

# Developing a Citizen Science Web Portal for Manual and Automated Ecological Image Detection

Marshall Mattingly III<sup>\*</sup>, Andrew Barnas<sup>†</sup>, Susan Ellis-Felege<sup>†</sup>, Robert Newman<sup>†</sup>, David Iles<sup>‡</sup>, Travis Desell<sup>\*</sup>  
Department of Computer Science<sup>\*</sup>, Department of Biology<sup>†</sup>

University of North Dakota  
Grand Forks, North Dakota 58202

Email: marshall.p.mattingly@und.edu, andrew.barnas@my.und.edu, susan.felege@email.und.edu,  
robert.newman@email.und.edu, tdesell@cs.und.edu

Department of Wildland Resources and the Ecology Center<sup>‡</sup>

Utah State University  
Logan, Utah 84322-5230  
Email: david.thomas.iles@gmail.com

**Abstract**—Image recognition is challenging in the field of wildlife ecology as samples of a specific species can be rare, making manual detection cumbersome. With over 2,060,000 images taken from motion-sensor trail cameras and unmanned aerial vehicle flights, a touch enabled web interface has been developed to allow citizen scientists and ecologists to categorize positive samples. To minimize categorization errors, the same images are shown to multiple separate users. The observations of each user are then compared using two novel validation strategies: percentage of overlapping area and maximum corner distance. Two novel methods for the extraction of final images from validated results are presented and compared as well: average corner points and area intersection. These methods were evaluated using a set of 142 images with a total of 811 observations of objects generated by citizen scientists that were manually inspected for ground truth. Results show that for this research a maximum corner distance of 10 pixels and the use of area intersection provided the best extracted imagery for future use as training and testing data by computer vision methods.

## I. INTRODUCTION

The Wildlife@Home project started as a hub for citizen scientists, non-field scientists who wish to help the processing of data, to help catalog video for a number of sub-projects with the goal of producing machine learning algorithms to automatically detect different species and their behaviors. With over 100,000 hours of video, cataloging would be impossible to do in a timely manner with ecologists and other trained professionals. This time limitation necessitated the creation of the Wildlife@Home web portal to allow citizen scientists to help. To date, citizen scientists have watched 34,234 hours of video and recorded 62,395 observations. This paper builds upon the Wildlife@Home project by allowing citizen scientists to review images, not just video. Citizen scientists can catalog different species and parameters in wildlife imagery to allow for machine learning algorithms to detect objects from the imagery in the future.

There are three current image sub-projects on Wildlife@Home: (1) trail camera (trailcam) images of common eiders in the Hudson Bay area of Canada; (2)

trailcam images of lesser snow geese in the Hudson Bay; and (3) unmanned aerial system (UAS) imagery taken from 75m, 100m, and 120m altitudes in the Hudson Bay consisting primarily of lesser snow geese. These sub-projects present different challenges, with the trailcam images being prone to obscuration by reeds, grass and other landscape elements and the UAS images having relatively small top-down species objects for detection. Even further, the species have evolved with *cryptic coloration* (camouflage), making them very difficult to identify. It is therefore important to ensure that citizen scientists are given proper training for both types of images to ensure objects extracted from the imagery can be used to train machine learning algorithms to automate the object detection in the future.

Similar to the video on Wildlife@Home, there are too many images for ecologists to manually inspect in a reasonable amount of time, with over two-million images currently collected and an estimated two-million more to collect each summer. Citizen scientists provide an excellent resource to review and catalog the images; however, given that the citizen scientists are not guaranteed to be field experts, precautions must be taken to minimize the potential of bad data entering the training dataset for machine learning.

This paper is organized into the following sections. Section II presents related works using crowd sourcing and citizen scientists to review field-specific imagery or video; Section III describes the different image data sets in use by Wildlife@Home; Section IV details the methodology describing the components of the image review interface and algorithms used to extract objects from the observations made by citizen scientists; Section V presents results showing a comparison of the algorithms used for object extraction; and the paper concludes with Section VI discussing the results and future areas of research.

## II. RELATED WORK

Crowd sourcing has been successfully used by citizen science projects to tackle problems requiring human feedback. GalaxyZoo [1], [2] has had great success in using volunteers to classify galaxies in images from the Sloan Digital Sky Survey [3]; and PlanetHunters [4] has been used to identify planet candidates in the NASA Kepler public release data. More recently, Snapshot Serengeti [5] has been created to classify images from camera traps in the Serengeti National Park.

In avian ecology, Cornell's NestCams project [6] has provided an outstanding resource for environmental education and gained popularity through the use of nest cameras to attract the public's interest in environmental science. NestCams primarily focuses on public outreach where video is collected opportunistically from cameras installed in bird houses, capturing a variety of cavity-nesting species. The CamClickr project has sparked applications of nest video archives for education in collegiate-level animal behavior courses [7]. More recently, eBird [8] is a citizen science project which allows users to upload observations of birds through handheld devices, providing spatio-temporal information about the bird distribution and abundance. Aside from CamClickr, few ecological citizen scientist projects have the volunteers reviewing images or videos to be used by researchers.

## III. WILDLIFE@HOME IMAGE DATA SETS

### A. Hudson Bay Trail Cameras

Data for the Hudson Bay Project was collected as part of David Iles' PhD research through the Utah State University [9]. Cameras were deployed to learn about predators destroying nests and in particular to estimate changes in predation rates by species such as polar bears which are coming ashore earlier than historically due to climate change [10], [11]. A total of 85 cameras and approximately 100 nests are monitored in each year between the Common Eiders and Lesser Snow Geese (see Fig. 1).

