

Aggregating User Input in Ecology Citizen Science Projects

Greg Hines *

Citizen Science Group
Department of Physics
University of Oxford
greg@zooniverse.org

Alexandra Swanson *

Citizen Science Group
Department of Physics
University of Oxford
ali@zooniverse.org

Margaret Kosmala

Department of Organismic and
Evolutionary Biology
Harvard University
kosmala@fas.harvard.edu

Chris Lintott *

Citizen Science Group
Department of Physics
University of Oxford
cjl@astro.ox.ac.uk

Abstract

Camera traps (remote, automatic cameras) are revolutionizing large-scale studies in ecology. The Serengeti Lion Project has used camera traps to produce over 1.5 million pictures of animals in the Serengeti. To analyze these pictures, the Project created Snapshot Serengeti, a citizen science website where volunteers can help classify animals. To increase accuracy, each photo is shown to multiple users and a critical step is aggregating individual classifications. In this paper, we present a new aggregation algorithm which achieves an accuracy of 98.6%, better than many human experts. Our algorithm also requires fewer users per photo than existing methods. The algorithm is intuitive and designed so that non-experts can understand the end results.

Ecology seeks to understand the interrelationships of species with one another and with their environment. Monitoring many species of animals simultaneously has traditionally been very difficult. Camera traps (remote, automatic cameras) are revolutionizing ecological research by providing a non-invasive, cost-effective approach for large-scale monitoring. Ecologists are currently using these traps in the Serengeti National Park, one of the world's last large intact natural areas, to understand the dynamics of its dozens of large mammals (Swanson et al. 2014). As of November 2013, the ecologists have spent 3 years using more than 200 cameras spread over 1,125 square kilometers to take more than 1.5 million photos. In order to process so many images, the ecologists, along with Zooniverse (a citizen science platform), created Snapshot Serengeti, a web site where over 35,000 volunteers helped classify the species in the photos (Zooniverse 2014a).

Since volunteers can make mistakes, each photo is shown to multiple users. A critical step is to combine these classifications into one *aggregate* classification: e.g., if 4 out of 5 users classify a photo as containing a zebra, we might

decide that the photo does indeed contain a zebra. In this paper, we develop an aggregation algorithm for Snapshot Serengeti. Classification aggregation is an active area in machine learning; however, we show that much of the existing literature is based on assumptions which do not apply to Snapshot Serengeti, and must therefore develop a novel approach.¹ In addition, current machine learning work on classification aggregation often draws on ideas such as *expectation maximization* and *Bayesian reasoning*. While powerful, these methods obscure the connection between input and results, making it hard for non-machine learning experts to understand the end results. Thus, our algorithm must be both *accurate* and *intuitive*.

Our paper proceeds as follows. We begin by discussing Snapshot Serengeti and previous machine learning literature on classifier aggregation. We then discuss why much of this existing work is not applicable to Snapshot Serengeti. We next introduce a new classifier aggregation algorithm for Snapshot Serengeti and compare it against the current algorithm. Finally, we conclude and discuss possible future work.

Snapshot Serengeti

The Serengeti Lion Project is studying the population and community dynamics of wildlife in the Serengeti National Park, Tanzania (Swanson et al. 2014). To process the 1.5 million photos produced so far by their remote camera survey the project partnered with Zooniverse to create the website Snapshot Serengeti. Citizen science projects such as Snapshot Serengeti use volunteers to help analyze data (Zooniverse 2014b). With Snapshot Serengeti, users are presented with randomly selected photos and choose, from a list of 48 species, which species were in each photo. Users also note how many animals were present in each photo, whether they were eating and whether there were any young animals.

*These authors are part of the Zooniverse project, funded in part by MICO.

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Classification aggregation is often referred to as *classification combination* in machine learning and *data reduction* in citizen science projects. All are closely related to *boosting* but without the theoretical error bounds.

