VTRC
Virginia Transportation Research Council

We Bring Innovation to Transportation

Development of a Test Method for Evaluating Staining on Structures as an Indicator of Bat Presence

http://www.virginiadot.org/vtrc/main/online_reports/pdf/20-r18.pdf

DEVIN K. HARRIS, Ph.D.
Associate Professor
Engineering Systems and Environment
University of Virginia

TIANSHU LI
Graduate Research Assistant
Engineering Systems and Environment
University of Virginia

BRIDGET M. DONALDSON
Associate Principal Research Scientist
Virginia Transportation Research Council

MOHAMAD ALIPOUR, Ph.D.
Post-Doctoral Researcher
Engineering Systems and Environment
University of Virginia

Final Report VTRC 20-R18

VIRGINIA TRANSPORTATION RESEARCH COUNCIL
530 Edgemont Road, Charlottesville, VA 22903-2454
vtrc.virginiadot.org

Standard Title Page - Report on Federally Funded Project

1. Report No.: FHWA/VTRC 20-R18	2. Government Accession No.:	3. Recipient's Catalog No.:

4. Title and Subtitle:	5. Report Date:
Development of a Test Method for Evaluating Staining on Structures as an Indicator of Bat Presence	February 2020 6. Performing Organization Code:
7. Author(s): Devin K. Harris, Tianshu Li, Bridget M. Donaldson, and Mohamad Alipour	8. Performing Organization Report No.: VTRC 20-R18
9. Performing Organization and Address: Virginia Transportation Research Council 530 Edgemont Road Charlottesville, VA 22903	10. Work Unit No. (TRAIS): 11. Contract or Grant No.: 111999
12. Sponsoring Agencies' Name and Address: Virginia Department of Transportation, 1401 E. Broad Street, Richmond, VA 23219 Federal Highway Administration, 400 North 8th Street, Room 750, Richmond, VA 23219-4825	13. Type of Report and Period Covered: Final Contract 14. Sponsoring Agency Code:

15. Supplementary Notes:
This is an SPR-B report.

16. Abstract:

To protect threatened and endangered bat species in Virginia, VDOT conducts bat inventory surveys on bridges, structures, and dwellings to identify whether bats are using the infrastructure. Observing guano droppings and staining is a common indicator of bat presence, but it can be difficult to verify whether certain stains originated from bats or other sources such as water seeps, asphalt leaching, or other structural deterioration mechanisms. While bat indicators are hard to distinguish by humans without training, from a computer vision perspective they show different features and patterns that, when coupled with expert opinion, can be used for automated detection of bat presence. To facilitate the detection of bat presence in structures maintained by VDOT and streamline bat surveys, this project leverages recent advances in visual recognition using deep learning to develop an image classification system that identifies bat indicators. To overcome the shortage of data needed to train a deep learning model, the bat identification task used the parameters previously trained on large-scale image datasets to transfer the learned feature representation. Using a pool of data collected through VDOT, a visual recognition model was developed and achieved 92.0% accuracy during testing. To facilitate the application of the developed model, a prototype web application was created to allow users to upload images and receive classification results from the developed model. The study recommends that VTRC staff: (1) work with the VDOT Information Technology Division (ITD) to host the image classification web app and make it accessible for use by VDOT bridge inspectors and environmental staff, and (2) conduct a pilot evaluation of the web app in several VDOT Districts before widespread deployment of the web app.

17 Key Words:	18. Distribution Statement:
Bat stain, bat staining, visual recognition model, deep learning	No restrictions. This document is available to the public through NTIS, Springfield, VA 22161.

19. Security Classif. (of this report):	20. Security Classif. (of this page):	21. No. of Pages:	22. Price:
Unclassified	Unclassified	22	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

FINAL REPORT

DEVELOPMENT OF A TEST METHOD FOR EVALUATING STAINING ON STRUCTURES AS AN INDICATOR OF BAT PRESENCE

Devin K. Harris, Ph.D.
Associate Professor
Engineering Systems and Environment
University of Virginia

Tianshu Li
Graduate Research Assistant
Engineering Systems and Environment
University of Virginia

Bridget M. Donaldson
Associate Principal Research Scientist
Virginia Transportation Research Council

Mohamad Alipour, Ph.D.
Post-Doctoral Researcher
Engineering Systems and Environment
University of Virginia

In Cooperation with the U.S. Department of Transportation
Federal Highway Administration

Virginia Transportation Research Council
(A partnership of the Virginia Department of Transportation
and the University of Virginia since 1948)

Charlottesville, Virginia

February 2020
VTRC 20-R18

DISCLAIMER

The project that is the subject of this report was conducted under contract with the Virginia Department of Transportation, Virginia Transportation Research Council. The contents of this report reflect only the view of the authors, who maintain responsibility for the facts and accuracy of the data presented herein. The contents of this document do not reflect the views or policies of the Virginia Department of Transportation, the Commonwealth Transportation Board, or the Federal Highway Administration. This report does not claim to constitute a standard, specification, or regulation. Use of manufacturer names, trade names, or trademarks herein are solely for identification purposes and are not to be considered as endorsements.

Each contract report is peer reviewed and accepted for publication by staff of the Virginia Transportation Research Council with expertise in related technical areas. Final editing and proofreading of the report are performed by the contractor.

Copyright © 2020 by the Commonwealth of Virginia
All rights reserved.

ABSTRACT

To protect threatened and endangered bat species in Virginia, VDOT conducts bat inventory surveys on bridges, structures, and dwellings to identify whether bats are using the infrastructure. Observing guano droppings and staining is a common indicator of bat presence, but it can be difficult to verify whether certain stains originated from bats or other sources such as water seeps, asphalt leaching, or other structural deterioration mechanisms. While bat indicators are hard to distinguish by humans without training, from a computer vision perspective they show different features and patterns that, when coupled with expert opinion, can be used for automated detection of bat presence. To facilitate the detection of bat presence in structures maintained by VDOT and streamline bat surveys, this project leverages recent advances in visual recognition using deep learning to develop an image classification system that identifies bat indicators. To overcome the shortage of data needed to train a deep learning model, the bat identification task used the parameters previously trained on large-scale image datasets to transfer the learned feature representation. Using a pool of data collected through VDOT, a visual recognition model was developed and achieved 92.0% accuracy during testing. To facilitate the application of the developed model, a prototype web application was created to allow users to upload images and receive classification results from the developed model. The study recommends that VTRC staff: (1) work with the VDOT Information Technology Division (ITD) to host the image classification web app and make it accessible for use by VDOT bridge inspectors and environmental staff, and (2) conduct a pilot evaluation of the web app in several VDOT Districts before widespread deployment of the web app.