The images are collected by placing trail cameras on a stake at a nest located along the Hudson Bay near Churchill, Manitoba. Because eiders and to some extent snow geese are colonial nesters, multiple nests can be monitored using one camera. Time-lapse photography was used to monitor behaviors such as attendance patterns (time birds spend incubating the nest rather than off the nest for their own self-maintenance). Therefore, one image was taken every 2 minutes and then if movement occurred at the nest, a burst of 30 photos would be taken. Currently there are over 2 million photos, resulting in over 2 terabytes of data.

### B. Hudson Bay UAS Imagery

Manned aerial wildlife surveys are one of the leading causes of death for wildlife biologists, accounting for 66% of all deaths from 1937-2000 [12]. As such, UAS have become an area of high interest for performing these surveys more

safely [13]–[23]. In summer 2015, a Trimble UX5<sup>1</sup> fixed wing UAS was flown at Wapusk National Park in Manitoba, Canada. Survey flights were conducted from 11-24 June during the nesting season of lesser snow geese and common eiders and 11-15 July during the post-hatch period. Flights were conducted at 75m, 100m and 120m above ground level. A 16 megapixel Sony red, green, blue (RGB) camera flown in the Nader position was used to capture imagery along pre-defined transects with 80% overlap.

From the overlapping images, mosaics were created using Trimble Business Center (version 3.51)<sup>2</sup>. These mosaics were then processed into 100 smaller images for presentation on the Wildlife@Home image review interface due to size limitations. These flights produced over 60,000 images and 10 mosaics, resulting in over 1 terabyte of data.

## IV. METHODOLOGY

This section details the methodology behind the systems that allow the citizen scientists to make observations of objects in Wildlife@Home images, and the algorithms and methods that are applied to the observations to select a dataset for machine learning to automate the detection of objects in the future. The three facets required are: (1) an interface to show images for citizen scientists to create their observations; (2) algorithms to pair up observations of a single object from unique citizen scientists for each image; and (3) extraction of sub-images representing objects based on the set observations to create a training dataset of images for machine learning.

To create a consistent collection of citizen scientist observation data, several students from the Biology and Computer Science departments of UND were invited to review the mosaic UAS imagery on 18 April, 2016. In total, 8 out of 10 mosaics were reviewed, with 4 mosaics having observations from 2 unique citizen scientists and 1 mosaic having observations from 3 unique citizen scientists.

### A. Web Interface Creation

The first step of this project was to create an accessible and easy-to-use web-based interface for citizen scientists to review images from the over two-million set of wildlife imagery from trailcam and UAS flights in the Hudson Bay area of Canada. Wildlife@Home already had a robust interface for citizen scientists to watch and categorize events and species from video content. This video review interface served as a guide for the creation of the image review interface seen in Fig. 3. The major obstacles to overcome during development of the image review interface were: (1) images can be significantly larger than the typical viewport of a desktop monitor, especially in the case of the UAS imagery; (2) objects that are too small do not make good candidates for machine learning; and (3) the interface must be accessible on a wide-variety of operating systems and interfaces, including touch interfaces.

Users are able to double-click or double-tap to add boxes, which are movable and resizable, to the image to signify a

<sup>1</sup><http://uas.trimble.com/ux5>

<sup>2</sup><http://www.trimble.com/Survey/trimble-business-center.aspx>



Fig. 1. Imagery taken using motion sensing cameras along the Hudson Bay (credit David Iles). Photos can have multiple nests, *e.g.*, there are two nests in these pictures, one in the foreground and another in the background. The photos also have varying lighting conditions. Original photo resolution is 2048 x 1536 pixels.



Fig. 2. Imagery taken using an unmanned aerial system with a camera attached along the Hudson Bay. Photos can have multiple lesser snow geese and include both white and blue-phase lesser snow geese. Each image has two snow geese, with the right image having a blue phase snow goose to the right of a white phase snow goose. Original photo resolution varies and each image shown is 200x200 pixels and centered on the objects.



Fig. 3. Screenshot of the web-based image review interface on Wildlife@Home. The left half of the interface shows, from top to bottom: (1) the current image number; (2) a discussion button that allows users to share the image on the Wildlife@Home forums; (3) help buttons for both species identification and interface elements; (4) selection input to identify the species and parameters for the observation; (5) comment area to include text comments about the image; and (6) buttons to skip the image and get a new image, indicate that there is nothing in the image, or submit the observations for the image. The right half of the interface includes the HTML5 Canvas element with the image for review, scrollbars on the right and bottom indicating the current location within the image, and a current scale multiplier in the bottom right.

object, as shown surrounding the blue-phase lesser snow goose on the right-middle of Fig. 3. After the box is overlaid on to the image, the user is then able to specify the species of the creature and whether or not it is on a nest, as shown on the left side of Fig. 3. After the citizen scientist presses the submit button in the bottom of Fig. 3, the location, dimension, species, and on-nest status for each observation is stored in a database for processing by the object extraction algorithms before being used as the input training dataset for machine learning. If the citizen scientist does not notice any objects in an image, they may instead click the nothing here button in the bottom of Fig. 3 which is also stored in the database.

To overcome the issue of images significantly larger than a typical viewport, a script was written to break larger images down into 25 or 100 smaller images, depending on the size of the image. The resultant partial images are constrained to approximately 1280 pixels wide, allowing them to fit within a typical 1920 pixel width viewport, and stored in the reference database with their offset and dimension within the master image. In the case that the viewport is still smaller than the image being presented, an HTML5 Canvas element allows the user to pan along the image in all four directions and zoom in and out, as needed. The bars above and below the image in Fig. 3 indicate the current location within the image and the current zoom level is shown in the bottom right, with numbers greater than 1.0x indicating a zoom in and number less than 1.0x indicating a zoom out.

