SealNet 2.0: Automating for a Long-term Viable Solution

James Daus and Nicholas Diebold

Abstract

The harbor seal is a protected marine mammal constantly subject to human interaction, causing biologists to desire more advanced research methods to accurately study the species' social behavior, movement, and life history traits. Current methodologies are expensive, harmful, or produce biased results. Deep learning facial recognition programs using convolutional neural networks (CNNs) have shown to be extremely accurate for primates and have recently been optimized for non-primates without pelagic markings. SealNet was created based on the PrimNet program to assist biologists in their study of harbor seals and achieved an identification accuracy of 95.3%. The program was not ready for use as it did not significantly outperform PrimNet and was too inefficient and specific. Dozens of hours of manual labor were required for cropping and labelling of the research photos, and the software was set up to only classified already grouped seals. This paper discusses SealNet 2.0 which is much more usable than its predecessor. The face detection and cropping process has been automated by an additional CNN, and several scripts have been added to allow new, unseen seals to be classified based on past data, saving many hours of work and allowing for new kinds of studies. The utility of SealNet 2.0 is shown through a small ecological study using the data collected for training the model. SealNet 2.0 is a very promising program that will only need a few minor updates before it can be successfully used in the field.

1. Introduction

Long-term, large-scale ecological studies are extremely important tools to examine animal life history traits, social behavior, population dynamics, and extinction prevention (Clutton-Brock & Sheldon 2010). Current research methods in this field are outdated and suboptimal and must be updated to maximize efficiency and accuracy. Collecting data manually is unlikely to produce extensive results as it can only work on a small scale (Cunningham 2009). Capturing the animal and attaching GPS tags or trackers fix this issue, but is economically impractical as they can cost $400-$4000 per device ("GPS and VHF Tracking Collars Used for Wildlife Monitoring" 2017). The capturing process also interferes with normal social behavior and can be difficult for the researcher. Most importantly, after attachment the tag can cause injury or even death as a result of mobility and behavioral issues (Rosen, Gerlinsky & Trites 2018). The solution to this problem was to explore facial recognition software as an efficient, low-cost, large-scale method that is not intrusive.

Harbor seals are protected marine mammals constantly being studied by ecologists. They are hard to survey as they exhibit dynamic movement patterns (DiGiovanni et. al 2011, Honeywell & Maher 2017), but further research on their behavior has been called for as they inhabit locations with a high prevalence of human interaction. Such studies have already revealed that harbor seals show both intra- and interspecific competition based on population size and time of year (Murray 2008, Honeywell & Maher 2017). They have also been important in the identification of molting and pupping sites as well as harmful changes in life history traits that could endanger a specific population. (Harris, Lelli & Gupta 2008, Bowen, Ellis & Iverson 2003).

The research methods for harbor seal studies suffer from the same general setbacks as previously discussed. Visual and aerial surveys require too much time and money. Even capture-recapture methods result in too much bias and are very invasive (Cunningham 2009). Satellite bands are the most common tools used for tracking seals (Perras & Nebel 2012), however these have been shown to affect swim speed, mobility, oxygen consumption, and metabolic rates, making them more vulnerable to predation (Rosen, Gerlinsky & Trites 2018). Despite the accuracy of the individual tags, they often fall off resulting in suboptimal data collection (Arnason & Mills 2011). Manual photographic matching has been tried as harbor seal coat patterns do not change, however the coat colors fluctuate depending on the time of year and thus results are biased and inaccurate (Cunningham 2009).

A facial recognition solution for harbor seals has been called for to fix these issues (Cunningham 2009). Deep learning programs using convolutional neural networks have been developed mainly for primates as they are most similar to humans, and have seen much success. PrimNet (Deb et. al 2008) modified FaceNet (Schroff 2015) - which recognized human faces - and boasted a lemur identification accuracy of 93.75%. Crouse (2017) achieved an accuracy of 98.7% with lemurs, and Schofield (2019) used a program on chimpanzees with 92.5% accuracy. New programs have been created for animals in the Ursidae family. Chen (2020) identified pandas with 97.27% accuracy, however recognition of pandas is much easier due their distinct, pelagic facial markings. BearID (Clapham 2020) was the first to attempt the identification of a non-primate without pelagic markings. While it only resulted in a complete pipeline accuracy of 82.4%, it contained an extremely accurate face chipping method that greatly reduced the amount of manual labor required in raw photo pre-processing.

