strength is in adaptive learning of features and classifiers during training.[31] In analogy to this history of computer vision and machine learning, a clean, detailed, and realistic dataset is also required for automatic bird detection and classification.

This paper describes the construction of the first image dataset that is of practical value for recognition of birds around wind farms. The dataset is based on time-lapse images captured at a wind farm in Kinki, Japan. Each bird image is annotated by experts with a bounding box and a tree-structured label indicating its species, eg, "bird—hawk-black kite". The dataset contains over 60 000 annotated bounding boxes of birds and 6000 annotated bounding boxes of non-birds. It consists of 32 000 images of 5616 × 3744 resolution, and the total dataset size is over 100 gigabytes. Our dataset is unique and practical, because the birds tend to appear at low resolution within high-resolution images. Such a large resolution difference comes from the need to cover a wide field of view in order to assess the distribution of birds in a wide area and to notice their approach well ahead of time. As shown in Figure 1, the actual appearance of birds is significantly different from those in generic datasets,[32-35] on which most computer vision methods are designed and experimented.

Our experimental results on the constructed dataset reveal the accuracy, precision, and recall of the state-of-the-art computer vision methods in practical environments for wild bird monitoring, which have remained uncertain until now. We evaluated various recognition methods exploiting hand-designed image features and deep learning, as the performance of learning-based methods highly depends on the properties of the dataset, such as the image resolution, number of training samples, and visual similarity between categories. In fact, because of the large visual difference between images in generic object detection competitions[32,33] and those in our dataset, the methods need to be re-examined for a realistic wind farm setting. Although several computer vision researchers have started focusing on bird detection and classification,[34,35] they have not considered this actual situation. Our results also reveal whether a simpler learning algorithm that is easy to train on a common central processing unit (CPU) suffices, or whether a more powerful deep learning method that makes a massive number of GPU (graphics processing unit) computations is necessary. This determination is important for efficient design of a practical monitoring system. Our results show that shallow learning works as well at bird detection as state-of-the-art deep learning. However, deep learning achieves better inter-dataset generalization in detection when it is applied to data acquired at different locations. Species classification is a harder problem, and deep learning outperforms shallow learning combined with various hand-designed features with a sufficient margin. Our dataset and codes for the experiments are publicly available at http://bird.nae-lab.org/dataset.

The contribution of our study is 3-fold. First, it provides the first practical image dataset for the task of bird recognition at wind farms. Among the bird image datasets[34,35] for image recognition, ours is unique in that it is based on images taken at a wind farm, where a bird monitoring system is actually needed. Analysis of this data provides insights for wild bird recognition, ie, on the low-resolution image properties that indicate birds and the existence of hard negatives such as insects and airplanes. Second, the dataset can be used to evaluate various established image recognition methods for bird monitoring and reveal their actual performance. By applying image recognition methods to our dataset, we concretely assess their performance, which will be useful for designing actual systems. These results are valuable because the performance of bird detectors has been hard to assess due to the lack of available benchmarks. Our results indicate that a simple AdaBoost-based detector works as well as a deep-learning-based one in classifying birds and other objects in our dataset, but deep learning has an advantage in generalizability. Third, our study provides a state-of-the-art image recognition method from the research field of computer vision for bird monitoring at wind farms. Deep neural networks, especially convolutional neural networks (CNN), are the main driving force behind the recent advancements in image recognition. In our evaluation, a CNN outperformed other methods at bird species classification and showed the possibility that this task can be automatically performed, something that existing bird detectors are not capable of.[12,13]

This work is an extension of 2 previous studies of ours.[36,37] It differs from them in 4 ways: First, our latest dataset contains more birds and produces more precise labels because of the larger effort we put on manual checking. Second, we conducted additional experiments on the latest deep learning method[38] for bird detection and classification. Third, we conducted more detailed experiments to analyze the behavior of the

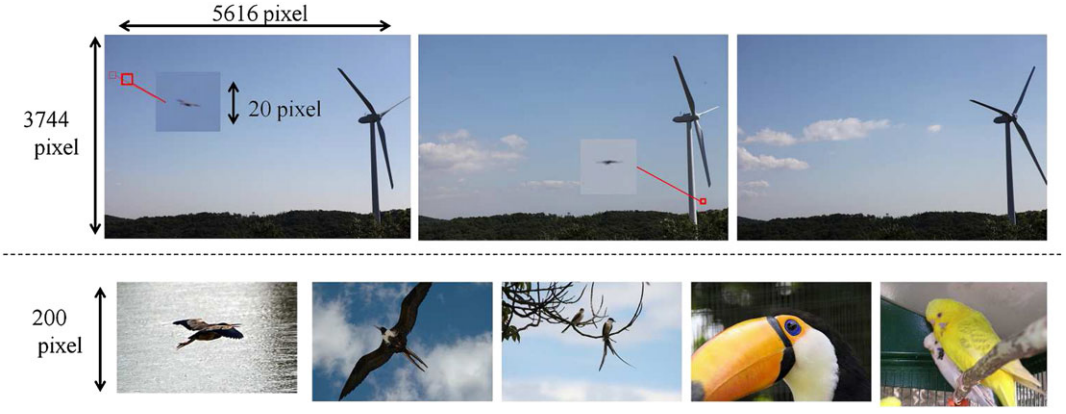

FIGURE 1 The appearances of birds in images taken around a wind farm (upper) are significantly different from those in a generic image-recognition dataset (lower)[30] [Colour figure can be viewed at wileyonlinelibrary.com]

methods in realistic settings. These experiments are designed to show the effect of unspecified bird images in the dataset and the generalizability of trained detectors to another dataset. Finally, this paper merges the content of the 2 previous papers on the dataset construction and image-recognition methodology to make it easier for readers to grasp the overall picture of our work.

2 | CONSTRUCTION OF THE DATASET

2.1 | Image capturing

We designed a camera setup especially for capturing images of birds in the distance. The setup consisted of a digital still camera (Canon EOS Mark II 5D) controlled by a laptop and equipped with a telephoto lens (Canon EF70-200 mm F4 L USM). The resolution of the sensor was 5616 × 3744 pixels, the focal length of the lens was set to 70 mm, and the field of view was 27° × 19°. In the images shot with this system, a bird with a 1-m wingspan 580 m away, ie, the distance between the cameras location and the wind turbine, would cover an area of 20 pixels. Examples of captured images are shown in Figure 1. The images include those of a 2-MW wind turbine that is 80 m in diameter and located in the Kinki region of western Japan. The images were recorded near the wind turbine for 3 days. The capture system took a picture every 2 seconds for 7 hours from 9:00 to 16:00. We obtained 10 814 images per day, 32 442 images in total. The frame rate was 0.5 fps, because of the large amount of data and slow data transfer speed. Image variances other than birds include movements of clouds, the spinning blades of the wind turbine, shaking of nearby bushes by the wind, and illumination changes. Such variances pose a challenge when we try to detect birds from image differences.

