Research Proposal

Seal photo-identification database conversion

Matej Kucera

April 20, 2023

Abstract

Photo-identification of seals is important for population monitoring purposes in animal research centres. ExtractCompare is an outdated photo-id software which is still in use, however it lacks modern features and no longer works as intended in some cases. The proposed solution is to create software which can convert ExtractCompare based databases into modern software. The software best suited for photo-identification of animals is IBEIS. Therefore, it is proposed to create software which can convert an ExtractCompare database into an IBEIS database using the API provided by IBEIS. The conversion tool should ideally work for any species supported by ExtractCompare, not just seals.

1 Introduction

In animal research centres which focus on releasing animals back into the wild, it is essential to be able to identify animals uniquely for tracking purposes. This is useful not just to monitor population size, but also survival rates of released animals and other important metrics. For animals which can be identified visually, artificial marking (tags, chips, etc.) is not needed and it is much simpler to identify them from photos. This is also better for the animals as it is much less invasive. There are many different applications which allow photo-identification of various animals.

The Zeehondencentrum Pieterburen (Seal centre Pieterburen) uses the ExtractCompare application to photo-identify harbor seals (Phoca Vitulina). According to the researchers from Pieterburen, the problems with this software are that it is slow, difficult to use and that it is no longer supported by the developer. However, they are forced to use it because their two huge databases of thousands of seal images and id's are only stored as an ExtractCompare databases. Therefore it would be beneficial for them if they could convert their databases into ones used by other, ideally better software. A careful investigation must be conducted in order to choose a replacement which can improve on the existing process.

The current process, as explained by the researchers, consists of multiple steps.

1. First, a new seal picture must be inserted into the database manually using MS Access. This process cannot be avoided since it is necessary to be able to add additional information with each picture.

2. Then the ExtractCompare app is opened. The researchers state that it has to be opened as Administrator, otherwise an error occurs. After the app is opened, the image that was just uploaded has to be selected again, since the application does not know which image it is.

3. Then, the application can be used to compare the image against a database. However, there is a limitation on how many comparisons are allowed at once. This obviously impedes the software's ability to be useful.

4. The software comes up with a list of potential matches, which can then be manually reviewed by the researcher to confirm them.

This process takes quite a long time (over a minute) per image. It is also quite clunky and not user-friendly. The ExtractCompare app also lacks some needed functionality. The researchers also state that they want to merge their two databases into one and identify the overlap. The two databases represent seals spotted in the wild and seals released from the seal centre. ExtractCompare should in theory be able to do this, but the researchers don't know how and the documentation is lacking.

The focus of this project will be to improve the current process which is described above by converting the databases into different software. Once the databases are converted into other equivalent software, this should come naturally.

From the problem statement we can find the following research questions:

1. **Which software is the best at the moment to photo-identify seals?**
2. **What is the best way to convert an ExtractCompare based seal image database to another database type?**
3. **What does the conversion need to do in order to enable the merging of two databases?**

The comparison and overall definition of 'best' in this context is further elaborated in later sections.

2 Motivation

There is a clear benefit to allowing animal researchers to transition to using better software for identifying animals. This saves them time, grants them access to better tools which have higher accuracy and more new features and allows them to do their job more effectively. Once completed, the proposed software can help any researcher stuck with outdated software to transition to using modern software. This can save them time, money and resources which can be used elsewhere to contribute to animal well-being.

3 State of the Art

There are multiple existing apps for seal recognition based on different principles. Some criteria therefore need to be specified which can be used to compare various applications that exist. The comparison is shown in Table 1.

An important factor is cost. Cost can be measured in terms of money, since some software has to be purchased as opposed to other, free alternatives. Cost can also be measured in terms of time, since some apps take longer than others both in the human-assisted phase (if present) and in the automated recognition phase [3]. Time must also be spent to learn to use a new application, so ease of use and user-friendliness are also an important metric. Therefore cost is represented in the comparison as two columns, actual cost and the GUI column, which represents how user-friendly the application is. That ties to the time a user must spend inputting a photo. The automated time is not as important since researchers usually process photos incrementally.

Of course, the accuracy of any application is important as well. For many applications the actual accuracy is unknown or incomparable, so it is not mentioned in the comparison table. All the mentioned application are accurate enough to be considered. This is proven by the fact that all of these applications are being used by animal researchers.

Another factor that plays a role in determining the robustness of an application is maintenance. Some of the existing apps are over 10 years old, which can introduce problems while running them as well as make it impossible to deal with any errors and bugs that might arise. An app which is being actively maintained can get new features, improved recognition algorithms and bug fixes which make it much more likely to be usable in the future as opposed to an old app which is no longer being supported. Therefore, a open-source column is evaluated in the comparison table, along with how recent the support for the application is.

Even though not particularly important for this project, each application is also evaluated based on how many animal species it supports. This is just to see whether the applications are portable and whether the

	Cost	Open-Source/Support	GUI	Supported species
ExtractCompare	Free	No, unsuported	Bare	20
IBEIS	Free	Yes, active support	Very good	Any
WildID	Paid	No, active support	Very good	Any
SealNet	Free	Yes, recent support	None	Seals
AmphIdent	Paid	No, unknown	Yes	5, no seals
APHIS	Free	No, unsupported	Yes	Any
Discovery	Free	No, unsupported	Yes	Any

Table 1: Comparison table of current applications

proposed solution might extend to other animal species than just seals.