The goal of this research was to improve upon SealNet, an existing CNN-based seal facial recognition software based directly off of PrimNet. PrimNet underwent slight changes by adding channel shuffling and group convolution while using 6 convolutional layers instead of 4. SealNet identified seal faces in 1:1 verification and closed set verification with an accuracy of 95.3%, which is excellent but insignificant to running the same data through the original PrimNet software (94.7% accuracy). An additional CNN using the Dlib machine learning libraries for C++, originally trained for detecting dog faces, was adapted for use as a seal facial detector, and was paired with Dlib functions for cropping and resizing photos; we call this addition SealFindr. SealFindr allows for a significant automation of work, as in situations where research assistants needed to sift through hundreds of photos to find and crop seal faces, the program can now do it in one call. Folders of hundreds of raw 4K resolution field photos can now be turned into

112x112 pixel seal face chips ready for classification in a matter of hours. Additionally, several scripts have been added to the original SealNet pipeline, which automate the process of creating splits for new, unclassified seal chips to be classified based on a subsection of past data, allowing for a large level of customizability for any variety of studies.

SealNet 2.0 is used in a small ecological study on yearly seal migration patterns using the data collected for the training of the program to provide a proof of concept and to demonstrate how useful it can be for future research.

2. Methods

2.1. Data Collection

Photographs were collected at Casco Bay in Harpswell, Maine in 2019 and 2020. The 2019 data consisted of 1102 raw photographs which resulted in 489 112x112 manually-cropped face chips of 77 different seal individuals. 30 of these seal individuals were collected from photos from a different source in England. These photos were taken at four different haul-out sites across five dates in July (Figure 1). The 2020 data consisted of over 1500 raw photos which yielded ~550 112x112 automated chips (via SealFindr) and ~80 seal individuals. Photos were taken at three haul-out sites across five dates in January. Different locations were photographed for no reason other than the presence of seals on that particular day. No locations were photographed in both 2019 and 2020.

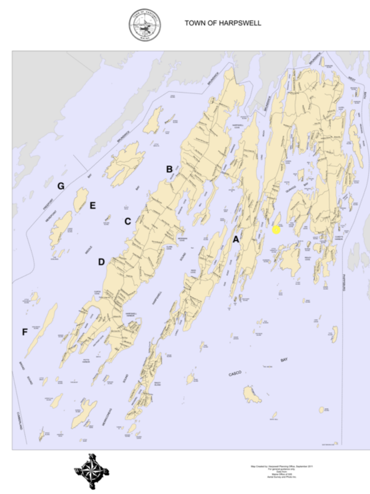


Figure 1. Locations of Haul-Out Sites in 2019 and 2020.

A — Seal Rock (2019)
B — Wilson Cove (2019)
C — Brandt Ledges (2019)
D — Mitchell Field (2020)
E — Branning Ledge (2020)
F — Whaleboat (2020)
G — Bustins Ledge (2019)

2.2. Changes to SealNet 1.0

<u>2.2.1 SealFindr</u>

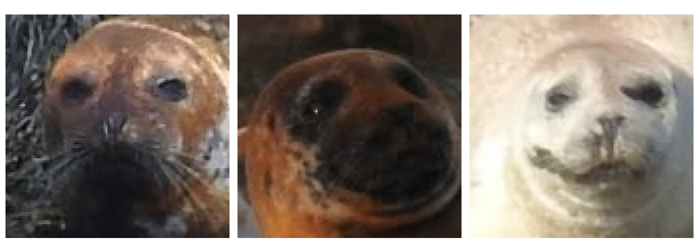


Figure 2: SealFindr chips created from 2020 data

An additional program, SealFindr, was created to aid in the seal classification pipeline. SealFindr takes as input raw photos of any size and any common photo format, and searches them for seal faces. Once detected, the program scales the face up or down and crops a 112 pixel by 112 pixel box containing the face, which is then exported into a new .jpeg file, ready to be used by a CNN-based classifier model.