Machine learning relies on quality representative data to be able to automatically detect objects in future images. This means that images that are too small, determined to be less than 25-pixels square in the case of the UAS imagery, could make it difficult for our machine learning algorithms to detect objects. This issue is dealt with by enforcing a hard-limit on the minimum box size in the interface and providing clear instructions to citizen scientist on what should and should not be categorized which can be reviewed by pressing the interface help button in the left-top of Fig. 3. As shown in the results section, it is still not guaranteed that all users will adhere to these instructions. Presenting the same image to three users and selecting the objects for machine learning based on all responses helps to mitigate single-user errors.

An HTML5 Canvas library, built on JavaScript, was developed for the interface and released as a separate open-source project. Creating a stand-alone library was important as the interface is used in multiple sub-projects on the Wildlife@Home website, specifically trailcam images from the Hudson Bay, UAS imagery from the Hudson Bay, and mosaic images created from the UAS imagery in the Hudson Bay. The modularity of a separate library allows each of these project to control specific parameters and callbacks related to the project, while allowing all cross-platform development and optimization to happen separately. Using HTML5, JavaScript, and the open-source touch-interface library HammerJS<sup>3</sup>, the library is able to accept input from multi-touch devices (phones, tablets)

and traditional mouse / keyboard devices while scaling to the viewport. Future work for the library will be focused on adding further mobile interface improvements, specifically related to layout.

## B. Matching Observations

Before accepting observations by citizen scientists for inclusion in the output training dataset, we first compare all the observations in a single image with the observations of other citizen scientists on the same image. For this paper, all observations were produced on mosaic images from the UAS imagery and collected during a single event on 18 April, 2016 by students from the Biology and Computer Science departments at UND. For future work, three citizen scientists are required to view a single image before processing. To compare observations against each other, two different algorithms were developed: an area-overlap algorithm that compares the total amount of area shared between two observations and returns the percentage overlap, as shown in Eq. 1, and a corner-point distance algorithm that calculates the maximum distance between the each of the four corners of two observations, as shown in Eq. 2. Each of these algorithms is tested in the results section with different parameter values and compared against the actual matching observations as determined by manual examination.

$$\begin{aligned}
 l_{intersect} &= \min \|x_{12}, x_{22}\| - \max \|x_{11}, x_{21}\| \\
 h_{intersect} &= \min \|y_{12}, y_{22}\| - \max \|y_{11}, y_{21}\| \\
 A_{intersect} &= \max \|0, l_{intersect} * h_{intersect}\| \\
 A_{union} &= A_1 + A_2 - A_{intersect} \\
 A_{overlap} &= \frac{A_{intersect}}{A_{union}}
 \end{aligned} \tag{1}$$

where,

$l_{intersect}$  is the length of the intersection of two observations  
 $h_{intersect}$  is the height of the intersection of two observations  
 $A_{intersect}$  is the area of intersection of two observations  
 $A_{union}$  is area of the union of two observations  
 $A_1$  is the area of observation 1  
 $A_2$  is the area of observation 2  
 $A_{overlap}$  is ratio of overlap of two observations

$$\begin{aligned}
 c_0 &= \sqrt{(x_{11} - x_{21})^2 + (y_{11} - y_{21})^2} \\
 c_1 &= \sqrt{(x_{12} - x_{22})^2 + (y_{11} - y_{21})^2} \\
 c_2 &= \sqrt{(x_{12} - x_{22})^2 + (y_{12} - y_{22})^2} \\
 c_3 &= \sqrt{(x_{11} - x_{21})^2 + (y_{12} - y_{22})^2} \\
 c_{max} &= \max \|c_0, c_1, c_2, c_3\|
 \end{aligned} \tag{2}$$

where,  $c_0$ ,  $c_1$ ,  $c_2$ , and  $c_3$  are the distances between the top-left, top-right, bottom-right, and bottom-left corners of two observations, respectively, and  $c_{max}$  is the maximum distance between any two corners.

<sup>3</sup><http://hammerjs.github.io/>

### C. Extracting Objects

After objects identified by citizen scientists have been paired together, a single sub-image is extracted to represent the object as agreed upon by the matches. Automated object recognition relies heavily on an input dataset that minimizes negative space around an object while maximizing the amount of the object in the data. This is particularly challenging when using data gathered by citizen scientists due to potential lack of expertise and human error. Two methods were developed to attempt to satisfy the needs of the machine learning algorithms.

The average object extraction method, algorithm seen in Eq. 3, takes the average of the all four corners of a set of paired observations and uses that as the corner points to extract a partial image with the given extents to represent the object detected. This method gives equal weight to all observations by the citizen scientists, which allows a single observation to skew the resultant object. When the set of paired observations has little variability, the average method produces images that represent the objects well; however, as the variability of paired observations increases, the results can trend toward the poor observations, producing images that poorly represent objects by having too much negative space.

$$\begin{aligned} (x_0, y_0) &= \left( \frac{\sum_{i=1}^n x_{i0}}{n}, \frac{\sum_{i=1}^n y_{i0}}{n} \right) \\ (x_1, y_1) &= \left( \frac{\sum_{i=1}^n x_{i1}}{n}, \frac{\sum_{i=1}^n y_{i1}}{n} \right) \end{aligned} \quad (3)$$

where,

- $(x_0, y_0)$  is the top-left corner of the object
- $(x_1, y_1)$  is the bottom-right corner of the object
- $n$  is the number of observations in the matched set
- $x_{i0}$  is the left-most extent of the  $i^{th}$  observation
- $y_{i0}$  is the top-most extent of the  $i^{th}$  observation
- $x_{i1}$  is the right-most extent of the  $i^{th}$  observation
- $y_{i1}$  is the bottom-most extent of the  $i^{th}$  observation