FINAL REPORT

DEVELOPMENT OF A TEST METHOD FOR EVALUATING STAINING ON STRUCTURES AS AN INDICATOR OF BAT PRESENCE

Devin K. Harris, Ph.D.
Associate Professor
Engineering Systems and Environment
University of Virginia

Tianshu Li
Graduate Research Assistant
Engineering Systems and Environment
University of Virginia

Bridget M. Donaldson
Associate Principal Research Scientist
Virginia Transportation Research Council

Mohamad Alipour, Ph.D.
Post-Doctoral Researcher
Engineering Systems and Environment
University of Virginia

INTRODUCTION

Bats are an indispensable part of natural ecosystems as the primary predators of insects that are active at night, and a vehicle for distributing pollen and seeds that allow plants to fertilize. However, these important species confront severe threats that warrant concerns. Several bat species in Virginia are listed as endangered or threatened, including the federally endangered Indiana bat (*Myotis sodalis*), gray bat (*Myotis grisescens*), and the Virginia big-eared bat (*Corynorhinus townsendii virginianus*); the federally threatened northern long-eared bat (*Myotis septentrionalis*) (*1*); and the state-listed endangered little brown bat (*Myotis lucifugus*) and tri-colored bat (*Perimyotis subflavus*). Causes for the decline of bat populations in Virginia include loss of habitat, vandalism, human disturbance (*1*), and White-nose Syndrome (WNS). WNS is caused by the fungus *Pseudogymnoascus destructans* (*2*) and has been implicated in the mortality of more than 1,000,000 bats since 2006 (*3*). Ford et al. (*3*) found significant declines in overall summer activity between pre-WNS and post-WNS years for threatened and endangered bat species in Virginia.

Because bats often roost in large numbers, their populations are vulnerable to human activities that disturb or destroy their habitats. As the number of natural habitats declines, human-made infrastructure such as bridges and culverts become ideal alternatives for bat roosts, often offering stable microclimatic conditions and accessibility to vital resources such as water and foraging sites. Several Virginia bat species have been documented using bridges as roosting sites (*4-6*). To protect threatened and endangered bat species in Virginia, the Virginia Department of Transportation (*7*) is required to conduct bat surveys on bridges and culverts to identify whether bats are using the infrastructure prior to conducting maintenance activities (*7*).

To identify the presence of bats in highway infrastructures, a number of approaches have been proposed. One approach aims to identify bat presence through the detection of acoustic signals (*8; 9*). All bat species in Virginia emit acoustic pulses and use the echoes to learn about their surroundings. Monitoring bats through acoustic signals has been widely studied during the past decades (*10*). While commercial bat detectors are available, they require careful device set-up and experienced device operators to achieve accurate results. These types of bat detectors can be expensive, thus limiting their application and usage by infrastructure owners. Even with correctly operated bat detectors, bat identification results are still questionable due to discrepancies between different acoustic signal processing software programs (*11*).

Another approach to identifying bat presence on highway infrastructures is through visual inspection. Researchers have designed field studies with biologists and engineers to determine the extent to which bats are using highway infrastructures, their roosting preferences, and developing methods that predict bat usage (*10; 12-15*). A national survey was conducted including more than two thousand bridges to study bat usage of bridges and its extent (*10*). A set of best management practices were proposed to avoid conflicts between bats and transportation structures and preserve the threatened and endangered bats (*16*). To help ensure that U.S. Fish and Wildlife Service protocol is followed for assessing potential impacts to threatened and endangered bats from highway projects, many state Departments of Transportation have published bat inventory guidelines for bridges and buildings that specify the preferable locations for bats and features that suggest bat usage of infrastructure (*7; 17; 18*). In its Bat Inventory Guidelines, VDOT specifies that cracks in concrete, expansion joints, cave-like environments, and large rivers in wide floodplains are features favored by roosting bats. However, identifying bat presence during visual inspections requires human judgment based on proper biological training, and it can be difficult to verify whether some bat indicators originated from bats rather than other sources of staining such as water seepage or asphalt leaching.

Though bat indicators are hard to distinguish by humans without training, computer vision techniques applied to images of these bat indicators provide promise as a strategy to show different features that can be used to determine the presence of bats. Image classification is one of the fundamental tasks in the field of computational visual recognition that has established significant achievements during the past decade. Image classification techniques can be mainly categorized into two groups: machine-learning (ML) based and deep learning (DL) based. ML-based image classification models rely on carefully designed features to classify images into the desired categories. A variety of techniques have been developed to extract features used with classification models (*19; 20*). Feature engineering is a critical step in this image classification pipeline since proper features significantly increase classification accuracy. However, ML-based models were developed using relatively small datasets, and while certain features can be especially descriptive for a particular dataset, manually engineered visual features do not generalize well to different or large image sets.

DL-based image classification models build deep Convolutional Neural Networks (CNN) on a large image dataset (*21*). The convolution operation is powerful in recognizing local visual features, and the deep network architecture can integrate feature representations automatically (*22*). DL-based models have produced significant improvements in the standard ImageNet Large Scale Visual Recognition Challenge (ILSCRV) challenge (*21*). These model architectures were trained on well-known image datasets such as ImageNet (*21*), which is a dataset of millions of images in a few thousand categories of natural objects. AlexNet is an 8-layer network introduced in 2012 (*23*), which was the first CNN to win the ILSCRV by a significant improvement (over 10%) in state-of-the-art accuracy of ML-based methods (*19; 20*). VGG-net (*24*) and GoogLeNet (*25*) in 2014 presented deeper architectures with at least 20 layers and achieved accuracies over 90%. The deep residual network (ResNet) in 2015 presented very deep neural networks (up to 152 layers) outperforming humans (over 96%) (*26*).

This project develops a DL-based image classification model that identifies whether an image contains bat indicators. To alleviate the effects of the shortage of image data on bat indicators, this study builds a DL model using a transfer learning formulation. Figure 1 presents the concept of transfer learning in deep visual recognition, which builds from pre-trained neural networks on large datasets such as ImageNet (*21*) and repurposes the network for domain-specific tasks/datasets. Many recent works have demonstrated that intermediate features learned with Deep CNN pre-trained on large datasets can be transferable to other visual recognition tasks with limited training data (*27*-*29*). In addition to directly adopting pre-trained features, studies also show that manipulating or fine-tuning the pre-trained CNN features for domain-specific tasks/datasets boosts the performance even further (*30*-*33*). Similar transfer learning formulation has been applied to other domain-specific tasks including scene classification in remote sensing (*34; 35*) and disease detection from medical imagery (*36*), but to the best of our knowledge, there has not been any study identifying bat presence from a visual recognition perspective via transfer learning. In this project, we adopt the architecture and parameters from the deep networks that have been previously trained on a large-scale image dataset so that the feature representation learned by these deep networks can be transferred to the bat identification task.