A shell script called sealFindr.py is called from the command line with the name of a folder containing any number of photos, and a loop will commence which iterates through each photo in the folder. First, the photos are expanded up to a maximum size using a try-catch loop. This expansion of resolution allows for more accuracy from the CNN used to detect faces. In testing, a 4x expansion of input photos resulted in more than double the number of chips found, as can be seen in the following table.

Total Photos Used	100
Without expansion	28 Chips Found
With expansion	61 Chips Found (2.18x)

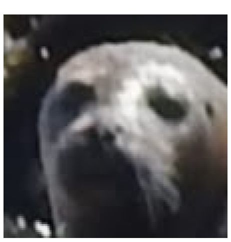
Table 1: A subset of 100 photos across each location and both years was tested using the final SealFindr program, with and without a 4x resolution expansion: 2x in each dimension.

Next, the trained facial detection model contained in the GitHub, which is originally from Dlib, called seal.dat, will detect matches for seal faces using a moving window CNN. These positive matches of the CNN are iterated through, allowing several seal faces to be detected per photo, which was an important focus as seals are often present in large groups. Next, using function calls from the Dlib libraries, the photo is manipulated so that each face respectively fits into a properly sized 112 pixel by 112 pixel square chip, which is another important feature as this size is required by the SealNet CNN. Finally, the chips are saved into a new folder in the location the script is located, and the loop continues.

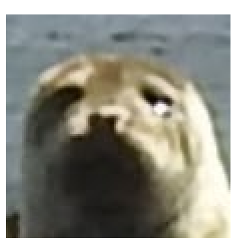
1. Raw 4608x3456 pixels photo is expanded using pyramid_up Dlib function

2. Faces are detected in new 9216x6912 pixels photo using moving window CNN

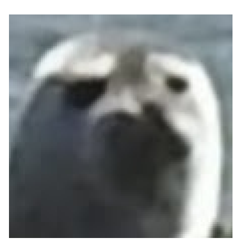
3. Each face is expanded or downsized to fit 112x112 pixels, then saved to .jpeg



/BrandtChips/Chip0.jpeg



/BrandtChips/Chip1.jpeg



/BrandtChips/Chip2.jpeg

Figure 3: SealFindr process visualized

2.2.2 SealNet Additions

In addition to the SealFindr pipeline, scripts were added to be used in tandem with the existing SealNet code (found at https://github.com/aylab/SealFaceRecognition). Two scripts were added which enable easy usage of the SealNet 1.0 architecture and model on new, unclassified data.

The first script, format_data.py, allows for automated creation of required .txt files for SealNet 1.0. When called, the script can take a folder of labelled seal chip to use as references. These references are the list of all possible classifications for incoming data, and any new unclassified chip will be matched to its closest counterpart, according to the trained SealNet CNN classifier. This script also automates the creation of a list of probe photos, by taking a folder of unlabelled chips, and preparing them to be fed into the classifier. Researchers can now simply run this script to set up their list of reference seals, and format their new unclassified chips, which can be used by the next script to finish the process.

The second script, seenBefore.py, takes as input the reference seals and seals to be classified, and uses the already trained SealNet model to find a best match for each of the unlabelled seal chips. These matches are found by projecting the chips into 128-dimensional space, and then finding the nearest neighbor out of all reference photo groupings. Finally, an easily usable, simply formatted result text file is created, which lists each photo with its nearest match.

All of these additions can be found in the SealNet 2.0 GitHub, found here: https://github.com/jamesdaus/SealNet2

2.3. Ecological Study

2.3.1. Data Collection

Data on the yearly migration patterns of harbor seals was obtained by running the new 2020 automated chips through the SealNet classifier trained on the 2019 data. Only half of the 2020 chips were used due to time constraints. These chips came from Branning Ledges (1/10) and Mitchell Field (1/31). SealNet returned the file number of the 2019 seal that most likely matched the 2020 face chip. Only seals 1-47 (2019) were used as the rest came from photos in England. Confidence levels for predicted matches were all above 80%. Each match was manually checked and rated on a scale from 1-3 (1: Definitely not a match, 2: Maybe a match, 3: very likely a match) with ratings of 2 or 3 warranting subsequent checks. A successful match would mean that one of the 47 seals seen in 2019 was present in a different location in 2020 (as no haul-out site was photographed both years).