2.2 | Labeling format

Each bird in the dataset is enclosed by a bounding box labeled with its species by experts. The labeling format is similar to those of other detection datasets such as the Caltech Pedestrian Detection Benchmark,[39] which includes bounding boxes on time-series images. In addition, ours has fine-grained category annotations on each bounding box. Negative samples of other flying objects such as planes and bugs are also labeled.

For annotating the categories of bird, we designed a tree-structured list of categories so that an expert can annotate the bounding box with labels consisting of the names listed in the tree. The names of the kinds of birds in the list were selected based on the results of a preparatory field survey. The granularity of the label can be selected depending on how clear the image of the bird is. For example, when a black kite appears, we may categorize it, depending on clarity, as a black kite, a kind of hawk, as a bird, or as an unclear flying object. These options become the nodes of the tree, and the depth of the tree corresponds to the level of detail. We made the list updatable, so that when an expert finds a bird that is not listed, he or she can add it to the list.

Besides birds, other flying objects, such as airplanes, helicopters, insects, and falling leaves, are also recorded. By doing so, these objects can be distinguished from birds that might have been missed by the experts. Non-bird images can also be used as negative samples for machine learning. Objects that are too ambiguous for experts to distinguish are also recorded and labeled. Thus, the dataset contains 3 types of object: birds, non-birds, and unclear flying objects.

2.3 | Manual labeling

Manually labeling sequential images of the dataset faces several issues. First, manual labeling is time consuming, because the images number as many as 32 442. To efficiently process the data, we developed a user interface that enables us to check images sequentially and label a bird with 2 actions, ie, by making a bounding box by dragging a mouse and selecting a category from the list. The user's actual procedure is as follows: a user goes through a sequence frame by frame and checks if there is any flying object. When a flying object is found, he or she inputs the bounding box and selects the category from the given list, or else types in the category if it is not listed. The procedure is iterated until the end of the sequence.

Second, it is often difficult for non-experts to confirm an image to be of a bird, because of their small size in the images. We thus requested dozens of members from a wild bird society to inspect the images and input the data. Their efforts ensured that the labeling is precise and fine grained.

Third, due to the large size of the images (5616 by 3744 pixels), we cannot display their full size on an ordinary display. Therefore, we divided the original image into 30 (6 by 5) parts. One segment was assigned to a user, who then went through a 1-day sequence of it (a total of 10 814 images). Birds that are on the boundaries of segments may be missed easily. To prevent this from happening, we asked the users to check the images twice, and we divided the images differently in each instance. In the first check, the images were divided into 30 (6 by 5) segments. In the second check, we shifted the dividing lines by half a segment.

Fourth, manual checks may easily miss birds due to the large image size. To prevent this, the checks of each sequence were conducted by 2 different experts. We saved every bird that was annotated by at least 1 user. When the 2 experts labeled the same bird, and their overlap was larger than 25% of the smaller bounding box, we left the smaller bounding box and saved more detailed label of species. Although this process might merge different birds into one, we did not find such cases because the birds in the images were sufficiently sparse and rarely overlapped.

2.4 | Statistics

Figure 2 shows examples of birds found by the users. Some images are relatively clear, and thus, they can be specified in detail. Even some of the not-so-clear images are specified in detail. For example, the eastern marsh harriers in Figure 2 are not so clear. These birds, however, could be identified by their actions. The 3 images are a sequence of a single individual, and it kept a V-pose while flying during the sequence. This is a characteristic feature of eastern marsh harriers, and it made it possible to specify the species of the individual.

Figure 3 shows the categories and their proportions. Hawks are the most frequent, with crows being second among specified birds; 30% and 5% of the overall observations were of hawks and crows, respectively. The percentage of unspecified birds is approximately 55%. This percentage is large but reasonable because small images of birds are more frequent than larger ones, and smaller ones are often difficult to specify. Such unspecified birds are, however, still useful for distinguishing birds from bird-like patterns and for identifying target species amongst the other birds. Hence, both specified and unspecified birds were utilized in the experiments described in Section 4. Other birds include falcons, gulls, meadow buntings, sparrows, and swallows. Their numbers of appearances are small.

Figure 4 shows the size distribution of bird categories, namely undefined birds, hawks, and crows in the images. The bird species cannot be distinguished on the basis of their apparent size. The proportion of specified birds is smaller when the size is less than 15 pixels, while around a third of all found birds are specified when the size is larger than 20 pixels. Crows smaller than 25 pixels appeared less often, while smaller hawks appeared more often, seemingly because hawks are more likely to fly high.

3 | IMAGE RECOGNITION METHODS

Our algorithm is a combination of background subtraction[40] and object classification. Background subtraction is a method for extracting moving objects from fixed backgrounds and works well with our scenes that are mostly static. However, extracted regions still include some background objects, such as parts of the turbine, trees, or clouds; thus, we utilize machine-learning-based classifiers to filter birds from other objects. Specifically, we will compare 2 classifiers in the next section. The first is AdaBoost,[27] a widely used learning algorithm in computer vision. This algorithm is often combined with image features such as Haar-like[22] or Histogram of Orientated Gradients (HOG)[23] for robustness. The performance of these methods is known to depend highly on both the type of target (faces, people, birds, etc.) and scene properties (indoor, street, wind farm, etc.).

The second type is a CNN,[21] the most successful deep network for object recognition to date. The strength of a CNN is that it learns features by itself; ie, it does not need manually designed image features that are not guaranteed to be optimal. Yet, it is important to determine whether

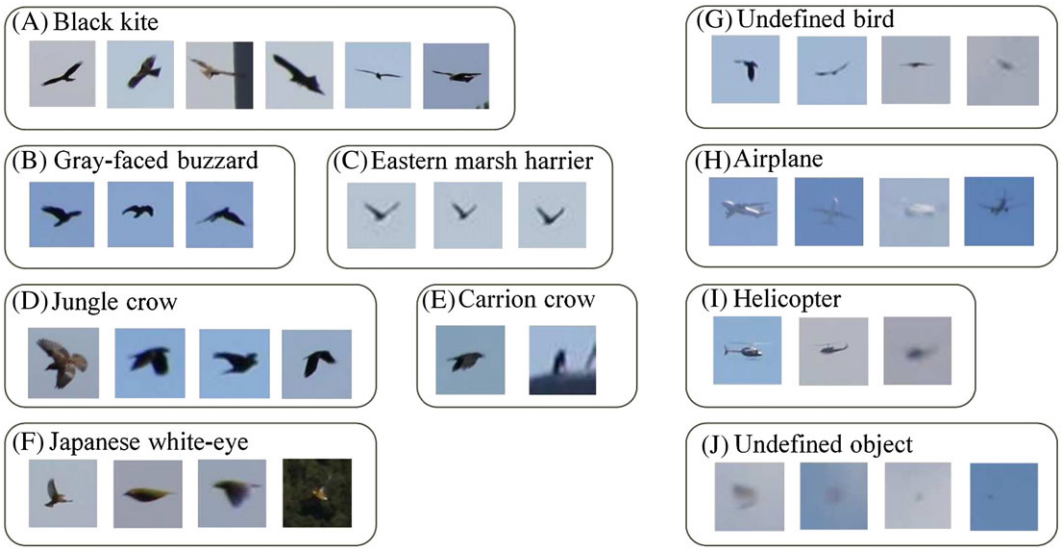

FIGURE 2 Examples of found birds and other objects. The images have been resized for visualization [Colour figure can be viewed at wileyonlinelibrary.com]