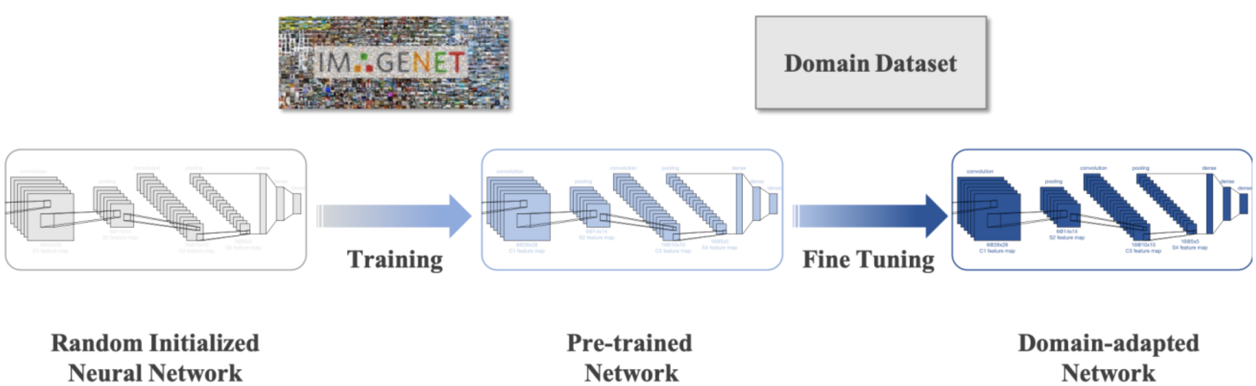


Figure 1. Transfer learning concept

PURPOSE AND SCOPE

The goal of this research was to develop a vision-based approach that leverages advances in computer vision and machine learning techniques to formulate a classification model for detecting visual indicators of bat presence on structures. By analyzing a photograph of a potential indicator, the model can be used to determine whether the indicator on structures has originated from bats or other sources with an estimation of confidence level. The research also developed a prototype web application (or web "app") that allows users to upload images and receive classification results from the developed model.

METHODS

Data Collection and Annotation

To construct the automatic bat identification model based on visual features, numerous images were collected and annotated with human expert knowledge. Photographs that included bat indicators (e.g., staining from bat guano and/or urine) were considered positive images and those with staining from other sources were considered negative images. To this end, a dataset including both positive and negative images was created. Researchers collected 324 photographs for the positive image dataset. These photographs were acquired by first contacting eight bat listservs specific to bat research and/or conservation in the U.S (*37*). Emails sent to the listservs described the purpose of this study and requested photographs of bat stains on bridges or culverts. In many instances, replies from listserv members included contact information for other experts in the field. These experts were subsequently contacted. Photographs were received from biologists and environmental scientists from consulting firms, transportation departments, natural science museums, state fish and wildlife departments, universities, and wildlife organizations, all of whom were experienced with identifying bats in culverts and bridge structures. These images were obtained during their past field studies and were confirmed to contain bat indicators. Figure 2 provides a series of representative positive images collected for this study.

Staining (black) Staining along the joint Staining (black & brown)

Droppings Bats & staining (*38*) Bats & staining (brown & white)

Figure 2. Examples of images containing bat indicators (positive images)

These positive images include bat indicators with a variety of characteristics and features:

1) The types of collected bat indicators include staining (white, brown or black), guano, and the actual presence of bats. Some images contain more than one type of bat indicator (as shown in Figure 2. c, e, f).
2) The scale of these images include detail level (where the image was mostly focused on the bat indicator), structural member level (where part(s) of structural member(s) can be seen from the image), and structure level (where more than one complete structure member can be identified from the image).
3) The lighting conditions can be categorized mainly as either daylight (314 images) or dark (10 images). Although most of the images were taken during daytime, some under-structure locations with various availability of natural light also caused the images to have different lightings.
4) The size (e.g. megabytes) of the images also varied.

Considering these variations in the collected images, having a dataset that covers a variety of targeted features ensures model applicability; however, it creates a more complicated task to train the model. This image balance is a common trade-off for all image recognition tasks, but having enough training examples from each characteristic helps overcome this trade-off.

The negative image set was designed to include visually similar and confusing features commonly present in bridge sites. To this end, 214 images were obtained from bridge inspection reports. These inspection reports were collected and maintained by VDOT on a biennial basis and contain both textual descriptions of bridge conditions as well as images taken during inspections to illustrate the extent of damages. Some of the damages have similar visual features to the bat indicators and can be confusing to the inspector when determining bat presence. Such images were collected as the negative image set. Figure 3 provides a series of representative negative images collected for this study.

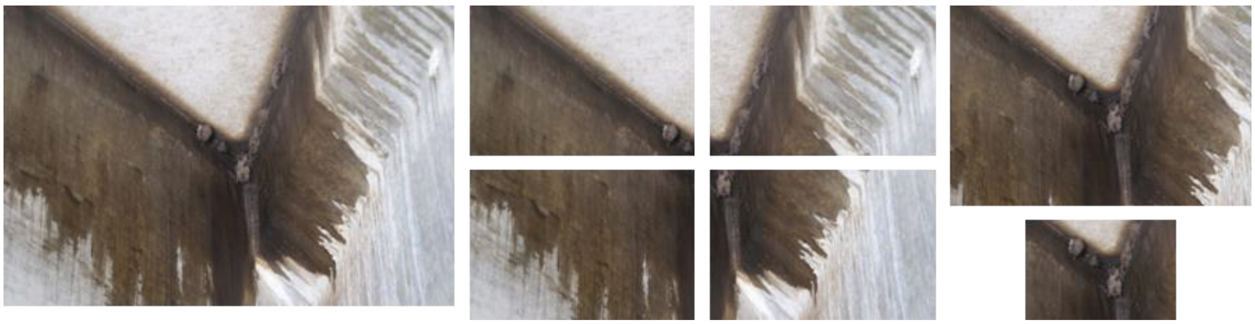
Figure 3. Examples of images containing visually similar and confusing features (negative images)

Before selecting negative images from bridge inspection reports, the collected positive images were characterized by scenes such as whether the image was focused on a specific location of the structure (e.g., expansion joints, rusted joints, the bottom of beams, abutment walls) or is a general top view or side view, and whether the images were taken during daylight or in the dark. Such characterization reveals a rough scene distribution of the collected positive images so that the same distribution could be maintained while collecting the negative images. This additional caution while collecting the negative images from bridge inspection reports aimed to minimize the impact of scene characteristic differences and to ensure that the positive set and the negative set differ primarily in the existence of bat indicators.