2.3.2. Data Analysis

Chi-square tests were used to find the significance of assumptions about yearly migration and social behavior based on the matches found. Significance was calculated through the following equation:

$$X^2_c = \sum(O_i - E_i)^2/E_i \,.$$

Population was estimated using the mark-recapture method as shown in the following equation:

$$P = MC/R.$$

M is the number of seals photographed in 2019 (47), C is one half of the number of seals photographed in 2020 (40), and R is the number of matches (13).

3. Results

3.1. Accuracy

Last year, the SealNet classifier and its parent model PrimNet were run 10 times and evaluated based on their accuracy at rank-1 identification of seal individuals, using the 2019 dataset. The data was randomly split by 80-20 into training and testing data, and these splits were used to train and test both CNNs. Because no significant changes were made to the classifier between years, the data still represents the accuracy of the model. The following table shows the results of the testing data per test run.

Run #	SealNet Correct	PrimNet Correct
1	**18/19**	17/19
2	**18/19**	17/19
3	**18/19**	**18/19**
4	**19/19**	18/19
5	18/19	**19/19**
6	**18/19**	**18/19**

7	16/19	**18/19**
8	**19/19**	18/19
9	**18/19**	**18/19**
10	**19/19**	**19/19**
Total Accuracy	**95.3%**	94.7%

Table 2: SealNet and PrimNet classification accuracies
Ttest_indResult(statistic=0.2873478855663523, pvalue=0.7771278487505222)