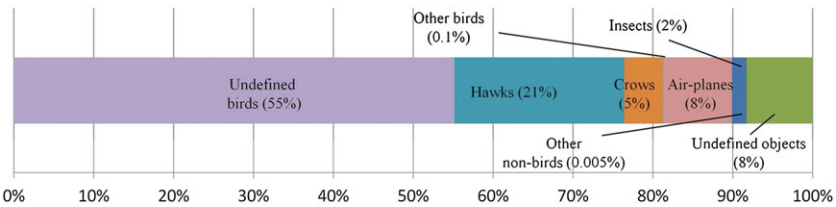

FIGURE 3 Proportions of categories of found objects. Hawks were the most frequently observed, crows second most [Colour figure can be viewed at wileyonlinelibrary.com]

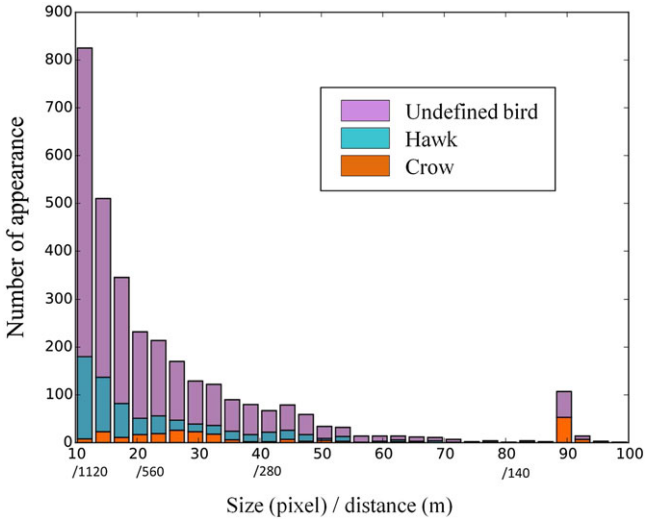

FIGURE 4 Size distribution of birds in the dataset. It also shows distances to birds from the camera corresponding to each size. Distances are calculated assuming that the bird size is 1 m [Colour figure can be viewed at wileyonlinelibrary.com]

CNNs outperform other methods in low-resolution detection and classification tasks. Because the CNN method has not been explored much, it is unclear what types of data and tasks it prefers. Below, we briefly explain the details of each method.

3.1 | AdaBoost

AdaBoost[27] is a 2-class classifier based on feature selection and weighted majority voting. A strong classifier is made from a weighted sum of many weak classifiers, and the resulting classifier is shallow but robust. The classifier is expressed as

$$y = \sum_{i=1}^{N} \alpha_i h_i(\boldsymbol{x}). \tag{1}$$

Here, $\boldsymbol{x}$ denotes an input feature vector, and y is a class score. This formulation means N weak classifiers $h_i(\boldsymbol{x}) \in \{0, 1\}$ (i = 1, 2, ..., N) vote for the output with weights α_i.. Training AdaBoost entails finding the set of weak classifiers and weights that minimize the classification error in the training samples. AdaBoost handles this optimization in a greedy manner, ie, through sequential selection of weak classifiers and weights. Given M training samples of input-output pairs and a set of weak classifier candidates H, the algorithm works as follows. First, it uniformly initializes the weights of the training samples by $D_j = \frac{1}{M}$ (j = 1, 2, ..., M), which is later updated and used to compute the weak classifiers weights. Second, it selects 1 weak classifier with the lowest error rate out of H by using the weighted training samples. The error rate of the i-th weak classifier is defined as

$$e_i = \frac{1}{M} \sum_{j=1, h_i(\mathbf{x}_j) \neq y_j}^{M} D_j. \tag{2}$$

Third, the weight of the selected weak classifier is set on the basis of the error it produces, as

$$\alpha_i = \frac{1}{2} \ln\left(\frac{1-e_i}{e_i}\right). \tag{3}$$

Here, a larger weight is set for a smaller error rate, because weak classifiers with smaller error rates are more reliable. Fourth, it updates the weights of the training samples on the basis of the error rate of the reweighted classifier by

$$D_j \leftarrow \frac{D_j \exp\left(-\alpha_j y_j h_i(\mathbf{x}_j)\right)}{Z}, \tag{4}$$

where Z is a normalization factor defined as the sum of the numerator values over j = 1,...,M. This reweighting gives larger weights to the samples misclassified by and helps to select a complementary weak classifier to the current one in the next step. After that, the algorithm repeats the steps from 2 to 4 a fixed number of times. In practice, we need to select the number of weak classifiers N to be used and a set of weak classifier candidates H. H is given as elements of the features described below, and N is a tunable parameter.

3.2 | Haar-like

Haar-like image features[22] utilize contrasts in images. It extracts the light and shade of objects by using black-and-white patterns, as shown in Figure 5A. Feature extraction using these patterns can be performed by convolution. Let us denote an input image of size $W \times H$ as I, and a pattern of size $k \times k$ as w; then, the feature extraction by convolution can be written as

$$f(x, y) = \sum_{dx \in [0,k]} \sum_{dy \in [0,k]} I(x + dx, y + dy) w(dx, dy), \tag{5}$$

The pattern resembles a 2-dimensional Haar function whose value is one in white regions and minus one in black regions. Convolution of Haar-like patterns is equivalent to subtracting the sum of the pixel intensities in the black regions of the patterns from the sum in white regions, and