Data Augmentation

The 538 collected images were randomly split into training, validation, and testing sets with a ratio of 8:1:1. The training set was used for model training iterations, while the validation set was used to evaluate the performance of the model after each training iteration. The evaluation of the validation set guides the decisions on model development, such as hyper-parameter settings. The testing set was held unseen during model development for an objective evaluation of the model performance once finalized.

As the size of the collected image set was relatively small for training a deep neural network, the training and validation set was expanded and diversified through a series of alterations and transformations, usually referred to as data augmentation (*23*). The testing set remains unchanged for the purpose of subjective evaluation. First, to include a variety of scales, each collected image was processed by quadrant cut and center-crops. As shown in Figure 4, the original image, as well as the four quadrants and the two center-crops (1/3 and 2/3 width) were used for training. The quadrants and center-crops were manually examined to remove those without a bat indicator. To further enhance the variability of the collected bat image dataset, the images were resized to a fixed resolution of 256 × 256 and random 224 ×224 patches were extracted from the resized images. The patches were also augmented via random transforms including horizontal or vertical flip (p = 0.5) as well as random rotation (maximum 120 degrees). Following data augmentation, a total of 3,238 images were available for model construction.



a) Original positive image; **b)** Four quadrant crops; **c)** 2/3 width center crop (*3*), 1/3 width center crop (bottom);

d) Original negative image; **e)** Four quadrant crops; **f)** Center crops: 2/3 width (*3*) and 1/3 width (bottom)

Figure 4. Data augmentation for example positive and negative image

Model Construction

As presented in Figure 5, a typical deep learning image classification model contains a convolutional base that extracts various levels of features from images, and a classifier that classifies the image based on the extracted features (*22*). An efficient method to create a deep learning model, especially for smaller datasets, is to use Transfer Learning (*30*), which builds upon state-of-art image classification models and repurposes a pre-trained model for a new task. The pre-trained models were trained using a large benchmark dataset for image classification tasks (*21*)) and have learned to extract valuable features from general images. Since the available dataset for this bat identification task is relatively small, adopting a transfer learning setting makes use of the previously learned knowledge and achieves better performance with limited data. Models with different architectures have been developed for the benchmark ImageNet classification task (*21*). These include a number of deeper and more complex models that achieve better performance, but contain more parameters and require more computational resources in both training and testing. Transferring complex models to small datasets requires extra caution to avoid overfitting, where the excessive model parameters were all optimized for the training data and do not generalize to other use cases.

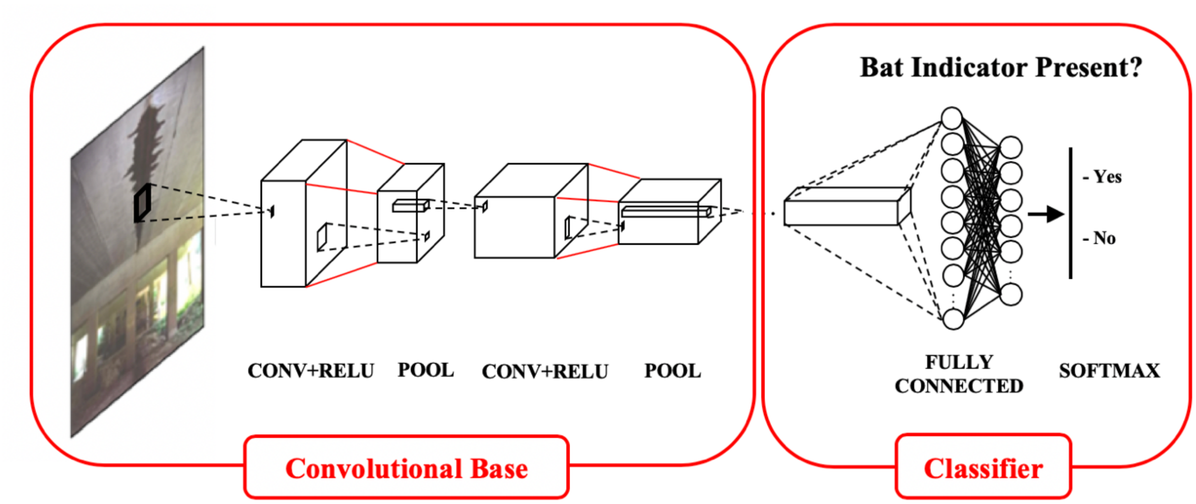


Figure 5. Architecture of a typical convolutional neural network

To evaluate how well the pre-trained models adapt to the bat indicator image set, this study first selected a few well-adopted pre-trained models, replaced the classifiers and trained the classifier using the bat indicator image set for a few epochs. During this initial training, only parameters from the classifier were optimized. The selected models include VGG-16 (*24*), Resnet (*26*) with various depths, DenseNet-161(*39*) and GoogLeNet (*25*). As previously noted, these models represent some of the most well-recognized studies of visual recognition models available in the literature. Table 1 presents a comparison of the selected models regarding size, depth, number of parameters, and performance on benchmark task, as well as performance on the bat indicator dataset after initial adoption (averaged over three runs).

In evaluating the selected models, the overarching goal was to determine suitability of these well-established models with respect to the dataset specific to this study, but also to determine the potential suitability for model extension through the proposed transfer learning framework. GoogLeNet was crafted to save computational resources, but the performance on bat identification was sacrificed severely as presented in Table 1. VGG-16 model is the architecture that achieves lower accuracy on bat identification with its considerably larger number of parameters compared to ResNet and DenseNet. Resnet-152 is much larger compared to DenseNet-161 with only a slight improvement in bat identification. Therefore GoogLeNet, VGG-16, and Resnet-152 were excluded from further parameter optimization. Resnet-18, Resnet-50, and DenseNet-161 were selected for further model tuning because it is good practice to experiment with both small and large networks to avoid overfitting a small dataset with too complex models.