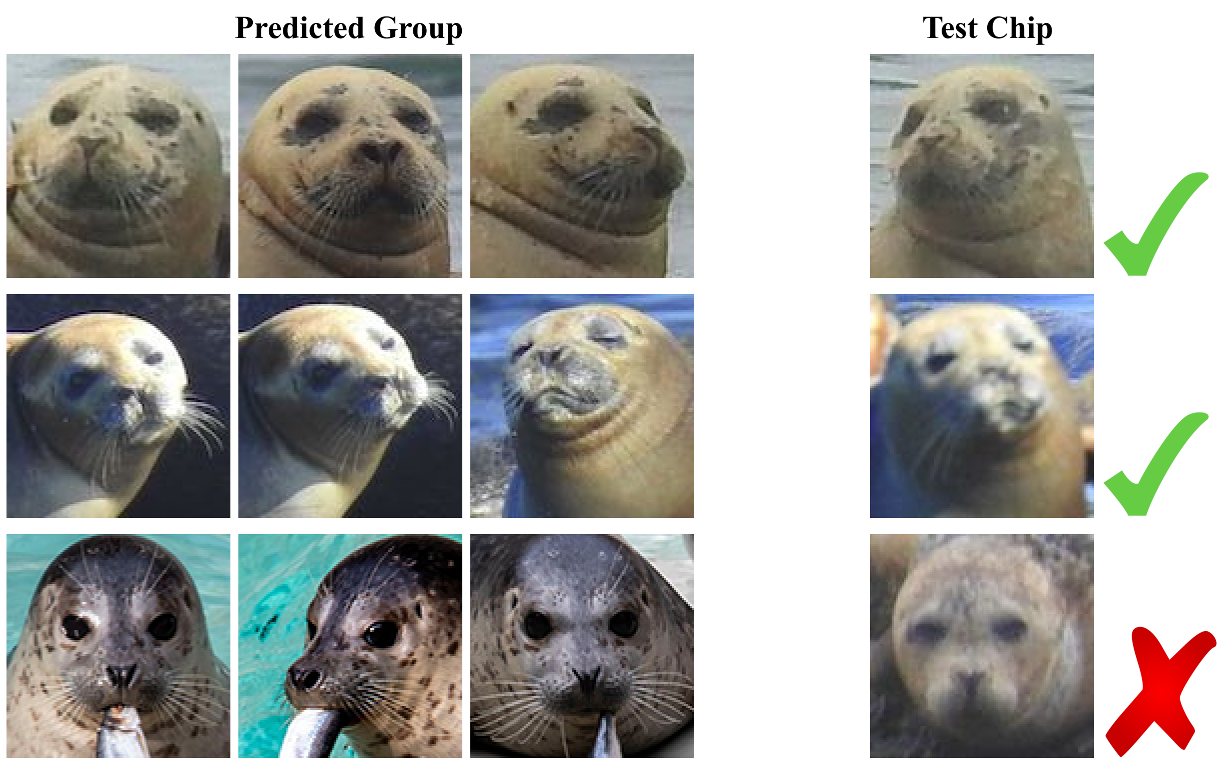


Figure 4: Example 2019 classifications of manually chipped testing data

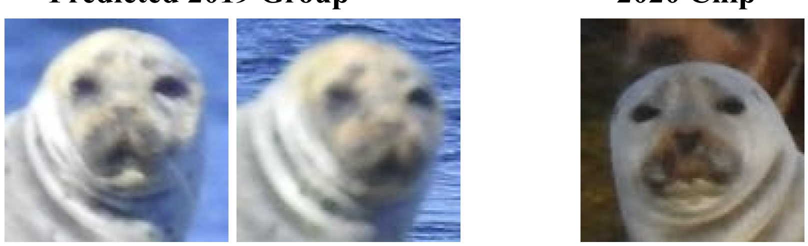
Pipeline Element	Result	Accuracy
SealFindr	**548/550 Valid Faces**	**99.6%**

SealNet 1.0	**181/190 Across Runs**	**95.3%**
Expected SealNet 2.0	**N/A**	**94.9%**

Figure 5: SealNet 2.0 pipeline accuracy; the final row represents the expected accuracy for a previously seen 2019 seal to be correctly classified when seen in 2020, found by combining loss in both stages.

Figure 6: Sample of 550 chips created from 2020 raw photos, including the only two failed chips