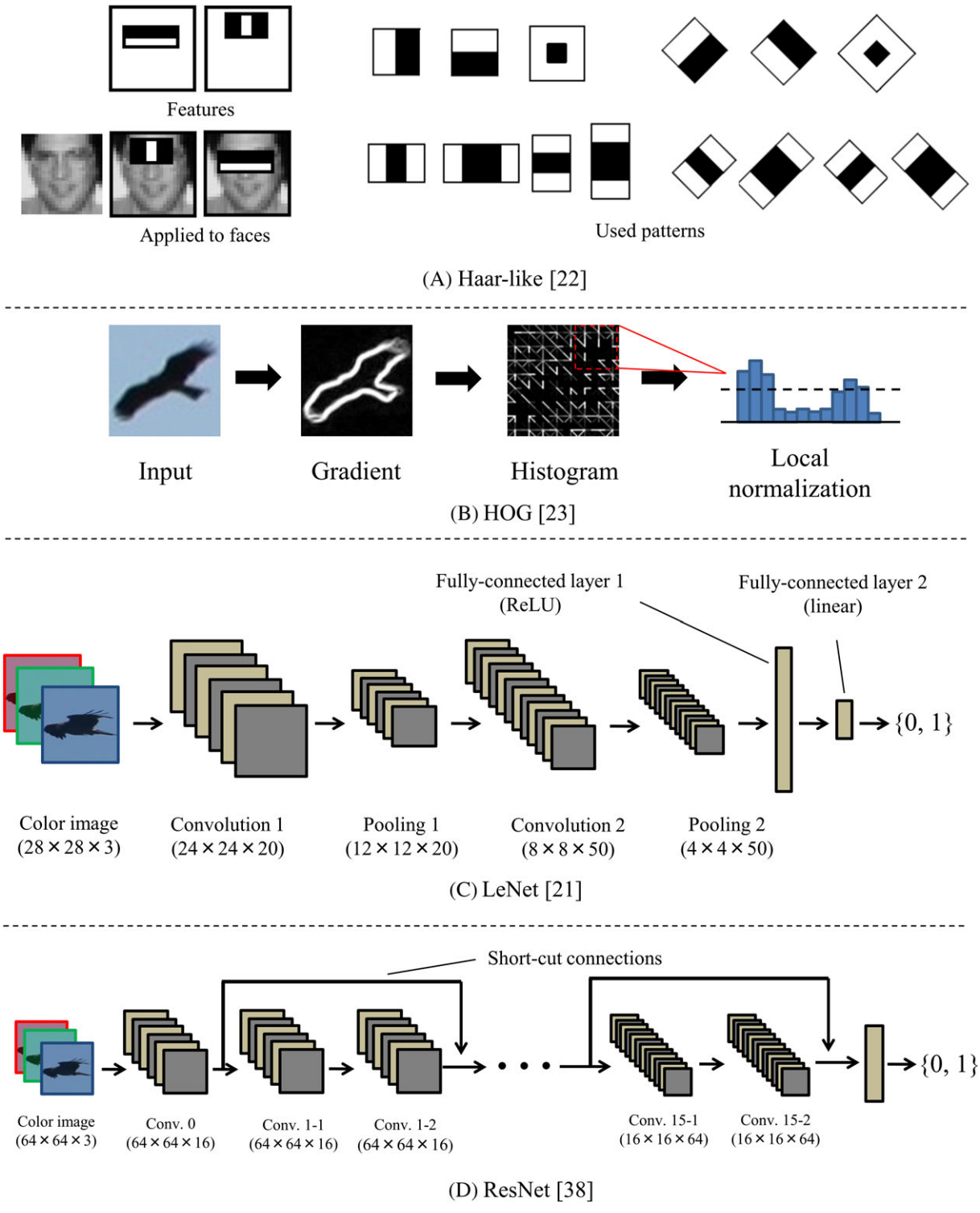

FIGURE 5 Image recognition methods we tested [Colour figure can be viewed at wileyonlinelibrary.com]

hence, the features encode contrast in images. Specifically, we adopted all of the 14 patterns used in Viola and Jones[22] (see Figure 5A). Haar-like features have been used for face detection and are considered fast and robust.[22]

3.3 | HOG

HOG[23] is a feature used for grasping the approximate shapes of objects. The method using this feature computes the spatial gradient of the image and makes a histogram of the quantized direction of the gradient in each local region, called a cell, in the image. Next, it concatenates the histograms of cells in the neighboring groups of cells, ie, blocks, and normalizes them by dividing by their Euclidean norms in each block. The pipeline of HOG feature extraction is shown in Figure 5B. HOG was first used for pedestrian detection,[23] and it has since been applied to various tasks including generic object detection.[29]

3.4 | CNN

CNN[21] is a type of neural network characterized by convolutional layers. Convolution of Equation 5 is used in a similar way as in the computation of Haar-like features. In CNN, each convolutional layer has multiple kernels and outputs multichannel feature maps. These kernels in the convolutional layers are interpreted as local connection weights between neurons, and they learn the local patterns in the images through optimization in training. Other components are pooling layers and fully connected layers. The pooling layers are placed after the convolutional layers to downsample the feature maps. These layers output lower-resolution feature maps by taking the maximum in each local region, eg, a 2 × 2 patch, of the input feature maps. The fully connected layers are placed at the end of the network. These layers act as a classifier, which receives the features from the convolutional and pooling layers and outputs the class of the input image.

Among the various CNN architectures, we examined 2 different ones, LeNet and ResNet. Our LeNet is based on a handwriting recognition method,[21] in which the CNN appeared for the first time in the literature. LeNet is easier to train than modern deeper networks for middle-scale datasets such as MNIST[21] (10 classes, 60 000 training samples) because its number of parameters is smaller than deeper networks. Our LeNet was refined by utilizing 2 recent discoveries for improving performance: rectified linear units (ReLU)[41] and dropout.[25] ReLU is a type of activation function, that is, a relationship between the input and output of a single neuron. It has a low computing cost and is easy to optimize because of its simple derivative. The relative effectiveness of ReLU among a variety of functions was discovered recently. ReLU is formulated as follows:

$$y(\boldsymbol{x}) = \max\{0, \boldsymbol{w}\boldsymbol{x} + b\}. \tag{6}$$

Here, $\boldsymbol{w}$ is a weight parameter, and b is a bias parameter. Dropout[25] is a training heuristic for removing randomly selected neurons in each iteration of the parameter update. The removed neurons are regarded to output zero independently from their inputs. The network is illustrated in Figure 5C. The network has 7 layers.

The other CNN we used is a deep residual network (ResNet). ResNets are a special type of CNN that perform better than previous CNNs on generic image-recognition tasks.[38] The largest difference between ResNets and traditional CNNs is that ResNets have shortcut connections which jump over multiple convolutional layers. These shortcut connections can make the training of deeper networks easier because they prevent the gradients of the training error from vanishing or exploding by propagating the error directly to the lower layers. Our ResNet is a modified version of the one described in Takeki et al[42] and is more suitable to small-bird detection than the original one. The network configuration is shown in Figure 5D. The network has 32 layers and is deeper than LeNet.

The training of CNNs entails computing the weights and biases that minimize the classification error rate. In practice, a surrogate function is introduced because the error rate is not differentiable and is thus hard to optimize. We used the cross entropy between ground-truth labels and the networks output for 2-class classification, which is expressed as

$$E = \sum_i^N H(y_i, \widehat{y}_i) = -\sum_i^N [y_i \ln(\widehat{y}_i) + (1-y_i)\ln(1-\widehat{y}_i)]. \tag{7}$$

Here, $y_i \in \{0,\ 1\}$ denotes the ground-truth label of the i-th training sample, and $\widehat{y}_i \in [0,\ 1]$ denotes the output of the network. Gradient methods are widely used for optimizing weights and biases. We use stochastic gradient descent.[43] This method allows us to approximately acquire the minimum at a relatively low computational cost.