Algorithm 1 Current Snapshot Serengeti classification aggregation algorithm

```

if first 5 classifications are blank then
  return blank
else if any 10 classifications are blank then
  return blank
else if any 10 users have given classification  $c$  then
  return  $c$ 
else if 25 users have classified  $i$  then
   $m :=$  median number of species each user reports
  return  $m$  most frequent species
end if

```

An advantage of Snapshot Serengeti is that it was able to rely on a large pool of volunteers to analyze the data (almost 35,000 users have processed at least one image). The challenge is that volunteers are not experts and may not be good at identifying different species. For example, a volunteer could understandably confuse a Grant’s gazelle with a Thomson’s gazelle. In addition, for many of the photos, the animals were mostly off the camera or too close and out of focus. The solution was to show each photo to multiple people and *aggregate* their results into one overall *classification*. In this paper, we focus on aggregating species classifications.

The aggregation algorithm currently used by Snapshot Serengeti, shown in Algorithm 1, was created by Swanson *et. al.* specifically for their project. Each photo was shown to users until one of the conditions in Algorithm 1 was met, at which point the image was retired. The first two conditions are designed to find *blank* photos, *i.e.* those photos not containing any animals. Blank photos were common because blowing grass often activated the motion sensitive cameras. Users were therefore given the option to label photos as blank. Users could use the blank label on non-blank photos, however, to indicate they were unsure how to classify that photo. Thus, the first two conditions aim to find the photos that are actually blank. The third condition in Algorithm 1 is for the case when 10 users (not necessarily consecutive) have given the same classification (only in terms of species present). The final condition is for when 25 users have seen the photo: if m is the median number of species each user reports seeing, the m most common species are returned.

Most photos (74.6%) were blank. Of those blank photos, 93.8% were retired via condition 1. For non-blank photos, 97.3% were retired via condition 3. Non-blank photos required an average 16.6 user classifications. On average, over all photos (blank and non-blank), Algorithm 1 needed 7.9 user classifications per photo.

Swanson *et. al.* created a set of 4,149 non-blank photos with gold standard classifications. As of July 2014, Algorithm 1 achieved an accuracy of 96.4% with respect to this gold standard data. While this is an excellent rate, we note two issues. First, although Snapshot Serengeti has been a very popular project to date it is not a guarantee of future success. In the future an average of even 8 users per photo may not be a realistic target. Thus, there is a real need to

	0	1
0	TN	FN
1	FP	TP

Table 1: A general confusion matrix for a binary classification problem. Columns denote the actual class (either 0 or 1) and rows denote the reported class (0 or 1). TP is true positive, FN is false negative, FP is false positive and TN is true negative. Note that $TN + FP = 1$ and $FN + TP = 1$.

process photos more quickly. Second, for photos which have already been retired, there is still room for improving upon the current accuracy. This is especially important as the accuracy of citizen science data in ecology is often questioned; to demonstrate the usefulness of such projects, we need to maximize the accuracy of our results.

Classifier Aggregation

We next review the current work done on classifier aggregation, starting with the standard formal model (Kim and Ghahramani 2012). Let the set of all users be K and the set of all subjects be I . Let J be the set of all possible distinct classes for the subjects in I . A subject in Snapshot Serengeti is a photo and an example class is $\{grantGazelle \wedge elephant\}$, *i.e.* a photo contains at least one grant Gazelle and at least one elephant. For notational simplicity, we will assume that every user views every subject (this is a straightforward assumption to relax). Let $c_i^{(k)} \in J$ be the *reported classification* of subject i according to user k . Let $t_i \in J$ be the actual class of subject i . The goal of classifier aggregation is, given $\{c_i^{(k)}\}$ for all users and subjects, to “estimate” t_i with some *aggregate classification* c_i .

In the simplest case J is isomorphic to $\{0, 1\}$ in which case we would have a *binary classification problem*. For such classification problems, a simple method for aggregating all of the users’ individual classifications into an overall classification is *majority voting* (MV) where the aggregate classification is set as (Littlestone and Warmuth 1994)

$$c_i = \begin{cases} 1 & \text{if } \sum_k c_i^{(k)} \geq |K|/2 \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

With *weighted majority voting* (WMV) a weighting factor w_k is included with each user (Littlestone and Warmuth 1994).

Dawid and Skene proposed an approach to classifier aggregation based on modeling users with *confusion matrices* (Dawid and Skene 1977). For user k , the confusion matrix $\pi^{(k)}$ contains the element $\pi_{t_i, c_i^{(k)}}^{(k)}$ which is the proba-

bility of the user reporting $c_i^{(k)}$ given t_i . An example confusion matrix for binary classifications is shown in Table 1. Dawid and Skene’s approach assumes that the actual class for each subject is chosen independently at random according to some probability distribution.