Predicted 2019 Group **2020 Chip**



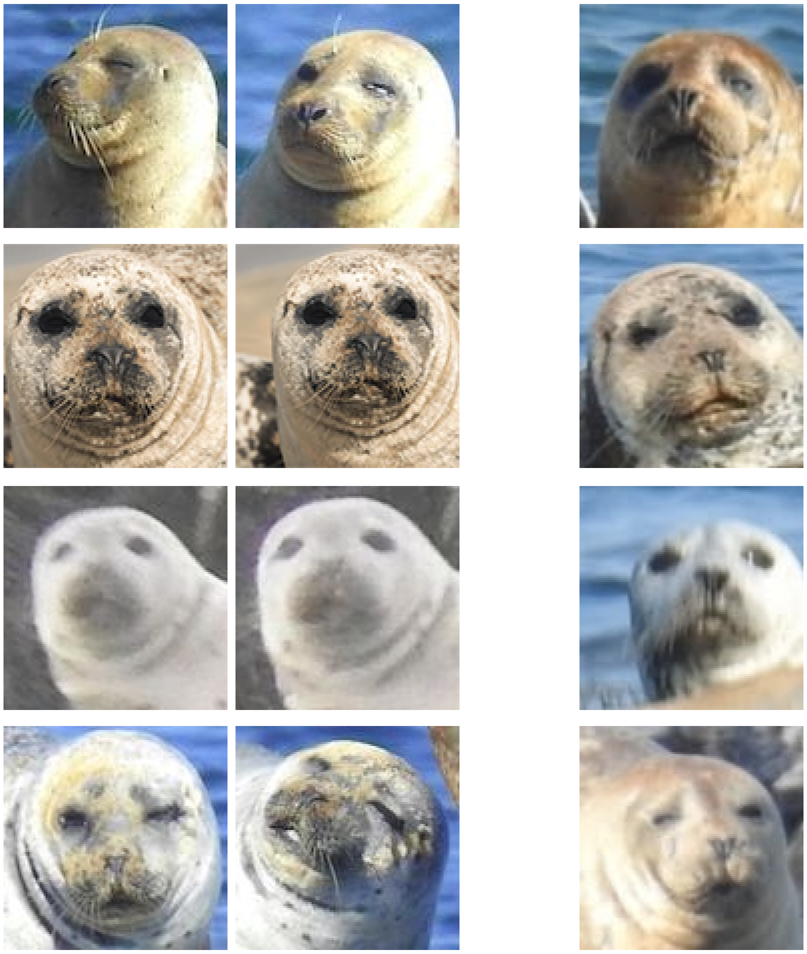


Figure 7: Variety of 2020 classifications

3.2 Ecological Study

13 matches were found when running half of the 2020 data through the 2019 classifier despite consistently high accuracies (Table 3). This means that 13/47 seals present in 2019 were found in 2020 at different locations; more would have likely been found if the other half of the 2020 data was used. If the assumption is made that another 13 matches would have been found in the other half which photographed different locations, 55% (26/47) of the 2019 seals would have been found at different haul-out sites in 2020.

2020 Location	2020 Date	2019 File Number	2019 Date	2019 Location
Branning	10-Jan	4	20-Jul	Brandt
Branning	10-Jan	26	30-Jul	Bustins Ledge
Branning	10-Jan	41	24-Jul	Seal Rock
Branning	10-Jan	23	27-Jul	Wilson Cove
Mitchell	31-Jan	17	20-Jul	Brandt
Mitchell	31-Jan	8	20-Jul	Brandt
Mitchell	31-Jan	1	20-Jul	Brandt
Mitchell	31-Jan	45	24-Jul	Seal Rock
Mitchell	31-Jan	9	20-Jul	Brandt
Mitchell	31-Jan	14	20-Jul	Brandt
Mitchell	31-Jan	40	24-Jul	Seal Rock
Mitchell	31-Jan	2	20-Jul	Brandt
Mitchell	31-Jan	42	24-Jul	Seal Rock

Table 3. Locations and dates of the 13 harbor seals present in 2019 that were photographed in 2020. Matches were manually checked to ensure accuracy.

This suggests that harbor seals exhibit high migratory site fidelity to Casco Bay, but a low local migratory site fidelity (to their haul-out site). If seals showed a high local migratory site fidelity, no matches would have been found as no location was photographed across both years.

The Seal Rock haul-out site is far from the rest of the haul-out sites (Figure 1). The fact that 4/13 seals migrated from Seal Rock was unexpected. The 13 seals showed no preference when choosing a "near" or a "far" migration (chisq = .69, p = .41, NS). "Far" migrations were those from Seal Rock (2019) to any 2020 location as well as from Wilson's Cove (2019) to Branning Ledges (2020). This suggests that distance was not a deciding factor within Casco Bay.

It was also evident that seals seen in the same location and date in 2019 were also seen in 2020 at the same date and location (e.g. seals with file numbers 40,42, & 45). Analysis was performed on the number of seals that started in the same location/date and migrated to the same location/date against how many of those original seals ended up at a different location. Seals 23 and 26 were excluded as they were the only ones from their respective haul-out sites. Seals were more likely to stay in social groups than to migrate on their own (chisq = 4.45, p = .035).

The population size was also calculated for Casco Bay using the mark-recapture method and resulted in an estimation of 145 harbor seals.