Table 1. Pre-trained model characteristics and initial bat identification performance (Models selected for further parametric study are shaded grey)

Model	Size (MB)	Depth	Number of Parameters (M)	Top-1 Accuracy ImageNet	Top-5 Accuracy ImageNet	Bat Identification Accuracy	
						Mean	STD
GoogLeNet	49.7	22	6.6	69.78%	89.53%	66.00%	1.63%
VGG-16	528	23	138.3	71.59%	90.38%	72.00%	1.63%
Resnet-18	44.7	18	11.7	69.76%	89.08%	75.33%	1.89%
Resnet-50	97.8	50	25.6	76.15%	92.87%	76.67%	0.94%
Resnet-152	230	152	60.2	78.31%	94.06%	84.67%	0.94%
DenseNet-161	110	161	28.7	77.65%	93.80%	84.00%	3.27%

The three selected pre-trained models (Resnet-18, Resnet-50, and DenseNet-161) were trained using the same process as proposed in previous research related to model development (*28; 40*). First, the classifier layer from the pre-trained model was replaced with random initialized parameters to accommodate the new binary bat identification categories. The parameters from the new classifier were then trained for a few iterations while the parameters from the convolutional base remained unchanged. Then, some layers (or all) from the convolutional base were unfrozen and trained together with the classifier. This fine-tuning step uses a smaller optimization step size (termed as learning rate) to avoid sudden severe deviation from the pre-trained model. The models were optimized using the Adam (*41*) optimizer that minimizes the cross-entropy loss by iteratively updating model parameters. A weight decay factor was used to encourage small parameters to prevent overfitting (*42*). Figure 6 provides an illustration of the transfer learning pipeline for developing the proposed bat identification model.

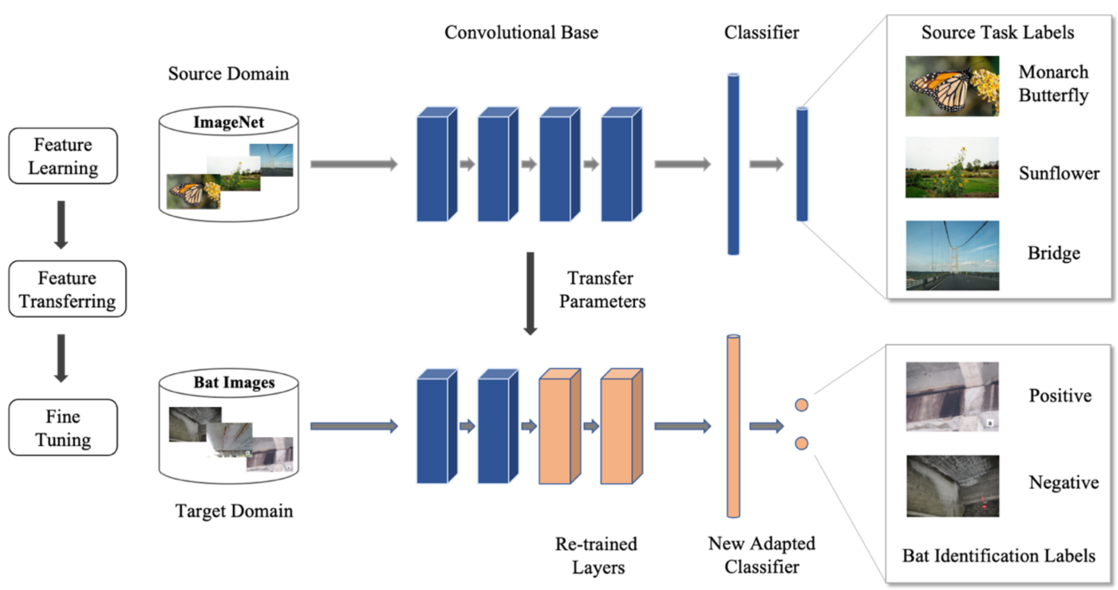


Figure 6. Transfer learning: adapt new classifier and fine tune part of the pre-trained model

Implementation and training of the models were carried out in the PyTorch environment. PyTorch is a Python package with tensor computations with integrated GPU acceleration, and deep neural networks based on automatic differentiation (*43*). The training process was deployed on NVIDIA Tesla P100 GPU nodes with 80GB of RAM.

RESULTS AND DISCUSSION

Parametric Study

A parametric study was designed to evaluate the sensitivity of the model's performance to a number of hyper-parameters. These hyper-parameters included the initial training learning rate (Lr_1), fine-tuning learning rate (Lr_2), weight decay factor, and number of iterations for initial training. Model performances were compared in terms of training and validation accuracy. A larger gap between the training and validation accuracies may indicate overfitting on the training set and weaker generalization on other images.

Table 2 presents the results of the parametric study for Resnet-18. The study experimented with initial learning rate 0.001 (*41*) and 0.0001; fine-tuning learning rates were selected to be much smaller than the initial learning rates to avoid sudden, large deviation from the pre-trained model. Initial trial and error revealed that the model reaches a plateau sooner when only training the parameters from the classifier layer, therefore a relatively small number of iterations (5 and 8) were selected for initial iterations before unfreezing all layers and fine-tuning. In Table 2, the reported performance for each parameter was the average over all other parameter variations. As shown in Table 2, the Resnet-18 model was not sensitive to the weight decay factor and number of iterations for initial training. Furthermore, a larger learning rate in both the initial training and fine-tuning led to slightly better accuracy.

Table 2. Parametric study for Resnet-18

Parameter Name	Initial Learning Rate (Lr_1)		Fine-tuning Learning Rate (Lr_2)		Weight Decay		Initial Iterations	
Parameter Value	0.0001	0.001	1.00E-05	1.00E-04	0.0001	0.01	5	8
Training Accuracy	90.83	92.64	93.41	**98.12**[1]	91.72	91.74	91.70	91.77
Validation Accuracy	85.03	86.67	88.70	**89.03**[1]	85.80	85.90	85.97	85.73

[1] Highest training accuracy displayed in bold.

Table 3 presents the results of the parametric study for Resnet-50. The initial iterations were selected to be larger than those used in the Resnet-18 model considering that the classifier in Resnet-50 is much larger in size than Resnet-18 (512 neurons) thus requiring more iterations to converge. Considering Resnet-50's depth, fine-tuning of Resnet-50 included two settings: retraining all layers and retraining only the last convolutional block (four in total) to explore the effect of retraining depth. As presented in Table 3, the Resnet-50 model was not significantly sensitive to the initial learning rate (Lr_1), the number of initial iterations and the weight decay factor. Retraining all layers performed better than retraining only the last convolutional block without causing more gap between the training and validation accuracies.