The intersection object extraction method, algorithm seen in Eq. 4, uses the intersection of the set of paired observations to extract a partial image with the given extents to represent the object detected. This method gives the most weight to the smallest observation in the matched set, which allows a single observation to skew the resultant object. When paired observations are all large enough to surround the actual object being detected, the intersection method ensures that the negative space is minimized, producing excellent object results; however, if a single observation is smaller than the object being detected, the resultant partial image will be an object with too little positive space. As shown in the results section, the drawbacks of the average algorithm are more severe than the drawbacks of the intersection algorithm.

$$\begin{aligned} (x_0, y_0) &= \left( \max_{i=1}^n \|x_{i0}\|, \max_{i=1}^n \|y_{i0}\| \right) \\ (x_1, y_1) &= \left( \min_{i=1}^n \|x_{i1}\|, \min_{i=1}^n \|y_{i1}\| \right) \end{aligned} \quad (4)$$

where the variables definitions are the same as in Eq. 3.

### V. RESULTS

Two observation equivalence algorithms were run against a set of 142 images with a total of 811 observations of objects made by citizen scientists. Two observations by two different citizen scientists are said to match if they represent the same detected object, while observations that do not have any matches are considered non-matched. Each image in the set comes from mosaics created from UAS imagery from the Hudson Bay, Canada and was viewed and cataloged by 2 or 3 citizen scientists using the Wildlife@Home image review interface. The observations were manually inspected to record the actual observation equivalence set, which was used to determine the ratio of correct matches and non-matches, as well as the false positives and false negatives of the two observation equivalence algorithms.

The matched ratio and non-matched ratio are key values used throughout this section to determine the success rate of the observation equivalence algorithms. The matched ratio is the number of matched observation pairs created by an observation equivalence algorithm divided by the actual number of matched observation pairs as determined via manual examination. The non-matched ratio is the number of observations created by an observation equivalence algorithm that are not matched with any other observations in the set divided by the actual number of non-matched observations as determined by manual examination. A ratio of 1.0 indicates perfect matching or non-matching while a ratio below 1.0 means the object equivalence algorithm is missing matches or non-matches and a ratio above 1.0 means the object equivalence algorithm has too many matches or non-matches.

#### A. Overall Observation Equivalence Algorithm Accuracy

The overall accuracy of the two observation equivalence algorithms is shown in Table I with multiple parameter values which are then compared against the actual observation equivalence set as determined by manual examination of the observations. The area overlap equivalence algorithm has a decent matched ratio, detecting 0.88 times as many observation matches as actually exist (matched ratio) while failing to match 1.34 times as many objects as should actual fail to match (non-matched ratio). As the area overlap requirement is increased, there is a linear decrease in number of matches and an exponential increase in non-matched observations, as in Fig. 4.

The corner-point equivalence algorithm starts at a maximum distance of 5-pixels between all corners, resulting in a low 0.59 matched ratio and a high 3.75 non-matched ratio, shown in Table I. However, increasing the threshold to 10-pixels gives

TABLE I  
OBSERVATION EQUIVALENCE ALGORITHM RESULTS AND COMPARISONS AGAINST THE ACTUAL OBSERVATION EQUIVALENCE SET AS DETERMINED BY MANUAL EXAMINATION OF THE 811 OBSERVATIONS FROM 142 IMAGES IN THE TEST DATASET

Algorithm	Matches	Non-Matched	Matched Ratio	Non-Matched Ratio	False Positives	False Negatives
Actual	400	91	1.00	1.00	0	0
Area (50%)	352	122	0.88	1.34	0	54
Area (60%)	329	159	0.82	1.74	0	92
Area (70%)	266	282	0.66	3.10	0	214
Area (80%)	186	440	0.46	4.83	0	370
Area (90%)	81	649	0.20	7.13	0	566
Point (5px)	238	341	0.59	3.75	0	272
Point (10px)	379	106	0.95	1.16	0	24
Point (15px)	404	91	1.01	1.00	8	0
Point (20px)	414	91	1.03	1.00	18	0

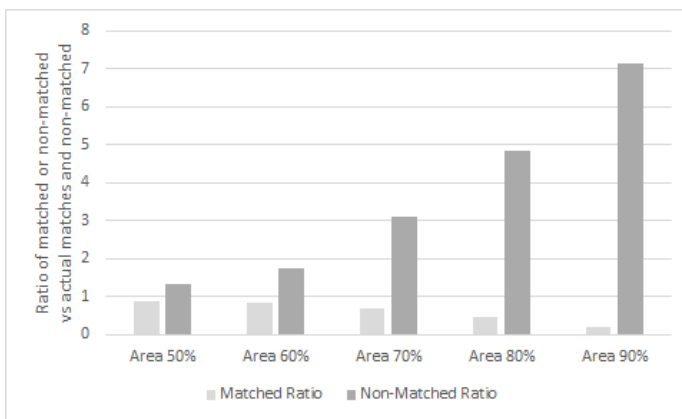


Fig. 4. This chart shows the matched and non-matched ratio of the area overlap equivalence algorithm when compared against the actual matches and non-matches as determined by manual examination of the observations. The area overlap equivalence algorithm is shown with 50%, 60%, 70%, 80%, and 90% minimum area overlap requirements.