4. Discussion

SealNet 2.0 took the basic idea of the original SealNet program to the next level. SealNet performed well identifying seals that had already been classified, however this is more impractical for field use and harder to use at a large scale. SealNet also required an overwhelming amount of manual labor in chipping the photos. The SealFindr CNN solved the chipping problem by automating detection and cropping of seal faces in a photo through just one call. It only takes a few hours for hundreds of 4K photos to be processed, whereas it previously took days or weeks. The SealNet 2.0 pipeline also includes additional scripts which allow for the classification of new and unclassified chips based on previous data. This gives SealNet 2.0 a much larger flexibility in the types of studies able to be performed, and makes it much easier for researchers to work with new data and build a long-term data source. While manual checking is still necessary, this is due to the dramatic coat color fluctuations displayed by the harbor seal throughout the year and not due to the inefficiencies of the new script. SealNet 2.0 can still efficiently turn new photographs into easily-checkable classifications of seals seen previously, and thus overall manual involvement is minimal. An additional appeal of the SealNet 2.0

pipeline is that it is an evolving model; once checkable outputs are created and verified, these new confirmed classifications can be added back into the pipeline as new reference photos, adding to the robustness of the network over time. As years pass, and the same seals are identified, their new chips can be added into their existing reference folder, and new seals which were not seen can be given new reference folders, thus allowing the system to grow in both accuracy and scope.

The versatility of SealNet 2.0 has been shown through a brief ecological study using a subset of the data. The study was the proof of concept and was not meant to find any significant or remarkable results. This is because the data was not originally collected for ecological analysis, only the training and testing of SealNet. Nonetheless, SealNet 2.0 was able to easily provide information on yearly migration patterns, social behavior, and population size. The significance of this cannot be understated; SealNet 2.0 would perform incredibly well with more data that was collected with the intent to test a previously-determined hypothesis.

There are some notes that a researcher using this program should keep in mind when collecting data. It is important to take many photos of the same seals as photo quality and seal head orientation are not always guaranteed. With SealFindr, taking more photos does not substantially slow down the chipping process as it did with the original SealNet. If processing speed is an issue, an individual can crop the land and water surrounding seals to reduce the file size, though if researchers have access to servers, SealFindr can simply be run in the background for several days and nights to handle even very large amounts of data. Another small effort is that researchers should go back into their references and add the newly classified chips to the corresponding folders if correct, or create new reference folders for new seals. This manual work however is extremely less time consuming than the original fully-manual chipping and classifying method.

Despite the success of SealNet 2.0 and the vast improvements it has undergone since its original creation, SealNet 2.0 could still be optimized further. To create an even smoother process for researchers, SealNet 2.0 could include confidence cutoffs so that seals which likely do not have a match in the references are listed as such, rather than being paired to a poor match within the references, which currently demands consideration even if the network is not very confident in the result. In addition, if there are two very similarly ranked potential classifications, only the top choice is shown, which may cover up the correct choice that had a marginally lower confidence. To fix this, up to three potential choices could be listed per seal, assuming they are all above some tunable model confidence cutoff. These additional features would allow researchers to add data year after year, knowing they are getting the best classifications and avoiding oversights.

References:

Arnason, A, and K Mills. Bias and Loss of Precision Due to Tag Loss in Jolly–Seber Estimates for Mark–Recapture Experiments. Canadian Journal of Fisheries and Aquatic Sciences, Apr. 2011.

Blundell GM, Pendleton GW. (2015). Factors Affecting Haul-Out Behavior of Harbor Seals (*Phoca vitulina*) in Tidewater Glacier Inlets in Alaska: Can Tourism Vessels and Seals Coexist? PLoS ONE 10(5): e0125486. https://doi.org/10.1371/journal.pone.0125486

Bowen WD, Ellis SL, Iverson SJ, Boness DJ. Maternal and newborn life-history traits during periods of contrasting population trends: implications for explaining the decline of harbour seals (Phoca vitulina), on Sable Island. J Zool. 2003;261(2):155–63.

Chen, P., Swarup, P., Matkowski, W. M., Kong, A. W. K., Han, S., Zhang, Z., & Rong, H. (2020). A study on giant panda recognition based on images of a large proportion of captive pandas. Ecology and Evolution, 10(7), 3561– 3573. https://doi.org/10.1002/ece3.6152

Clapham, M, Miller, E, Nguyen, M, Darimont, CT. Automated facial recognition for wildlife that lack unique markings: A deep learning approach for brown bears. Ecol. Evol. 2020; 10: 12883–12892. https://doi.org/10.1002/ece3.6840

Clutton-Brock, T.,Sheldon,B. C. 2010. Individuals and populations: The role of long-term, individual-based studies of animals in ecology and evolutionary biology. Trends Ecol. Evol. 25, 562–573

Crouse, David, et al. “LemurFaceID: Using Facial Recognition Software to Identify Lemurs.” BMC, BMC Zoology, 2017, www.biomedcentral.com/about/press-centre/science-press-releases/17-02-17.