4 | EXPERIMENTS

We conducted experiments involving 3 recognition tasks: bird detection, species classification, and species filtering using images taken at the wind farm. Here, we defined detection as a classification of objects into birds and non-birds, given the candidate regions suggested by the motion information. We defined classification as a classification between hawks and crows. These are the most frequent classes of birds in the area, and we had a sufficient amount of data for making an accurate evaluation of them. This 2-class classification is also practical because many endangered species

are hawks. Finally, we defined filtering as a classification between hawks and all the other birds, including other species and unspecified birds, to extract target species from the whole dataset. Our dataset revealed that a large number of birds are not specified due to their small size in the image. While classification is an idealized setting within labeled data, through filtering, we can see whether the classifiers work as well in a practical situation as in an idealized one. Figure 6 show examples of the candidate regions in each experiment. In addition to these intra-dataset experiments, ie, training and testing on our dataset, we tested the trained detector on another wild bird dataset[44] to see inter-dataset generalizability, ie, how well detectors trained on our dataset generalize to another environment.

4.1 | Experimental setup

Any machine learning method needs positive and negative samples for training. Both positive and negative samples were created by background subtraction[40] from the images in our dataset. In the detection experiment, positive samples (birds) were regions labeled as birds in the dataset. The negative samples (non-birds) were background regions, or regions not labeled as birds in the dataset. Examples of birds and non-birds are shown in Figure 6. We used 6000 positive samples and 20 000 negative samples. We used 5-fold cross-validation to conduct the experiment efficiently.

In the classification experiment, hawks labeled in the dataset were positive samples, and crows were negative samples. Classification is a more difficult task than detection on this dataset; thus, in order to analyze each methods behavior in detail, we investigated the effect of image resolution by dividing the positive and negative images into groups based on resolution. Specifically, the images of hawks and crows were divided into groups of 15 to 20, 21 to 30, and 31 to 50 pixels, as shown in Figure 6. On each group, we conducted holdout validation using 800 hawks and 150 crows for the training data and rest of each group for the test data.

In the filtering experiment, we used the set of hawks, the same data as in classification, and the set of other birds including crows, other labeled birds, and unidentified birds. The set of others consisted of 15 000 samples, and we conducted 5-fold cross-validation again. We re-evaluated the 3 best methods in the classification experiment (hawk-vs-crow), namely RGB + AdaBoost, LeNet, and ResNet.

We evaluated 2 CNNs, LeNet, and ResNet,[21] as well as AdaBoost[27] combined with 3 types of features, Haar-like,[22] HOG[23] features, and RGB (image pixel values without transformation). We quantified the detection and classification performance by using 2 measures, the true positive rate (TPR) and false positive rate (FPR). TPR is the number of true positives divided by the number of all positives in the test data. FPR is the number of false positives divided by the total number of negatives in the test data. Because there is a trade-off between TPR and FPR, the total performance of an algorithm is represented by the receiver operating characteristic curve (ROC), a curve of TPR versus FPR. A curve that goes near the upper-left-hand corner means better performance.

Finally, we applied the bird detectors (LeNet and RGB-AdaBoost) trained only on our dataset to another bird dataset from a different location[44] to clarify the limitations of the trained classifiers and to see how generalizable they are to a new environment. The dataset used in Trinh et al[44] was

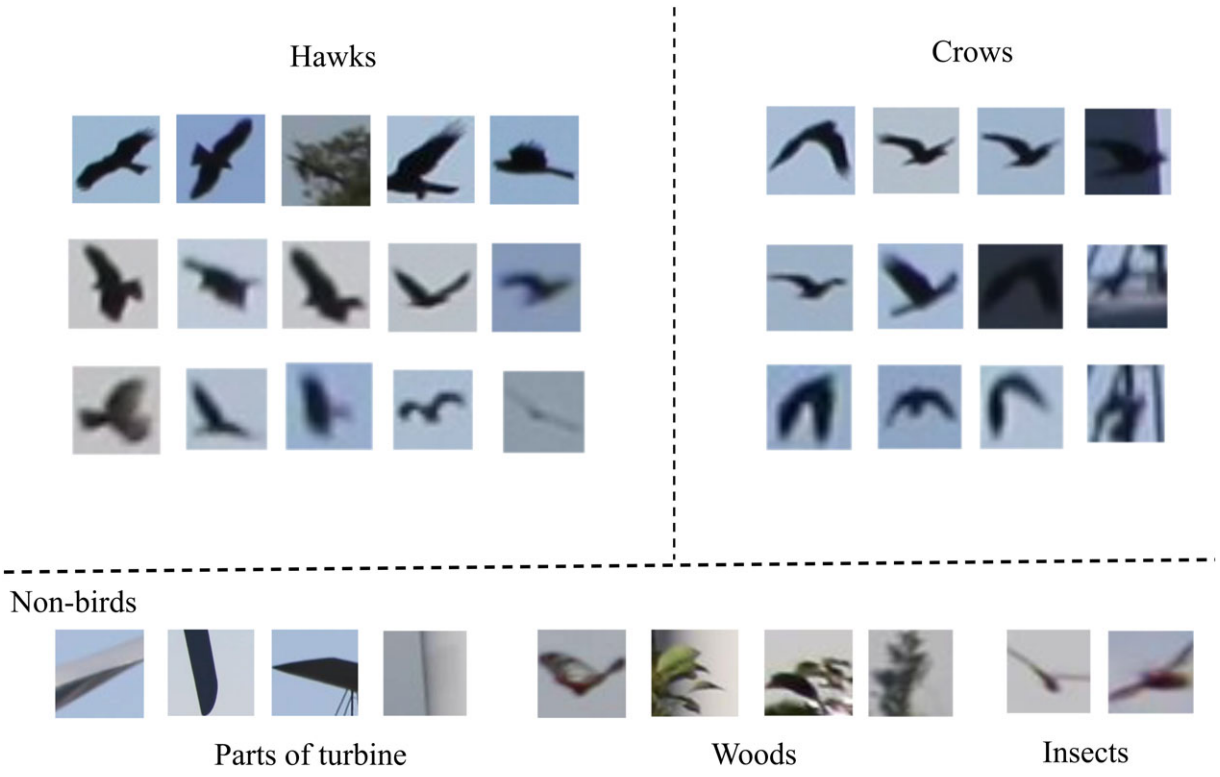

FIGURE 6 Examples of bird and non-bird images used in the evaluation [Colour figure can be viewed at wileyonlinelibrary.com]

taken from 4K-resolution, 30-fps video of a different wind farm and has annotations of birds in a similar format to ours. The major differences between this dataset and ours are in the backgrounds, eg, in the colors of the sky and clouds and in the shapes of the wind turbines.