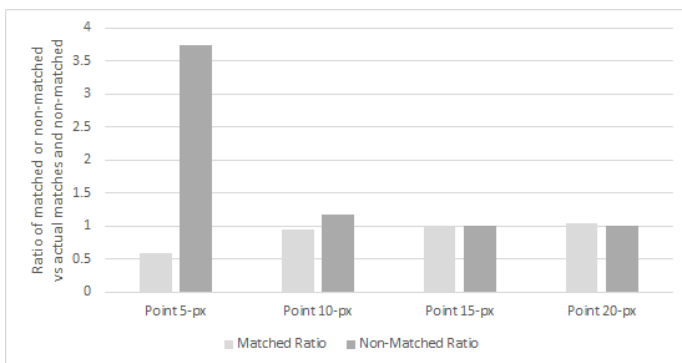


Fig. 5. This chart shows the matched and non-matched ratio of the corner-point equivalence algorithm when compared against the actual matches and non-matches as determined by manual examination of the observations. The corner-point equivalence algorithm is shown with 5-px, 10-px, 15-px, and 20-px maximum distances.

a dramatic improvement resulting in a 0.95 matches ratio and a 1.16 non-matched ratio, significantly better than best result from the area overlap equivalence method (0.96 and 1.34 matches and non-matched ratios, respectively). While the area overlap algorithm provided an exponential change between cutoffs, the corner-point algorithm actually plateaus at 10-pixels, as in Fig. 5. In fact, the corner-point algorithm begins to over-match as the cutoff is increased, meaning the parameters are too loose, resulting in observations being matched not only to their true matches as confirmed by manual examination, but to additional observations nearby.

#### B. False Positive / Negative Analysis

The previous section looked at the overall accuracy between the two observation equivalence algorithms with various parameters when compared to the results of manual examination. These results provide an excellent understanding of basic effectiveness of the algorithms with different parameters; however, to determine the best algorithm to use on a large dataset, the error-rates, seen as false positives and false negatives in Table I, of the equivalence algorithms must also be known. False positives indicate observations that were matched by the algorithm, but do not match as determined by the manually examined observation equivalence set. False positives result in the over-matching seen in the corner-point algorithm when the cutoff was 15-pixels or more and will pollute the training dataset with incorrect or misaligned objects. False negatives are observations that the algorithm failed to match, but should be matched according to the manually examined observation equivalence set. False negatives leave the training dataset with too few objects for machine learning. The effects of these two must be taken into strong consideration when determining the appropriate observation equivalence algorithm.

The area overlap observation equivalence algorithm in Fig. 6 does not produce any false positives for the given parameters and observation set; however, the false negatives increase exponentially as the area overlap algorithm is tightened. The corner-point equivalence algorithm in Fig. 7 starts with a significant amount of false negatives and no false positives. The false negatives disappear quickly while the false positives

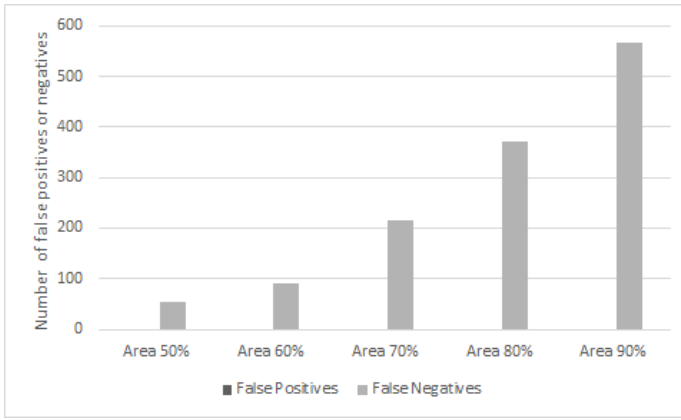


Fig. 6. This chart shows the total number of false positives and false negatives created by the area overlap equivalence algorithm when compared against the actual observation equivalence set as determined by manual examination. The area overlap equivalence algorithm false positives and false negatives are shown with 50%, 60%, 70%, 80%, and 90% minimum area overlap requirements.

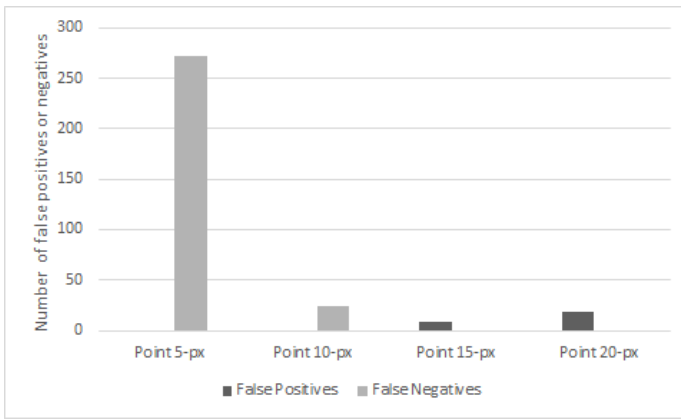


Fig. 7. This chart shows the total number of false positives and false negatives created by the corner-point equivalence algorithm when compared against the actual observation equivalence set as determined by manual examination. The corner-point equivalence algorithm false positives and false negatives are shown with 5-px, 10-px, 15-px, and 20-px maximum distances.

actually increase from 0 to 18 at a corner-point distance of 20-pixels. This phenomenon confirms the over-matching in Fig. 5.

False positives correlate well with the overall matched ratio of a given observation equivalence algorithm. If the matched ratio is below 1.0, false positives are held to 0; however, as the matched ratio increases above 1.0, known as over-matching, false positives occur. Similarly, false negatives correlate well with the overall non-matched ratio of a given observation equivalence algorithm. As the non-matched ratio increases, the false negatives similarly increase with an  $r^2$  of 0.997. This suggests that maximizing the matched ratio below 1.0 and minimizing the non-matched ratio above 1.0 should result in the best equivalence algorithm, resulting in the corner-point equivalence algorithm at 10-pixels as the best overall method. However, it will be important to continue testing with more algorithms, more parameters, and different datasets to ensure that the results hold.