Cunningham, Louise. Using Computer-Assisted Photo-Identification and Capture-Recapture Techniques to Monitor the Conservation Status of Harbour Seals (Phoca Vitulina). Aquatic Mammals, Mar. 2009, www.researchgate.net/publication/.

Deb, Debayan, et al. Face Recognition: Primates in the Wild. 2018 IEEE 9th International Conference on Biometrics Theory, Applications and Systems (BTAS), 24 Apr. 2018, arxiv.org/abs/1804.08790v1.

DiGiovanni R, Wood SA, Waring GT, Chaillet A, Jospehson E. (2011). Trends in harbor and gray seal counts and habitat use at southern New England and Long

Island index sites. 19th Biennial Conference on the Biology of Marine Mammals.

Dubey, Shiv Ram (2020). ArXiv. A Decade Survey of Content Based Image Retrieval using Deep Learning. arxiv.org/abs/2012.00641.

"GPS and VHF Tracking Collars Used for Wildlife Monitoring." Wildlife ACT, 16 May 2017, wildlifeact.com/blog/gps-and-vhf-tracking-collars-used-for-wildlife-monitoring/.

Harris, D., Lelli, B., & Gupta, S. (2003). Long-Term Observations of a Harbor Seal Haul-out Site in a Protected Cove in Casco Bay, Gulf of Maine. *Northeastern Naturalist, 10*(2), 141-148. doi:10.2307/3858283

Honeywell, A., Maher, C. (2017). Intensity, rate, and outcome of agonistic interactions in harbor seals (*Phoca vitulina concolor*) vary with density on haul-out ledges, *Journal of Mammalogy*, 98(1): 135–142, https://doi.org/10.1093/jmammal/gyw155

Kenney, M.K. 1994. Harbor seal population trends and habitat use in Maine. M.Sc. Thesis. University of Maine, Orono, ME. 55 pp

King, David E. (2009). Dlib-ml: A Machine Learning Toolkit. Journal of Machine Learning Research 10 (pp. 1755-1758).

Liu, et al. "SphereFace: Deep Hypersphere Embedding for Face Recognition." ArXiv.org, 29 Jan. 2018, arxiv.org/abs/1704.08063.

Ming-Yu Liu, Thomas Breuel, Jan Kautz. Unsupervised Image-to-Image Translation Networks. 31st Conference on Neural Information Processing Systems (NIPS 2017), 4 Dec. 2017, https://arxiv.org/pdf/1703.00848.pdf.

Montgomery, Robert & Ver Hoef, Jay & Boveng, P. (2007). Spatial Modeling of Haul-out Site Use by Harbor Seals in Cook Inlet, Alaska. Marine Ecology-progress Series - MAR ECOL-PROGR SER. 341. 257-264

Murray, M.J. Behavioral interactions between harbor seals (Phoca vitulina) and gray seals (Halichoerus grypus) on Cape Cod, Massachusetts. 2008.

Perras, M. & Nebel, S. (2012) Satellite Telemetry and its Impact on the Study of Animal Migration. Nature Education Knowledge 3(12):4

Rosen, D.A.S., Gerlinsky, C.G. and Trites, A.W. (2018). Telemetry tags increase the costs of swimming in northern fur seals, Callorhinus ursinus. Mar. Mam. Sci., 34: 385-402.

Schofield, D., Nagrani, A., Zisserman, A., Hayashi, M., Matsuzawa, T., Biro, D., & Carvalho, S. (2019). Chimpanzee face recognition from videos in the wild using deep learning. Science Advances, 5(9), 1–10. https://doi.org/10.1126/sciadv.aaw0736

Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (pp. 815–823). https://doi.org/10.1109/CVPR.2015.7298682