Kim and Ghahramani developed the *Independent Bayesian Classifier Combination* (IBCC) method, a

Bayesian adaptation of Dawid and Skene’s method (Kim and Ghahramani 2012). The IBCC model keeps the independent sampling assumption and the confusion matrix representation of users. Subsequent work has improved both the accuracy and computational efficiency of the IBCC model (Simpson et al. 2013; Venanzi et al. 2014).

Other approaches do not assume independent sampling or use confusion matrices. Work by Liu *et al.* and Karger *et al.* assumes that each subject is equally difficult to correctly classify (Karger, Oh, and Shah 2011; Liu, Peng, and Ihler 2012). Dalvi *et al.*’s approach requires dealing with matrices of size $|K| \cdot |I|$ (Dalvi et al. 2013). For Snapshot Serengeti, with 35,000 users and 1.5 million subjects (photos), this would result in matrices with 54 billion cells.

Challenges of Classifier Aggregation with Snapshot Serengeti

In this section, we discuss three critical assumptions made in existing literature on classifier aggregation which do not hold with Snapshot Serengeti: samples are taken independently, all photos are equally difficult to classify, and users can be modeled using a confusion matrix.

Swanson has shown that the first two assumptions, independence and equal difficulty, do not hold with Snapshot Serengeti (Swanson et al. 2014). Independence is often violated due to animals lingering in a given areas. Equal difficulty is violated when some photos show a complete, in-focus animal, while in other photos the animals may be partly out of the photo, or so close that they are out of focus and blurry. Photos taken at night are often harder to classify.

We next examine whether users can be modeled using confusion matrices. One approach to such modeling might be to consider every possible set of species which could occur in a photo. For example, we would have the class $\{wildebeest \wedge zebra\}$ for all photos containing both a wildebeest and a zebra. With 48 species, this would lead to 2^{48} different classes. Most of these classes would never occur in practice; for the photos with gold standard data available, the maximum number of species in one photo was three. However, we cannot know in advance which classes occur and which do not. A more fundamental problem is the nonsensical inferences that such an approach could lead to. For example, one element of each user’s confusion matrix would be $\pi_{(wildebeest, zebra), zebra}^{(k)}$, corresponding to a user confusing the combination of a wildebeest and a zebra with just a zebra. This is nonsensical; either the user simply missed a wildebeest in the photo or the user confused a wildebeest with a zebra.

Alternatively, we could use confusion matrices which focus on individual species. For example, the element $\pi_{wildebeest, zebra}^{(k)}$ would correspond to the probability of a user confusing a wildebeest with a zebra. To use such a confusion matrix, we would still need to understand the user’s mistake exactly. There are multiple reasons why the user might report $\{zebra\}$ instead of $\{wildebeest \wedge zebra\}$. To determine which mistake the user actually made, we need to

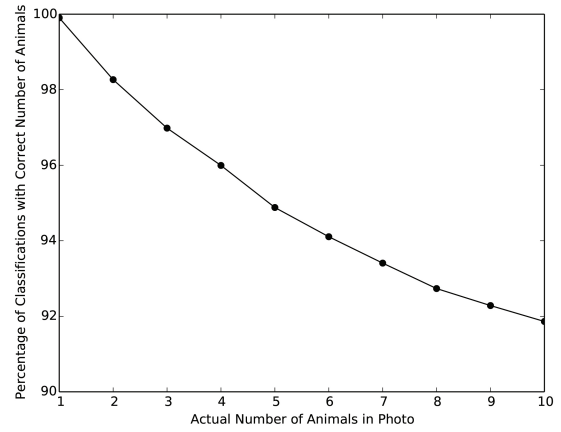


Figure 1: Percentage of classifications with the correct animal count as a function of the number of animals actually in the photo.

look the animal count for both species. If there is 1 wildebeest and 1 zebra and the user reported 1 zebra, we know that the user missed the wildebeest. If the user reported 2 zebras, we know that the user confused the wildebeest with a zebra.