Table 3. Parametric study for Resnet-50

Parameter Name	Initial Learning Rate (Lr_1)		Fine-tuning Learning Rate (Lr_2)		Weight Decay		Initial Iterations		Re-trained Layers	
Parameter Value	0.0001	0.001	1.00E-06	1.00E-05	0.0001	0.01	5	10	all	last
Training Accuracy	93.05	93.10	90.10	**96.05** [1]	93.20	92.95	93.50	92.64	93.89	91.53
Validation Accuracy	88.72	88.78	87.82	**89.78** [1]	88.75	88.86	88.98	88.63	89.73	87.74

[1] Highest training accuracy displayed in bold.

Table 4 presents the results of the parametric study for DenseNet-161. Considering the large depth of DenseNet-161, three settings were included for re-trained layers including all layers (four dense blocks in total), the last two dense blocks and only the last dense block (*39*). As shown in Table 4, a larger learning rate in both initial training and fine-tuning led to better accuracy. Training the model with 5 initial iterations instead of 10 slightly improved the model performance. Fine-tuning more layers improved the validation accuracy and also reduced the gap between training and validation accuracies. A better parameter setting for fine-tuning DenseNet-161 was using the larger Lr_1 and Lr_2, training for 5 initial iterations and then unfreezing all layers for fine-tuning.

Table 4. Parametric study for DenseNet-161

Parameter Name	Initial Learning Rate (Lr_1)		Fine-tuning Learning Rate (Lr_2)		Initial Iterations		Re-trained Layers		
Parameter Value	0.0001	0.001	1.00E-06	1.00E-05	5	10	all	last two	last
Training Accuracy	92.28	93.99	90.03	**96.24** [1]	93.28	92.99	94.37	93.33	91.70
Validation Accuracy	88.67	89.59	87.15	91.11	89.37	88.89	**91.37** [1]	89.35	86.67

[1] Highest training accuracy displayed in bold.

As revealed in the parametric study, having a larger setting for both initial learning rate (Lr_1) and fine-tuning learning rate (Lr_2) improved model performance by different scales. Reducing the number of initial training iterations did not hurt the model performance, and helped avoid overfitting the classifier layer for DenseNet-161. Different weight decay factors did not have a significant impact. Retraining more convolutional layers improved validation accuracy without significant increases in the gap between training and validation accuracy.

Model Performance Analysis

Based on the results of the parametric study, the DenseNet-161 model achieved the highest accuracy, and was therefore selected as the final model. This model was obtained by replacing the classifier of pre-trained DenseNet-161, training initially for 5 iterations with a learning rate of 0.001, and then releasing all layers for fine-tuning with a learning rate of 1e-05. The final model was then evaluated using the independent testing set, which was not used during training and validation. Following the application of the model on testing data, the predictions of the model were compared with the ground truth and the results form a confusion matrix as shown in Table 5. Using this confusion matrix, recall (*38*), precision (PRE), and F1 score are calculated using equations 1-3. In defining these criteria, bat indicators and negative images were referred to as positive (+) and negative (-) instances respectively, as shown in Table 5.

Table 5. Confusion matrix for the classification task

Evaluation		**Ground Truth**	
		Bat Indicator (+)	Non-bat Scene (-)
Predicted	Bat Indicator (+)	100%	16%
	Non-bat Scene (-)	0	92%

$$REC = \frac{TP}{TP + FN} \tag{1}$$

$$PRE = \frac{TP}{TP + FP} \tag{2}$$

$$F1 = \frac{2 \times PRE \times REC}{(PRE + REC)} \tag{3}$$

High precision indicates low false positive, high recall indicates low false negative, and the F-1 score evaluates the balance between precision and recall. The final model achieved a testing performance of 86.21%, 100%, and 92.59% for precision, recall and F-1 score, respectively. The overall testing accuracy was 92.0%, which indicates that the overall success of the model in detecting bat indicators. These results also highlight that while the model does not miss bat indicators (100% recall), there is room for improvement with regard to its performance in distinguishing between bat indicators and visually similar distractors (86.21% recall). Expansion of the training set through the collection of more data from both the positive and negative classes is required to improve this performance. The F-1 score also shows a satisfactory overall performance and confirms that the model does generalize well to similar images on which it has not been trained.

Web Application Development

To realize the practical benefits and facilitate the utilization of the proposed approach, an interactive prototype web application interface was developed that allows users to access the web page and upload an image, which is then passed through the developed bat identification model. Figure 7 provides an illustration of the basic structure of the web app developed.

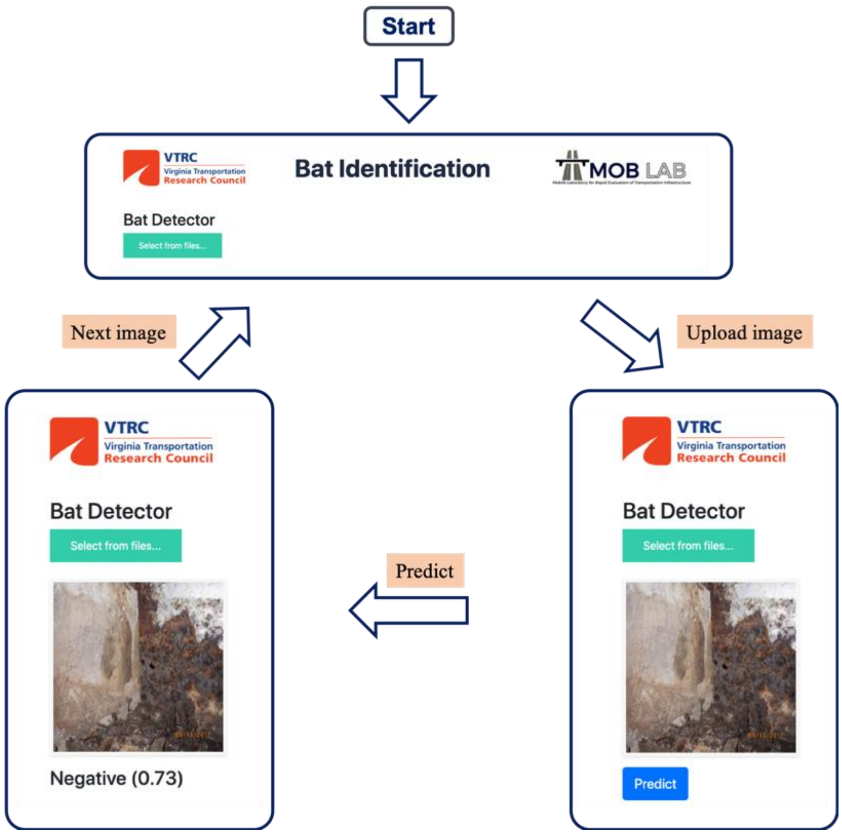


Figure 7. Screenshot of the initial web application interface

The classification result (whether or not the image contains bat indicators) is provided to the user display as well as a probability of the prediction as computed in Equation 4

$$P(y_i) = \frac{\exp(u_i)}{\sum_j \exp(u_j)} \quad (4)$$

where y_i is bat yes (positive)/bat no (negative) classification result and u_i is the output score for class i from the model.