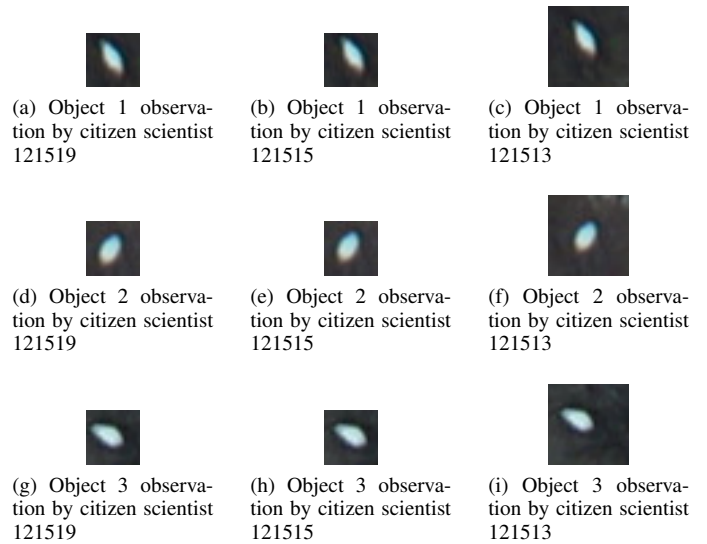


Fig. 8. Comparison of the partial images created from three sets of observations made by three different citizen scientists on the same three objects. The left and middle observations by citizen scientists 121519 and 121517, respectively, are similar and would make for decent inputs for machine learning. The right observations by citizen scientist 121513 are skewed and create too much negative space for machine learning.

### C. Observation Variability by User

Variability in observations on the same object made by multiple citizen scientists makes pairing observations and selecting the correct bounds that represent the actual object difficult. A single citizen scientist can create an observation that may potentially skew the object extraction by creating bounds that are too large and potentially not centered on the object, as in Fig. 8.

To examine the variability in citizen scientist observations, a subset of 9 images with a total of 25 objects observed by three citizen scientists was extracted from the manually inspected observation equivalence set. This allowed for the comparison of the corner-point distances of the three matched observations between the citizen scientists, *i.e.*, the observation pair of citizen scientists 1 and 2; the observation pair of citizen scientists 1 and 3; and the observation pair of citizen scientists 2 and 3 for the same object. This data for each image is enumerated in Table II, where Observation Pair (1, 2), Observation Pair (1, 3), and Observation Pair (2, 3) are the data for the pair of observations of citizen scientists 1 and 2, the pair of observations of citizen scientists 1 and 3, and the pair of observations of citizen scientists 2 and 3, respectively. The *Mean* and *STDEV* results were calculated for each matched observation pair set and included for analysis.

The mean corner-point distance for the matched observation pairs of citizen scientists 1 and 2 was 2.50 pixels with a standard deviation of 1.23 pixels, meaning citizen scientists 1 and 2 created similar observations of the 25 objects. The mean corner-point distance for the matched observation pairs with citizen scientist 3 was 8.99 pixels and 9.27 pixels with a standard deviation of 2.51 pixels and 2.36 pixels for the

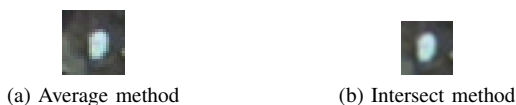


Fig. 9. This figure shows the similarity between the average corner-points extraction method (left) and the area intersection extraction method (right) when the observations by citizen scientists have near identical bounds around the object detected.



Fig. 10. This figure shows the potential major difference between the average corner-points extraction method (left) and the area intersection extraction method (right) when the observations made by citizen scientists have greatly different bounds around the object detected. The average corner-points extraction method produces skewed bounds around the object with a large amount of negative space, while the area intersection extraction method produces bounds around the object with comparatively little negative space.

matched observation pairs with citizen scientist 1 and 2, respectively. Smaller corner-point distance average (less than a third) and standard deviation (less than half) of citizen scientists 1 and 2 when compared against their respective observations with citizen scientist 3, it is apparent that the citizen scientist 3 created poor observations of the objects. In fact, 20 out of 50 (or 40%) of the matched observation pairs with citizen scientists 3 had a corner-point distance greater than 10 pixels and would fail to be matched by the best performing observation equivalence algorithm from Table I.

#### D. Selecting Final Object for Machine Learning

The previous sections have been concerned with pairing observations identified by citizen scientists. After the observations have been paired, a singular partial image of the object is extracted from the image using a combination of the bounds in the observations. Similar to the area overlap and corner-point equivalence algorithm, two methods for selecting a single partial image representing an object are compared: (1) an image using the average corner-points of the paired observations; and (2) an image with the intersect of the paired observations.

The average corner-points method returns similar results, as in Fig. 9, to the intersect method when citizen scientists are consistent in placing boxes around objects. However, a single citizen scientist that produces a box significantly too large or too small can have negative impact on resultant partial image for the object, leaving the machine learning algorithm with an object that has too much negative space, as in Fig. 10, or not enough positive space.

The intersection method deals with this issue by creating the smallest box possible based on the input observations from the citizen scientists. This ensures that a citizen scientist who produces too large a box cannot negatively affect the resultant partial image for the object; however, a citizen scientist producing too small a box can still impact the quality

of the resulting objects by not producing enough positive space. Initial manual analysis indicates that citizen scientists produce more boxes that are too large than too small; in fact, there wasn't a single instance in the observation sets where the smallest observation in the set cut off more than a couple pixels of the object. This suggests that the intersection method is the most consistent method to ensure good partial images of objects for machine learning.