Such reasoning requires users to give the correct animal count. Figure 1 shows the percentage of user classifications which give the correct animal count from the gold standard data as a function of the number of animals in the photo. We see that when there are only a few animals in a photo, users almost always give the correct count (even if they do not correctly identify the species). However, this consistency decreases as the number of animals increases. If the user’s count does not match the gold standard count, we cannot deduce what mistakes the user may have made. Users are also given the possibility of listing the number of animals as “11-50” and “51+”. In these cases, even if the user is correct, we (by design) cannot know exactly how many animals the user saw.

Furthermore, how users make mistakes is dependent on the number of animals in a photo. For example, suppose a user is shown a series of photos all containing one wildebeest (and nothing else) and correctly classifies each photo with a probability p . The user is then shown a series of photos all containing two wildebeests (and nothing else). If the user is able to classify each wildebeest with the same “ability” as before, the probability of correctly classifying each photo would improve to $1 - (1 - p)^2$.

Figure 2 shows the percentage of correct classifications for a real Snapshot Serengeti user faced with the above scenario with the number of wildebeests varied between 1 and 10. The user correctly classified 71.4% of the photos containing only 1 wildebeest. The dashed line in Figure 2 shows the expected percentage of correct classifications if the user’s classification ability was constant, *i.e.* for n wildebeests, we have $1 - 0.286^n$. For 2, 3, 4, 7, 8 and 10 wildebeests, we see that the user’s ability is roughly constant and

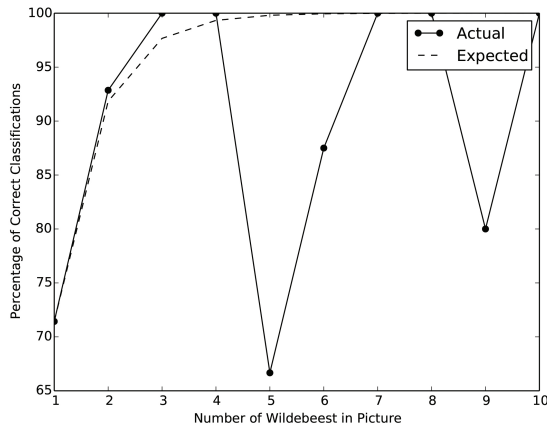


Figure 2: For a single example user, the expected versus actual percentage of correct wildebeest classifications as a function of the number of wildebeests in the photo. The expected percentage is based on the assumption that the user correctly identifies each wildebeest with a constant and independent probability.

matches our expected results. However, for 5, 6 and 9 wildebeests, the percentage of correctly classified photos is much lower than expected. In fact, the percentage of correctly classified photos for 5 and 6 wildebeests is actually lower than for 4 wildebeests. (The results are statistically significant. For 9 wildebeests, the sample size is too small to reach a conclusion.) The results are at first counterintuitive; increasing the number of wildebeests in a photo should increase the probability of classifying the photo as containing wildebeest. One possible explanation is that to fit a larger number of animals into a picture, they may, on average, have to be smaller. Additionally, a larger number of animals increases the chances that some will be partially blocked, hiding features necessary for a correct classification. Regardless of the reason, this means that the way users analyze photos with a single animal in them is not necessarily how they analyze photos with multiple animals.

Thus, we have shown that confusion matrices are not a reasonable approach to modeling users; we cannot always understand the mistakes they make, in part because the mistakes are highly dependent on many different factors.

A New Approach

In this section, we present our classifier aggregation algorithm developed for Snapshot Serengeti. We had three main goals for our algorithm. First, given the existing user data, to improve on the accuracy achieved by Snapshot Serengeti’s current aggregation algorithm. Second, to minimize the number of users needed to achieve any desired level of accuracy. This will help speed up Snapshot Serengeti in the future. Our final goal was simplicity. We needed to create an algorithm that was intuitive and clear to non AI-specialists, such as the Snapshot Serengeti ecologists.

Threshold	1	2	3	4
Error	8%	2.4%	0.2%	0%

Table 2: Estimated percentage of non-blank photos incorrectly classified as blank as a function of the threshold value set in condition 1 in Algorithm 1.