4.2 | Implementation details

Careful parameter setting of the used classifiers is necessary for a fair comparison of the methods. For AdaBoost, we set the number of weak classifiers used with Haar-like, HOG, and RGB features to 400. We found that more weak classifiers resulted in slightly better scores with all features. For example, AdaBoost with 800 weak classifiers performed around 0.2%-point better than with 400 weak classifiers in detection, at the cost of using more memory and training time. However, it did not change the order of the score of the methods. Cascading of weak classifiers is often used to speed up AdaBoost, but we did not use it because we wanted to avoid degradation in accuracy. Thus, all of the test images were classified with all weak classifiers. The features were extracted from images resized to 24 pixels square in all the experiments. After resizing, the dimensionality of the features was 5567 in Haar-like, 1296 in HOG, and 1728 in RGB. CNNs have more parameters, ie, more layers and more and larger kernels in each layer, to specify the structure of networks. The parameters of LeNet were the same as in LeCun et al,[21] and those of ResNet were the same as in Takeki et al.[42]

5 | RESULTS

The detection results are shown in Figure 7. In the figure, FPR means the rate of misrecognizing backgrounds as birds, and TPR means the rate of correctly recognizing birds. The best performance is achieved by ResNet and RGB. Even at the FPR of 0.05, ResNet detected over 0.98 of the birds. Figure 9A shows example images that were misrecognized as birds. They are moving backgrounds such as parts of the turbine, trees blown by the wind, and flying objects such as airplanes and insects. Flying objects are more difficult negatives due to their visual similarity to birds. Note that the number of false detections depends on the number of negative samples in the data. More negative samples mean more false detections with the same FPR. Thus, the actual number of false detections may change depending on the test environment.

The results of the classification (hawk vs crow) are shown in Figure 8. Here, FPR is the rate of misrecognizing crows as hawks, and TPR is the rate of correctly recognizing hawks. This curve shows the overall performance in the resolution groups. Because of visual similarity, the species classification is more difficult than the birds-versus-others classification; thus, its performance is lower. However, among the methods, the deep learning methods showed relatively promising results for classification. For example, at the FPR of 0.1, LeNet detected 0.83 of the hawks. By contrast, when we set the TPR at as high as 0.9, LeNet misclassified 0.4 of the crows as hawks. Figure 9B shows examples of correct and incorrect classifications with LeNet in each resolution group. Sometimes, visually similar images are correctly classified, sometimes not. The CNNs do not have explicit misclassification trends because of their black-box training process.

The results of filtering (hawk vs other bird) are shown in Figure 10. In this case, ResNet slightly outperformed LeNet. For example, at the FPR of 0.1, ResNet detected 0.87 of the hawks. Similar to the results of hawk-vs-crow classification, the deep learning methods outperformed the methods based on hand-designed features. However, ResNet performed better than LeNet in filtering in contrast to classification.

The results of detection in the dataset of Trinh et al[44] are shown in Table 1. Here, we used the area under the curve (AUC), which shows the average TPR over all FPRs, to see the overall performance. While RGB's performance greatly deteriorated from 0.992 to 0.511 because of the difference between the training and testing data, LeNet showed a smaller degradation (0.991 to 0.915). The results indicate better generalizability

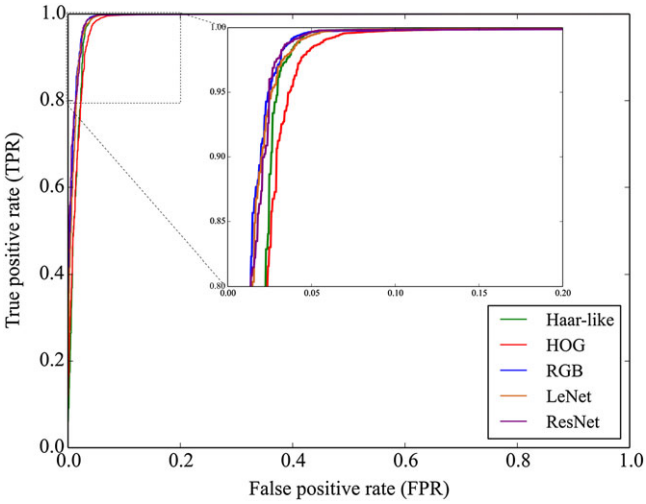

FIGURE 7 Results of detection (birds versus others). Curves that go closer to the upper left-hand corner have better performance [Colour figure can be viewed at wileyonlinelibrary.com]

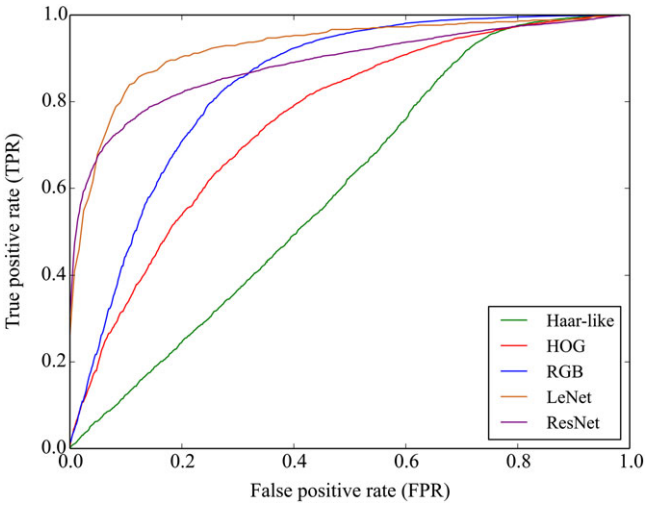

FIGURE 8 Results of classification (hawks versus crows) [Colour figure can be viewed at wileyonlinelibrary.com]

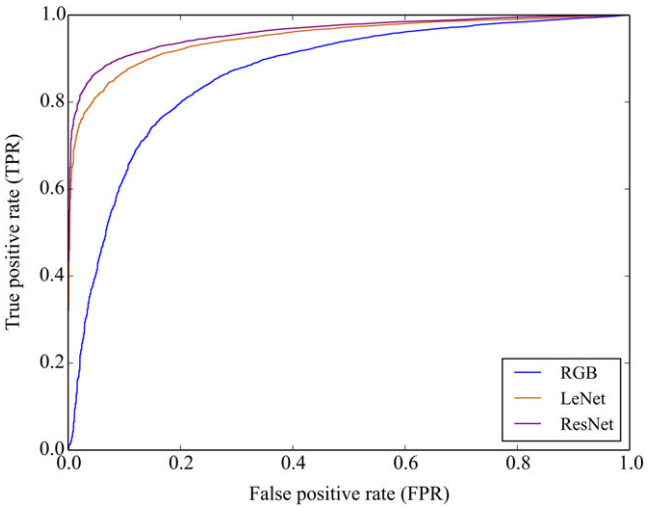

FIGURE 9 Examples of detection and classification. A, Images that were mis-detected as birds. B, Examples of correctly and incorrectly classified images [Colour figure can be viewed at wileyonlinelibrary.com]

when using LeNet than when using RGB-based AdaBoost. Examples of detection in our dataset and in that of Trinh et al[44] are shown in Figure 11. A bird was successfully detected in the dataset of Trinh et al[44] despite the large visual differences between the scenes, but there were more misdetections around the turbine and the ground than in our dataset.