## VI. CONCLUSION

Overall, citizen scientists did an excellent job agreeing on objects in a given image, with only 91 out of 811 (11.2%) observations failing to be matched to another observation. The corner-point object equivalence algorithm with a cutoff of 10-pixels did the best job pairing observations of a single object together, with a 0.95 matches ratio and 1.16 non-matched ratio. After observations are paired together, the intersection method produced the most consistent results to minimize the negative space around objects while maximizing the positive space of the object.

The 10-pixel equivalence algorithm will therefore be used to create a training dataset for a neural network for the automated detection of objects. After the neural network has been successfully training, a BOINC<sup>4</sup> client will be created to allow citizen scientists to provide computer resources to aid in the detection of objects in future imagery. The creation, training, and BOINC client for the neural network will be the focus of a complimentary paper.

Citizen scientist provide their time to help identify the objects in the imagery. However, as shown in the results sections, not all citizen scientists produce usable bounding boxes around objects. It is therefore important to both study the situations in which poor bounding boxes are created and to provide incentives to create better bounding boxes.

All citizen scientists responses in this paper were students at UND from multiple disciplines giving their time in return for free food. This incentive external to identifying objects could potential create bias in some respondents; it is therefore the intention of the authors to keep a log of any such events where respondents are given an external reward for providing data and comparing the bounding boxes with those of citizen scientists who are providing their time and effort without external factors. Additional research of bias in respondents, especially consistent outliers, will be a focus of future research.

To provide an incentive to citizen scientists working in their free time a system of gamification will be implemented. Currently, citizen scientists on the Wildlife@Home project are given points based on computer resources provided and for identifying objects and events in video. After the neural networks are trained and automated, citizen scientists will be given points for: (1) any identified objects that make it into the training set (*i.e.* that another citizen scientist also identifies); (2) any identified objects that are confirmed by the neural

<sup>4</sup><http://boinc.berkeley.edu/>



TABLE II  
 MEAN ACTUAL CORNER-POINT DISTANCE (IN PIXELS) BETWEEN OBSERVATION PAIRS OF THREE CITIZEN SCIENTISTS WHO MADE OBSERVATIONS ON THE SAME 25 OBJECTS FROM 9 IMAGES AS DETERMINED BY MANUAL EXAMINATION.

Image ID	Object #	Observation Pair (1, 2)	Observation Pair (1, 3)	Observation Pair (2, 3)
4016613	1	1.41	9.22	9.22
4016613	2	1	7.81	7.07
4016616	1	2.24	4.24	5
4016616	2	2	10.8	9.22
4016616	3	2	3.16	4.24
4016617	1	2	10.3	10.8
4016618	1	1.41	12.7	12.8
4016618	2	2.24	7.07	9.22
4016618	3	2	9.43	10
4016618	4	3.16	10	10.5
4016621	1	3	10	12
4016622	1	2.24	8.06	10
4016622	2	2	6.4	7.81
4016622	3	4.47	7.21	7.81
4016622	4	3	12.8	14.1
4016622	5	6.4	10	10.6
4016624	1	2.23	14.9	12.7
4016624	2	1.41	9.43	9.22
4016624	3	2.83	10.3	12.2
4016624	4	4.12	9.22	10
4016624	5	1.41	7.07	8.48
4016627	1	1.41	7.81	7.81
4016627	2	1.41	9.43	8.6
4016627	3	3.16	8.48	7.81
4016628	1	4	8.6	7.07
<b>Total</b>	<b>25</b>	<b>62.55</b>	<b>224.44</b>	<b>234.28</b>
<b>Mean</b>		<b>2.50</b>	<b>8.98</b>	<b>9.37</b>
<b>STDEV</b>		<b>1.23</b>	<b>2.51</b>	<b>2.36</b>

networks; and (3) providing computer resources via BOINC to aid in the automated detection of objects.

Further future work will be focused on: (1) resolving issues where only one or two citizen scientists make an observation of an object (is the object ignored because it was not observed by all three citizen scientists, or do we accept objects with at least two observers); (2) enhancing the web interface for mobile devices; (3) providing clearer instructions to citizen scientists to increase the probability of each citizen scientists making good observations; (4) comparing the citizen scientists against observations made by field experts to show that the citizen scientists produce results good enough for machine learning; and (5) providing a large annotated data release of training and testing imagery for the computer vision community.

#### ACKNOWLEDGMENTS

We appreciate the support and dedication of the Wildlife@Home citizen scientists who have spent significant amounts of time reviewing images and classifying objects within them. Funding was provided by North Dakota EPSCoR, the Hudson Bay Project, Central and Mississippi Flyways, National Geographic, and UND's College of Arts and Sciences.