For the reasons discussed in the previous section, we discounted any approach which assumed independent sampling, equal difficulty of classification or that a user could be modeled using a confusion matrix. This left us with simple algorithms such as *majority voting* (MV), *weighted majority voting* (WMV) and Algorithm 1.

We first considered how to classify photos as blank. MV and WMV were not useful because even if only a small minority of users classify a photo as non-blank, the photo probably is non-blank. People rarely see something when there is nothing: out of all the blank photos currently retired, only 6% are retired via condition 2 in Algorithm 1 (the case where at least one user classifies a blank photo as non-blank). For such photos, there were an average of 4 non-blank classifications.

We were left with using Algorithm 1 to classify blank photos. By the reasoning above, we believe that all blank photos would have been correctly classified. Thus, we cannot improve on the accuracy of Algorithm 1 (for blank photos) and we instead considered how to improve its efficiency, for processing future photos. We could process blank photos more quickly if we reduced the threshold in condition 1 of Algorithm 1, currently set to 5. The risk is that decreasing this threshold would increase the number of non-blank photos incorrectly classified as blank. Due to the lack of gold standard data for blank photos, we do not know how many such misclassified photos currently exist. However, we can estimate how many additional misclassifications would occur if we decreased the threshold: a blank (aggregate) classification would occur if, by chance, the first n (individual) classifications for a photo were blank where n is whatever threshold we choose for condition 1. We examined 500 non-blank photos and counted the number of “initial” blank classifications (*i.e.* those which occurred before any non-blank classifications). The resulting distribution is shown in Table 2. We see, for example, that if we were to decrease the threshold to 2, an estimated 2.4% of photos currently classified as non-blank would be accidentally classified as blank. There is concern that such photos would disproportionately feature rare animals; users might be less confident classifying species they are less familiar with. Thus, to err on the side of caution, we selected a threshold of 4.

We next considered how to classify non-blank photos. Our aim was to improve upon both the accuracy and efficiency of Algorithm 1. We started by using MV (Equation 1) with each species independently. For example, we labelled a photo as containing a zebra if at least 50% of the users found a zebra in that photo. Figure 3 shows the accuracy of this approach as a function of the number of users per photo. While the accuracy of MV improves with each additional user, the maximum accuracy of 96.0% is still slightly below the ac-

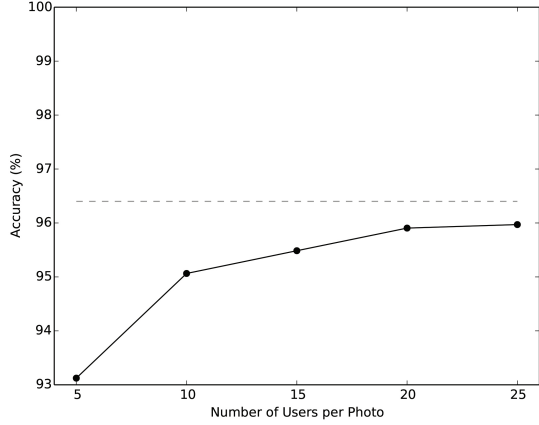


Figure 3: The accuracy of majority voting (% of images correctly classified) as a function of the number of user classifications per photo. The dashed line indicates the accuracy achieved with the current Snapshot Serengeti algorithm (Algorithm 1).

curacy of 96.4% achieved by Algorithm 1 (also shown in Figure 3).

To see if a WMV based approach could be more successful, we first divided the photos into two sets; those that were *easy* to correctly classify and those that were *hard* to classify. A photo was easy if it was correctly classified by MV and hard if it was incorrectly classified by MV. There was a positive correlation between the percentage of easy photos a user correctly classified and the percentage of hard photos that user correctly classified (Pearson correlation coefficient of 0.148). Users who classified fewer than 4 hard photos may have just been “lucky”; if we exclude those users, we get a significantly increased correlation (a Pearson coefficient of 0.401). This means that hard photos are not fundamentally different from easy ones. These results suggest that we could improve the accuracy of the classification of hard photos by using WMV to give more weight to those users who are good at classifying easy photos. Since we do not have gold standard data for all photos, in general we cannot be sure if a photo is easy or hard. Given the high accuracy of MV, a reasonable approximation would be to assume MV correctly classifies all photos, *i.e.* all photos are easy.