The web app was developed in Python using a flask framework. The user-uploaded images are stored in the server, and it is anticipated that user images from various sources can be used to further improve the model performance. The web app was developed for the selected DenseNet-161 model and transferred to VDOT at the completion of the project.

CONCLUSIONS

- *Model performance results indicate that the model performs well overall in determining whether an image of a stain originated from a bat or from another source.* Model testing results included an 86.21% precision (indicating a fairly low percentage of false positives), a 100% recall (indicating no false negative determinations were made), and a 92.59% F-1 score (indicating a good balance between precision and recall).

- *The performance of the model can be improved by expanding the training image set through the collection of more data from both the positive and negative classes.*

- *When deployed, the interactive web app interface developed in this study will allow VDOT users to upload images of stains for model determination of whether the image contains bat indicators*. The probability of the prediction is also provided. User images can be used to further improve model performance.

RECOMMENDATIONS

1. *The Virginia Transportation Research Council should work with the VDOT Information Technology Division to host the image classification web app and make it accessible for use by VDOT bridge inspectors and environmental staff.*

2. *Before widespread deployment of the web app, the Virginia Transportation Research Council should conduct a pilot evaluation in several VDOT Districts.* The pilot study should consist of two components: (1) the web app should be used by bridge inspectors and environmental staff as a VDOT structure screening tool to reduce the likelihood that a stain on a structure is falsely perceived as originating from a non-bat source (i.e., a false negative) and (2) photographs from both the positive and negative indicator classes should continue to be collected and provided to VTRC for use as a training image set to strengthen the model.

IMPLEMENTATION AND BENEFITS

Implementation

With regard to Recommendation 1, VTRC has begun to coordinate with VDOT's Information Technology (IT) staff to discuss the feasibility of hosting the web app. Researchers have provided IT staff with the relevant details of the image classification model, such as the size, the user base, and the expected volume of photo uploads. IT staff have submitted a ticket to begin the work. VTRC staff will request that IT staff provide information on the status of the web app deployment every two weeks beginning February 15, 2020.

With regard to Recommendation 2, VTRC and the VDOT Endangered Species Program Manager will contact the VDOT Endangered Species Workgroup to determine whether their staff would be willing to participate in the 1-year pilot study and to gather their input on factors to consider for the study. The study will be initiated within two months of web app deployment.

It will be made clear to those using the tool that its purpose is to assist VDOT staff with determining whether further expert evaluations of the structure are necessary. It should be used as a screening tool to avoid false negatives (i.e. instances where a stain is incorrectly perceived as originating from a non-bat source) and the tool is not intended to replace expert level opinion when a VDOT staff member suspects bat occupation of a structure. Staff using the tool will be asked to provide VTRC with any existing photographs from both the positive and negative indicator classes and to save any new images collected as part of the pilot study. These images will be used as a training image set to strengthen the model.

Benefits

Implementing Recommendation 1 will provide VDOT with a tool that will enable the agency to more confidently determine that bats inhabiting VDOT structures are being documented and reported in accordance with state and federal regulations. This will serve as one of several tools in place (one of which is currently being developed) for VDOT's accurate assessment of structure occupancy by bats.

Implementing Recommendation 2 (conducting a pilot study) will benefit VDOT in two ways. First, the pilot study will determine the ease of use of the tool in the field and whether the tool meets its intended purpose of enabling VDOT staff to more confidently determine that bats inhabiting VDOT structures are being documented and reported in compliance with regulations. Second, because the pilot study will include the collection of additional photographs to continue training the image classification model, the accuracy and performance of the model will be strengthened prior to wider deployment.

ACKNOWLEDGMENTS

This project would not have been possible without the help of numerous people from a variety of organizations who provided photos, including Alyssa Bennett, Richard Bostwick, Dan Bozone, Katherine Caldwell, Scott Civjan, Terry Doonan, Lee Droppelman, Laura Finn, Mary Frazer, Amanda Glaze, Joy O'Keefe, MacKenzie Hall, Michael MacDonald, Melissa Miller, Laci Pattavina, and Kathy Shelton. Thanks also to the VTRC project monitor, Bridget Donaldson, and the VDOT technical review panel: David Bova, Amy Golden, Amy O'Leary, and Meg Short, and Eric Wolf. Appreciation is also extended to VDOT's Allison Wishon for help with web app implementation.

REFERENCES

1. U.S. Fish and Wildlife Service. Service. *Endangered and Threatened Wildlife and Plants; Review of Native Species That are Candidates for Listing as Endangered or Threatened; Annual Notice of Findings on Resubmitted Petitions; Annual Description of Progress on Listing Actions.* 2013.78:70104-70162.

2. Alves, D. M., L.C. Terribile, and D. Brito. The Potential Impact of White-Nose Syndrome on the Conservation Status of North American Bats. *PloS one,* 2014.

3. Ford, W.M., E. R. Britzke., C.A. Dobony, J.L. Rodrigue, and J.B. Johnson. Patterns of Acoustical Activity of Bats Prior to and Following White-Nose Syndrome Occurrence. *Journal of Fish and Wildlife Management,* 2011. 2:125-134.

4. Geluso, K., E. C. Keele, N. M. Pauley, I. R. Gomez, and S. P. Tye. Night-Roosting Behaviors for the Northern Long-Eared Myotis (*Myotis septentrionalis*) Under a Bridge Revealed by Time-Lapse Photography. *The American Midland Naturalist,* 2018. 179:287-293.

5. Johnson, J. B., M. A. Menzel, J. W. Edwards, and W. M. Ford. Gray bat night-roosting under bridges. *Journal of the Tennessee Academy of Science,* 2002.77:91-93.

6. Kiser, J., J. MacGregor, H. Bryan, and A. Howard. The Use of Concrete Bridges as Night Roosts by Indiana Bats in South Central Indiana. *Bat Research News,* 2011. 42:31.