6 | DISCUSSION

In the detection experiment, RGB and ResNet performed the best among the methods, and the 2 methods performed almost equally. This may have been due to the low quality of the images. The existing features are designed for detecting objects such as faces and pedestrians, which are not often at low resolution. Thus, these features are not necessarily effective in our bird detection because of the limited object resolution. For example, HOG represents details of images by gradients and is preferred in tasks like pedestrian detection and generic object detection. However, it is less robust for low-resolution bird detection.

We also note that the performance of CNNs depends on the parameters of the network and optimization. Although we used the parameters established in handwriting recognition,[21] there may be better parameters for our images. A more extensive parameter search may improve the performance of both LeNet and ResNet. Our detection results are slightly different from those on a previous version of our dataset[36] because the dataset was updated. However, the qualitative results are consistent, that is, pleasant results with simpler methods (previously Haar-like features and currently RGB) and no significant advantage of deep learning.

In the classification experiment, LeNet outperformed the other methods in all groups with different resolutions, and ResNet performed the second best. The hand-crafted features may be less effective in classification because of the subtle differences between the classes. Conversely, the learned features of the CNNs succeeded in adapting to the classification task through training. ResNet and LeNet changed places in this

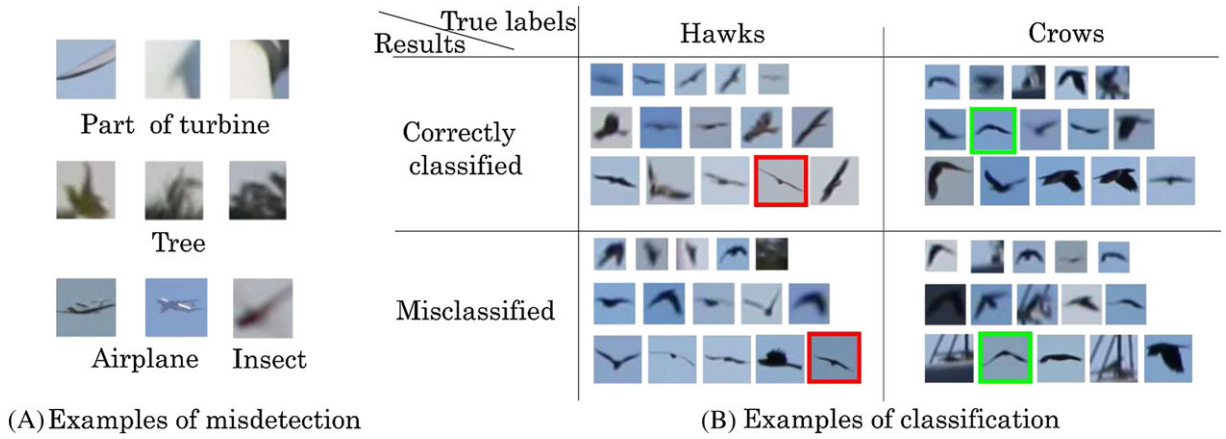

FIGURE 10 Results of filtering (hawks versus other birds) [Colour figure can be viewed at wileyonlinelibrary.com]

TABLE 1 The methods' performance on another bird dataset from a different location,[44] shown as the area under the curve. Larger values are better

	Test dataset	
Detection method	**At different location[44]**	**Ours**
RGB	0.511	**0.992**
LeNet	**0.915**	0.991

FIGURE 11 Examples of detection using images from our dataset (upper) and a dataset gathered at a different location[44] (lower) by LeNet trained on our constructed dataset. Green boxes show correct detections of birds; red ones show misdetections [Colour figure can be viewed at wileyonlinelibrary.com]

classification, although ResNet performed the best in detection. The size of the training data may have been the reason. The classification experiments were conducted with less training data than in detection, and this put deeper networks, which are more difficult to train, at a disadvantage.

The results of the filtering experiment suggest that classifiers work well even when unspecifiable birds exist in the environment. This means that our classifiers can extract a single species from all the data, and this is useful for investigation purposes. Each method performs 10% to

20% better in filtering than in classification. This seems to be because unspecifiable birds have distinguishable characteristics in themselves from specifiable hawks, and this makes filtering easier than classification. Another interesting observation is that the deeper ResNet outperformed LeNet in filtering, as opposed to the results in classification. The major barrier to utilizing deeper networks is the difficulty in training them, which especially matters with smaller datasets. The results suggest that the smaller class in the dataset may become a bottleneck to exploiting deeper networks.

7 | CONCLUSION

We constructed a bird-image dataset and evaluated typical image recognition methods for the purpose of developing an automatic bird detection and classification system for wind farms. By using our dataset from a realistic environment and representative methods in computer vision, we provided practical results and analyses of recognition performance. Interestingly, state-of-the-art deep learning did not outperform the simplest RGB features in bird detection, while deep learning was able to acquire generalizable features. The experimental results demonstrated the possibility of using image recognition for species classification. They also showed the effectiveness of using a state-of-the-art CNN for classification.

However, there is still room for improvement in the species classification problem. Finally, we would like to emphasize that our computer-vision-based bird detection system is a potential solution to the problem of bird strikes and would thereby promote the social acceptance of wind energy.

ACKNOWLEDGEMENTS

This work was part of a project sponsored by the Ministry of the Environment, Japan (MOEJ) examining measures for preventing birds, especially sea eagles, from colliding with wind turbines. This work was also supported by JSPS KAKENHI Grant Number JP16K16083 and Grant-in-Aid for JSPS Fellows JP16J04552.