Instead of having one overall measurement of each user’s accuracy, we created a species specific metric; a user who is highly accurate at classifying elephants is not necessarily as accurate at classifying gazelles. An initial choice for user k and species j was

$$w_k(j) = \frac{TP + TN}{TP + FP + FN + TN}. \quad (2)$$

All of the values on the RHS of Equation 2 are with respect to both k and j : we omit the subscripts for notational clarity. For species j and user k , Equation 2 is the number of correct classifications over the total number of classifications. However, the negative occurrences will dominate the posi-

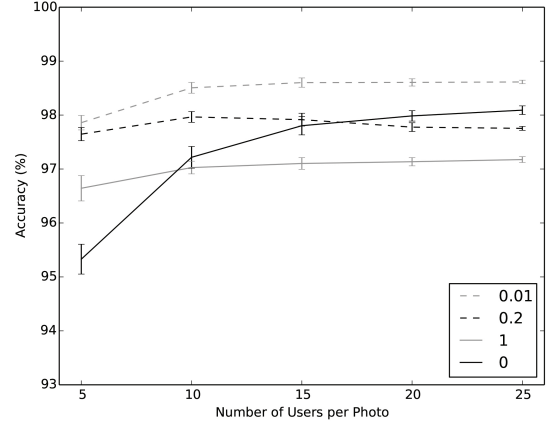


Figure 4: Average accuracy of Equation 4 as a function of β in Equation 3.

tive ones as most photos will not contain a given species and true negatives are a fairly easy outcome; people realize that most photos do not contain a given species. Thus, the easy cases will dominate the harder cases (*i.e.* the true positives) which are a better measure of a user’s accuracy. We found a more useful measurement of a user’s ability to be

$$w_k(j) = \frac{TP + \beta TN}{TP + FP + FN + \beta TN}, \quad (3)$$

where $\beta \in [0, 1]$ scales the contribution from the true negatives. Thus $c_i(j)$, which indicates whether species j is in photograph i , is set as:

$$c_i(j) = \begin{cases} 1 & \text{if } \sum_k w_k(j)c_i^{(k)} \geq |K|/2 \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

We next consider the optimal value for β . Figure 4 shows the accuracy resulting from setting β to 0, 0.01, 0.2 or 1, as a function of the number of user classifications per photo. For all numbers of classifications per photo, $\beta = 0.01$ achieved the highest accuracy. In fact, with only 5 users, $\beta = 0.01$ results in an accuracy of 97.5%, higher than the accuracy of Algorithm 1 in Figure 3. Based on these results, we chose $\beta = 0.01$ for our algorithm.

A standard method for increasing the accuracy of aggregation algorithms is to provide gold standard data for some of the subjects. We tested this approach using 5-fold cross validation, *i.e.* we provided gold standard data for one fifth of the photos and tested the accuracy of our algorithm on the remaining photos. We then repeated this process 4 more times, each time using a different fifth of the photos. The results (not shown) did not show any improvement, due to the fact that MV already performed well enough that the additional information provided relatively little benefit. Another standard technique would be to use an iterative approach and repeatedly update users’ weights. This technique (results not shown) also did not show any improvement since the users’ weights did not change after the first iteration.

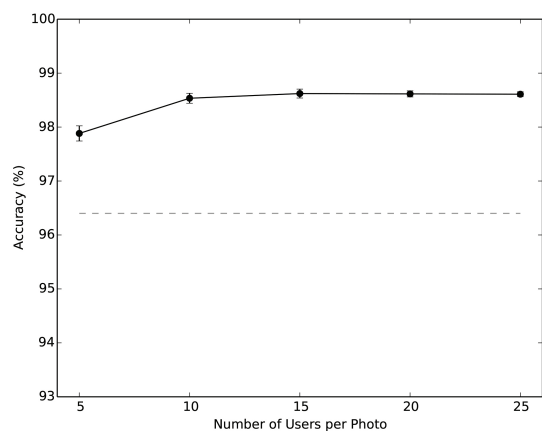


Figure 5: Average accuracy for our aggregation algorithm, Algorithm 2, as a function of the number of user classifications per photo. The dashed line indicates the accuracy achieved with the current Snapshot Serengeti algorithm (Algorithm 1).