#### REFERENCES

- [1] C. J. Lintott, K. Schawinski, A. Slosar, K. Land, S. Bamford, D. Thomas, M. J. Raddick, R. C. Nichol, A. Szalay, D. Andreescu, P. Murray, and J. Vandenberg, "Galaxy zoo: morphologies derived from visual inspection of galaxies from the sloan digital sky survey," *Monthly Notices of the Royal Astronomical Society*, vol. 389, no. 3, pp. 1179–1189, 2008. [Online]. Available: <http://dx.doi.org/10.1111/j.1365-2966.2008.13689.x>
- [2] C. Lintott, K. Schawinski, S. Bamford, A. Slosar, K. Land, D. Thomas, E. Edmondson, K. Masters, R. C. Nichol, M. J. Raddick, A. Szalay, D. Andreescu, P. Murray, and J. Vandenberg, "Galaxy zoo 1: data release of morphological classifications for nearly 900,000 galaxies," *Monthly Notices of the Royal Astronomical Society*, vol. 410, no. 1, pp. 166–178, 2011. [Online]. Available: <http://dx.doi.org/10.1111/j.1365-2966.2010.17432.x>
- [3] J. e. a. Adelman-McCarthy, "The 6th Sloan Digital Sky Survey Data Release, <http://www.sdss.org/dr6/>," July 2007, apJS, in press, arXiv/0707.3413.
- [4] D. A. Fischer, M. E. Schwamb, K. Schawinski, C. Lintott, J. Brewer, M. Giguere, S. Lynn, M. Parrish, T. Sartori, R. Simpson, A. Smith, J. Spronck, N. Batalha, J. Rowe, J. Jenkins, S. Bryson, A. Prsa, P. Tenenbaum, J. Crepp, T. Morton, A. Howard, M. Belevu, Z. Kaplan, N. vanNispen, C. Sharzer, J. DeFouw, A. Hajduk, J. P. Neal, A. Nemeec, N. Schuepbach, and V. Zimmermann, "Planet hunters: the first two planet candidates identified by the public using the kepler public archive data," *Monthly Notices of the Royal Astronomical Society*, vol. 419, no. 4, pp. 2900–2911, 2012. [Online]. Available: <http://dx.doi.org/10.1111/j.1365-2966.2011.19932.x>
- [5] Lion Research Center, University of Minnesota, [Accessed Online, 2012] <http://www.snapshotserengeti.org/>.

- [6] R. Bonney, C. B. Cooper, J. Dickinson, S. Kelling, T. Phillips, K. V. Rosenberg, and J. Shirk, "A developing tool for expanding science knowledge and scientific literacy," *BioScience*, vol. 59, pp. 977–984, 2009.
- [7] M. A. Voss and C. B. Cooper, "Using a free online citizen-science project to teach observation and quantification of animal behavior," *American Biology Teacher*, vol. 72, pp. 437–443, 2012.
- [8] C. Wood, B. Sullivan, M. Iliff, D. Fink, and S. Kelling, "ebird: engaging birders in science and conservation," *PLoS biology*, vol. 9, no. 12, p. e1001220, 2011.
- [9] D. Iles, D. Koons, R. Rockwell, C. Mulder, and S. Ellis-Felege, "Unpublished data," 2013–2014.
- [10] L. J. Gormezano and R. F. Rockwell, "Dietary composition and spatial patterns of polar bear foraging on land in western hudson bay," *BMC ecology*, vol. 13, no. 1, p. 51, 2013.
- [11] —, "What to eat now? shifts in polar bear diet during the ice-free season in western hudson bay," *Ecology and evolution*, vol. 3, no. 10, pp. 3509–3523, 2013.
- [12] D. B. Sasse, "Job-related mortality of wildlife workers in the united states, 1937–2000," *Wildlife society bulletin*, pp. 1015–1020, 2003.
- [13] M. Israel, "A uav-based roe deer fawn detection system," in *Proceedings of the International Conference on Unmanned Aerial Vehicle in Geomatics (UAV-g)*, H. Eisenbeiss, M. Kunz, and H. Ingensand, Eds, vol. 38, 2011, pp. 1–5.
- [14] B. E. Wilkinson, "The design of georeferencing techniques for unmanned autonomous aerial vehicle video for use with wildlife inventory surveys: A case study of the national bison range, montana," Ph.D. dissertation, University of Florida, 2007.
- [15] R. P. Breckenridge, M. Dakins, S. Bunting, J. L. Harbour, and S. White, "Comparison of unmanned aerial vehicle platforms for assessing vegetation cover in sagebrush steppe ecosystems," *Rangeland Ecology & Management*, vol. 64, no. 5, pp. 521–532, 2011.
- [16] W. Koski, P. Abgrall, and S. Yazvenko, "A review and inventory of unmanned aerial systems for detection and monitoring of key biological resources and physical parameters affecting marine life during offshore exploration and production activities," *IWC paper SC/61 E*, vol. 9, 2009.
- [17] P. Soriano, F. Caballero, A. Ollero, and C. A. de Tecnologías Aeroespaciales, "Rf-based particle filter localization for wildlife tracking by using an uav," in *On: 40th International Symposium of Robotics, Barcelona, España*, 2009.
- [18] M. Leonardo, A. M. Jensen, C. Coopmans, M. McKee, and Y. Chen, "A miniature wildlife tracking uav payload system using acoustic biotelemetry," in *ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2013, pp. V004T08A056–V004T08A056.
- [19] R. P. Breckenridge, R. Lee, A. Piscitella, and C. Eckersell, "Using unmanned aerial vehicles to assess vegetative cover in sagebrush steppe ecosystems," Idaho National Laboratory (INL), Tech. Rep., 2005.
- [20] J. Linchant, C. Vermeulen, J. Lisein, P. Lejeune, and P. Bouché, "Using drones to count the elephants: a new approach of wildlife inventories," 2013.
- [21] K. A. Steen, A. Villa-Henriksen, O. R. Therkildsen, H. Karstoft, O. Green *et al.*, "Automatic detection of animals using thermal imaging," in *International Conference on Agricultural Engineering*, 2012.
- [22] C. Cariappa, W. Ballard, S. Breck, A. J. Piaggio, and M. Neubaum, "Estimating population size of mexican wolves noninvasively (arizona)," 2008.
- [23] C. Vermeulen, P. Lejeune, J. Lisein, P. Sawadogo, and P. Bouché, "Unmanned aerial survey of elephants," *PLoS one*, vol. 8, no. 2, pp. 1–7, 2013.