ORCID

R. Yoshihashi http://orcid.org/0000-0002-1194-9663

REFERENCES

1. Snyder B, Kaiser MJ. Ecological and economic cost-benefit analysis of offshore wind energy. *Renew Energy*. 2009;34(6):1567-1578.
2. Kuvlesky WP, Brennan LA, Morrison ML, Boydston KK, Ballard BM, Bryant FC. Wind energy development and wildlife conservation: challenges and opportunities. *J Wildlife Mgt*. 2007;71(8):2487-2498.
3. Smallwood KS, Rugge L, Morrison ML. Influence of behavior on bird mortality in wind energy developments. *J Wildlife Mgt*. 2009;73(7):1082-1098.
4. Drewitt AL, Langston RHW. Assessing the impacts of wind farms on birds. *Ibis—the Int J Avian Sci*. 2006;148:29-42.
5. Drewitt AL, Langston RHW. Collision effects of wind-power generators and other obstacles on birds. *Ann N Y Acad Sci*. 2008;
6. Drewitt AL, Langston RHW. Risk evaluation for federally listed (roseate tern, piping plover) or candidate (red knot) bird species in offshore waters: a first step for managing the potential impacts of wind facility development on the Atlantic outer continental shelf. *Renew Energy*. 2011.
7. Desholm M, Fox AD, Beasley PDL, Kahlert J. Remote techniques for counting and estimating the number of bird–wind turbine collisions at sea: a review. *Ibis—the Int J Avian Sci*. 2006;148(s1):76-89.
8. Fox AD, Desholm M, Kahlert J, Christensen TK, Krag Petersen IB. Information needs to support environmental impact assessment of the effects of European marine offshore wind farms on birds. *Ibis—the Int J Avian Sci*. 2006;148(s1):129-144.
9. Bassi S, Bowen A, Fankhauser S. The case for and against onshore wind energy in the UK. Grantham Research Institute on Climate Change and Environment Policy Brief, London 2012.
10. Masden EA, Haydon DT, Fox AD, Furness RW, Bullman R, Desholm M. Barriers to movement: impacts of wind farms on migrating birds. *ICES J Mar Sci*. 2009;66:746-753.
11. Clough SC, McGovern S, Campbell D, Rehfisch MM. Aerial survey techniques for assessing offshore wind farms. International Council for the Exploration of the Sea, Conference and Meeting (CM) Documents 2012.
12. Rioperez A, de la Puente M. DTBird: a self-working system to reduce bird mortality in wind farms. European Wind Energy Association Conference 2010.
13. Wiggelinkhuizen E, Barhorst S, Rademakers L, den Boon H, Dirksen S. WT-bird: bird collision monitoring system for multi-megawatt wind turbines. European Wind Energy Association Conference 2007.
14. May R, Hamre O, Vang R, Nygård T. Evaluation of the DTBird video system at the Smola wind-power plant: detection capabilities for capturing near-turbine avian behaviour. NINA Report 910 2012.
15. Clough SC, Banks AN. A 21st century approach to aerial bird and mammal surveys at offshore wind farm sites. European Wind Energy Association Conference 2011.
16. Lack D, Varley GC. Detection of birds by radar. *Nature*. 1945;156:446.
17. Flock WL. Monitoring bird movements by radar. IEEE spectrum 1968:62–66.
18. Huansheng N, Weishi C, Xia M, Jing L. Bird-aircraft avoidance radar. *IEEE Aeros Electron Syst Mag*. 2010.
19. Buler JJ, Dawson DK. Radar analysis of fall bird migration stopover sites in the northeastern us. *The Condor*. 2014;116(3):357–370.

20. Fijn RC, Krijgsveld KL, Poot MJ, Dirksen S. Bird movements at rotor heights measured continuously with vertical radar at a Dutch offshore wind farm. *Ibis*. 2015;157(3):558-566.
21. LeCun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. *Proc IEEE*. 1998.
22. Viola P, Jones M. Rapid object detection using a boosted cascade of simple features. *Comp Vis Patt Recog*. 2001;1:I–511-1:I–518.
23. Dalal N, Triggs B. Histograms of oriented gradients for human detection. *Comp Vis Patt Recog*. 2005;1:886-893.
24. Csurka G, Dance C, Fan L, Willamowski J, Bray C. Visual categorization with bags of keypoints. Workshop on statistical learning in computer vision, European Conference on Computer Vision 2004.
25. Krizhevsky A, Sutskever I, Hinton G. Imagenet classification with deep convolutional neural networks. Annual Conference on Neural Information Processing Systems 2012.
26. Lowe D. Distinctive image features from scale invariant keypoints. *Int J Comp Vis(IJCV)*. 2004;60(2):91-110.
27. Freund Y, Schapire R. A decision-theoretic generalization of on-line learning and an application to boosting. *Computat Learn Theory*. 1995;904:23-37.
28. Corinna C, Vapnik V. Support-vector networks. *Machine Learn*. 1995;20(3):421-436.
29. Felzenszwalb P, Girshick R, McAllester D, Ramanan D. Object detection with discriminatively trained part-based models. *IEEE Trans Pattern Anal Mach Intell*. 2010.
30. Deng J, Dong W, Socher R, Li LJ, Li K, Fei-Fei L. ImageNet: a large-scale hierarchical image database. *Comp Vis Patt Recog*. 2009.248-255.
31. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature*. 2015;521:436-444.
32. Chen Q, Song Z, Dong J, Huang Z, Hua Y, Yan S. Contextualizing object detection and classification. *IEEE Trans Pattern Anal Mach Intell*. 2015.
33. Russakovsky O, Deng J, Su H, et al. ImageNet large scale visual recognition challenge, arXiv:1409.0575 2014.
34. Berg T, Belhumeur P. POOF: part-based one-vs-one features for fine-grained categorization, face verification, and attribute estimation. Computer Vision and Pattern Recognition 2013.
35. Berg T, Liu J, Lee SW, Alexander ML, Jacobs DW, Belhumeur PN. BirdSnap: large-scale fine-grained visual categorization of birds. Computer Vision and Pattern Recognition 2014.
36. Yoshihashi R, Kawakami R, Iida M, Naemura T. Construction of a bird image dataset for ecological investigation. IEEE International Conference on Image Processing 2015.
37. Yoshihashi R, Kawakami R, Iida M, Naemura T. Evaluation of bird detection using time-lapse images around a wind farm. European Wind Energy Association Conference 2015.
38. He K, Zhang X, Ren S, Sun J. Deep residual learning for image recognition. Computer Vision and Pattern Recognition 2016.
39. Dollar P, Wojek C, Schiele B, Perona P. Pedestrian detection: a benchmark. Computer Vision and Pattern Recognition 2009:304–311.
40. Massimo P. Background subtraction techniques: a review. IEEE International Conference on Systems, Man and Cybernetics 2004.
41. Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. International Conference on Artificial Intelligence and Statistics (AISTAT). 2011;15:275.
42. Takeki A, Trinh TT, Yoshihashi R, Kawakami R, Iida M, Naemura T. Detection of small birds in large images by combining a deep detector with semantic segmentation. IEEE International Conference on Image Processing 2016.
43. Bottou L. Stochastic gradient descent tricks. *Neural Netw: Tricks of the Trade*. 2012:421-436.
44. Trinh TT, Yoshihashi R, Kawakami R, Iida M, Naemura T. Bird detection near wind turbines from high-resolution video using LSTM networks. World Wind Energy Conference (WWEC), 2016.

How to cite this article: Yoshihashi R, Kawakami R, Iida M, Naemura T. Bird detection and species classification with time-lapse images around a wind farm: Dataset construction and evaluation. *Wind Energy*. 2017;1–13. https://doi.org/10.1002/we.2135