Algorithm 2 Our new aggregation algorithm.

```

if first 3 classifications are blank then
    return blank
else if after 10 classification then
    return Classification according to Equation 4
end if

```

Algorithm 2 shows our end algorithm, which classifies both blank and non-blank photos. Figure 5 shows the accuracy of Algorithm 2 with $\beta = 0.01$ against the accuracy of the current Snapshot Serengeti algorithm (Algorithm 1). Even with only 5 classifications per photo, our algorithm significantly outperforms Algorithm 1. With 10 users, our algorithm achieves an accuracy of 98.6%. More than 10 users did not result in any noticeable increase in accuracy. Based on this result, we recommend setting the number of users per photo to 10 for non-blank photos. With Algorithm 2, we can reduce the average number of users needed per photo to 5.8 while increasing the accuracy for non-blank photos to 98.6%. Our algorithm is more accurate (on average) than the experts who helped create the gold standard data.²

Conclusion

Camera traps are revolutionizing ecology while providing large amounts of raw data (images). The Serengeti Lion Project is using camera traps to study animals in the Serengeti. The Project, along with Zooniverse, created Snapshot Serengeti to get volunteers to help process this data. Since volunteers may make mistakes, each image is shown to multiple users and a critical task is to combine these indi-

²The gold standard data was created by aggregating, by hand, the classifications from 7 individual experts who, on average, had an accuracy of 97.7% (Kosmala 2013).

vidual classifications into aggregate classifications. We have developed a new aggregation algorithm which is more efficient (*i.e.* requires fewer users per photo) and more accurate than the algorithm currently used by Snapshot Serengeti. Our algorithm is even more accurate on average than individual experts.

The possibility of applying our algorithm to other citizen science projects analyzing camera trap studies is exciting. We are also interested in further improving our algorithm's efficiency and accuracy. In particular, to increase efficiency, we would like to retire blank photos more quickly. To increase accuracy we would like to take advantage of the lack of independent sampling; *e.g.* if a photo contains a zebra, the probability that the next photo will also contain a zebra is increased.

References

- Dalvi, N.; Dasgupta, A.; Kumar, R.; and Rastogi, V. 2013. Aggregating crowdsourced binary ratings. In *Proceedings of the Twenty Second International Conference on World Wide Web*, WWW '13.
- Dawid, A. P., and Skene, A. M. 1977. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*.
- Karger, D. R.; Oh, S.; and Shah, D. 2011. Iterative learning for reliable crowdsourcing systems. In *Advances in Neural Information Processing Systems* 24. 1953–1961.
- Kim, H.-C., and Ghahramani, Z. 2012. Bayesian classifier combination. In *Proceedings of Fifteenth International Conference on Artificial Intelligence and Statistics*, AISTATS '12.
- Kosmala, M. 2013. Summary of the experts. <http://blog.snapshotserengeti.org/2013/10/16/summary-of-the-experts/>.
- Littlestone, N., and Warmuth, M. K. 1994. The weighted majority algorithm. *Information and Computation*.
- Liu, Q.; Peng, J.; and Ihler, A. 2012. Variational inference for crowdsourcing. In *Advances in Neural Information Processing Systems* 25. 692–700.
- Simpson, E.; Roberts, S.; Psorakis, I.; and Smith, A. 2013. Dynamic bayesian combination of multiple imperfect classifiers. In *Decision Making and Imperfection*, volume 474 of *Studies in Computational Intelligence*. 1–35.
- Swanson, A.; Kosmala, M.; Lintott, C.; Smith, A.; Simpson, R.; Lynn, S.; Strauss, M.; and Packer, C. 2014. Snapshot Serengeti: A case study in camera traps and citizen science for multi-species monitoring. *Ecology and Evolution*. In review.
- Venanzi, M.; Guiver, J.; Kazai, G.; Kohli, P.; and Shokouhi, M. 2014. Community-based bayesian aggregation models for crowdsourcing. In *Proceedings of the Twenty Third International Conference on World Wide Web*, WWW '14.
- Zooniverse. 2014a. www.snapshotserengeti.org.
- Zooniverse. 2014b. www.zooniverse.org.