7. VDOT Environmental Division. Preliminary Bat Inventory Guidelines for Bridges. 2010.

8. Fukui, D., N. Agetsuma, and D. A. Hill. Acoustic Identification of Eight Species of Bat (Mammalia: Chiroptera) Inhabiting Forests of Southern Hokkaido, Japan: Potential for Conservation Monitoring. *Zoological Science,* 2004. 21:947-956.

9. Rydell, J., H. Arita, M. Santos, and J. Granados. Acoustic Identification of Insectivorous Bats (order Chiroptera) of Yucatan, Mexico. *Journal of Zoology,* 2002. 257:27-36.

10. Keeley, B., and M. Tuttle. *Bats in American Bridges: Resource Publication No. 4*. In, Bat Conservation International, Inc., 1999.

11. Civjan, S. A., A. Berthaume, A. Bennett, and E. Dumont. Bat Roosting in Bridges: Pros and Cons of Assessment Methods from a New England Regional Study. *Transportation Research Record: Journal of the Transportation Research Board,* 2017. 2628:120-128.

12. Hendricks, P., S. Lenard, C. Currier, and J. Johnson. *Bat Use of Highway Bridges in South-central Montana*. In, Montana Department of Transportation Research Programs, 2005.

13. Cleveland, A. G., and J. G. Jackson. *Environmental Factors Influencing the Status and Management of Bats Under Georgia (USA) Bridges*. Presented at 2013 International Conference on Ecology and Transportation. North Carolina State University, Raleigh, 2013.

14. Bektas, B. A., Z. Hans, B. Phares, E. Nketah, J. Carey, M. K. Solberg, and K. McPeek. Most Likely Bridges as Roosting Habitat for Bats: Study for Iowa. *Transportation Research Record: Journal of the Transportation Research Board,* 2018. 2672:1-10.

15. Adam, M. D., and J. P. Hayes. Use of Bridges as Night Roosts by Bats in the Oregon Coast Range. *Journal of Mammalogy,* 2000. 81:402-407.

16. Sparks, D. W., D. Tull, T. Cable, R. Tunison, R. Perez, and E. Samanns. *Bridging the Gap Between Bats & Transportation Projects*. Prepared as part of NCHRP Project 25-25, Task 102, National Cooperative Highway Research Program, 2019.

17. U.S. Fish and Wildlife Service *Preliminary Bat Assessment Guidelines for Bridges/Structures.* Adapted from the Indiana Department of Transportation 2010 Bridge Inspection Manual. 2012.

19. Lin, Y., F. Lv, S. Zhu, M. Yang, T. Cour, K. Yu, L. Cao, and T. Huang. Large-Scale Image Classification: Fast Feature Extraction and SVM Training. In *CVPR 2011*, IEEE, 2011. pp. 1689-1696.

20. Sánchez, J., and F. Perronnin. High-Dimensional Signature Compression for Large-Scale Image Classification. In *CVPR 2011*, IEEE, 2011. pp. 1665-1672.

21. Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein. Imagenet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 2015. 115:211-252.

22. Zeiler, M. D., and R. Fergus. Visualizing and Understanding Convolutional Networks. In *European Conference on Computer Vision*, Springer, 2014. pp. 818-833.

23. Krizhevsky, A., I. Sutskever, and G. E. Hinton. Imagenet Classification with Deep Convolutional Neural Networks.In *Advances in Neural Information Processing Systems*, 2012. pp. 1097-1105.

24. Simonyan, K., and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014. *arXiv preprint arXiv:1409.1556*.

25. Szegedy, C., W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going Deeper with Convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015. pp. 1-9.

26. He, K., X. Zhang, S. Ren, and J. Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016. pp. 770-778.

27. Yosinski, J., J. Clune, Y. Bengio, and H. Lipson. How Transferable are Features in Deep Neural Networks? In *Advances in Neural Information Processing Systems,* 2014. pp. 3320-3328.

28. Sharif Razavian, A., H. Azizpour, J. Sullivan, and S. Carlsson. CNN Features Off-The-Shelf: An Astounding Baseline for Recognition. In *Proceedings of the IEEE Conference On Computer Vision and Pattern Recognition Workshops*, 2014. pp. 806-813.

29. Donahue, J., Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A Deep Convolutional Activation Feature for Generic Visual Recognition. In *International Conference on Machine Learning*, 2014. pp. 647-655.

30. Oquab, M., L. Bottou, I. Laptev, and J. Sivic. Learning and Transferring Mid-Level Image Representations using Convolutional Neural Networks. In *Proceedings Of The IEEE Conference on Computer Vision and Pattern Recognition*, 2014. pp. 1717-1724.

31. Gong, Y., L. Wang, R. Guo, and S. Lazebnik. Multi-Scale Orderless Pooling of Deep Convolutional Activation Features. In *European Conference on Computer Vision*, Springer, 2014. pp. 392-407.

32. Girshick, R., J. Donahue, T. Darrell, And J. Malik. Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation. In *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition*, 2014. pp. 580-587.

33. Cimpoi, M., S. Maji, I. Kokkinos, and A. Vedaldi. Deep filter banks for Texture Recognition, Description, and Segmentation. *International Journal of Computer Vision,* 2016. 118:65-94.

34. Castelluccio, M., G. Poggi, C. Sansone, and L. Verdoliva. Land use Classification in Remote Sensing Images by Convolutional Neural Networks. 2015. *arXiv preprint arXiv:1508.00092,* 2015.

35. Hu, F., G.-S. Xia, J. Hu, and L. Zhang. Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery. *Remote Sensing*, 2015. 7:14680-14707.

36. Shin, H.-C., H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers. Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging,* 2016. 35:1285-1298.

37. Donaldson, B. Email to bat-specific listservs., 2018.

38. White-nose Syndrome Response Team. Acceptable Management Practices for Bat Species Inhabiting Transportation Infrastructure. 2018. http://onlinepubs.trb.org/onlinepubs/am/infoforauthors.pdf. Accessed January 26, 2020.

39. Huang, G., Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. Densely Connected Convolutional Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. pp. 4700-4708.

40. Girshick, R., J. Donahue, T. Darrell, and J. Malik. Region-Based Convolutional Networks for Accurate Object Detection and Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence.* 2015. 38:142-158.

41. Kingma, D. P., and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

42. Goodfellow, I., Y. Bengio, and A. Courville. *Deep learning*. MIT press, 2016.

43. Paszke, A., S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. *Automatic Differentiation in Pytorch*. 2017.