A facial recognition app called SealNet has been proposed [1]. The paper claims an accuracy of 85% for this app. However, the photos that can be used for this app are limited to photos where both eyes of the seal are visible in the picture. This limitation is imposed by the facial recognition mechanism which requires the face of the seal to be visible at a reasonable angle. The deployment of SealNet is extremely complicated, requiring advanced computer knowledge for installation and usage since there is no GUI.
Other software which can be used to identify seals by comparing the seal's color pattern on their flank, neck or head includes ExtractCompare, I^3S and Wild-ID [1]. These are comparable because they all use the same mechanism for seal recognition, which relies on the pattern instead of the seal's face. Out of those three, Wild-ID has been shown to provide the highest accuracy, followed by ExtractCompare [3]. Therefore, I^3S will be excluded from the comparison due to lack of accuracy.

However, there are many more examples of such software, some more advanced than others. One good example is IBEIS, a software which can pattern match any visually recognizable animal species. IBEIS is based on the HotSpotter algorithm [2] and is being actively maintained and developed. The IBEIS software also offers a user-friendly interface, hence making it a much better contender than for example SealNet with respect to the desired audience.

Paid alternatives exist too, such as AmphIdent. AmphIdent is rated highly compared to other software [4] for amphibian identification. Technically, it should work for any animal since only the preprocessing step changes depending on the species [4]. However, it is very expensive and unfeasible for small research laboratories.

Other software that should be mentioned for completeness' sake are Discovery and APHIS. They do not stand out from the competition in any substantial way, but they are included in the comparison to ensure nothing is missed.

After comparing the options, IBEIS seems to be the most viable current alternative for seal pattern recognition. The other alternatives are either complicated to use, expensive or no longer being supported by the developers. IBEIS has a very user-friendly interface, it boasts an accuracy of 95% or more [2] and is actively being maintained and developed by Jon Crall, its creator. Hence it can be considered to be a state-of-the-art application and should serve as an example of what is possible.

3.1 Search terms

The used search terms are: "seal pattern recognition" and "harbour seal photo-identification". These searches in the SmartCat software yielded several results, all of which are scientific papers. There is a lot of literature about image comparisons, but the above mentioned search term helped to narrow the search down to seal-specific articles. From a Google search, a webpage was found which compared software for animal identification. This webpage served as a starting point for more research into the current state of software. It also led to searching Google Scholar about the software IBEIS, which yielded the article about the underlying algorithm written by IBEIS's creator.

4 Proposal

The proposed solution to the problem is to create software which can convert databases made with ExtractCompare into a different software. This will not only help the specific use case of the Zeehondencentrum Pieterburen, but also provide a lasting solution for converting databases into a newer, more usable application. The target application for seal identification is IBEIS, since as described in the previous section, it is the best currently existing software for this purpose. Ideally, the final software will be able to accurately convert data into IBEIS, merge databases and introduce the researchers into IBEIS so they can learn to use it quickly and seamlessly transition away from ExtractCompare.

An important factor in seal identification is accuracy. The solution must be as accurate as possible in order to avoid misidentifying seals in both ways, since errors in identification can cause wrong estimates of population size. IBEIS is built on the HotSpotter algorithm which has been shown to have an accuracy of 95% or higher depending on animal species and other factors. User-friendliness is another important factor. Since the researchers using the software aren't always proficient with computers, the software they use needs to be user-friendly and easy to learn. For example, it should have a graphical user interface (GUI) which is much easier to navigate than a command-line interface. IBEIS does have a GUI where the user interacts with the application. ExtractCompare also has a GUI, however it is complicated to use, unintuitive and very slow. The time it takes to input a new image and compare it against the database in ExtractCompare has been shown to be around 1m30s [3]. As the researchers have to spend their valuable time performing this repetitive task, it should take as short as possible and ideally be completely automated. The last factor that impacts the solution is maintenance. The researchers shouldn't need to be very technically skilled in order to use the software, so any bugs or issues should ideally be solved by the developer. This is only possible if the app is being actively maintained. This is the case with IBEIS, making it a good candidate for the target software.

Multiple stages of the current process will be improved by this solution. After the conversion, the researchers will use IBEIS. Therefore, the database entry will be made much simpler since it is done directly in the application, without the need for splitting the task in two parts. This also helps avoid having to find the picture which was just inserted into the database. These improvements along with IBEIS's faster matching algorithm will substantially cut down the time it takes to insert and match each image. IBEIS should also offer a much smoother user experience than ExtractCompare, since it is up to date and getting active support. Overall, all of these improvements are gained by switching to the use of IBEIS. The database conversion only needs to happen once, therefore it does not need to focus on optimization or a GUI. Of course, documentation must be provided to allow any user to do the conversion on their own database with no problems.

Testing will be very important in the process. Obviously, the database must be tested after the conversion to ensure consistency with the original database, and to ensure that IBEIS is working as intended and correctly matching against the database. Testing will be a key part of the project to ensure that the conversion went smoothly and to ensure that the current process is being improved rather than degraded. If possible, the documentation should also be tested by non-technical personnel to ensure that it is clear and easy to follow without a technical background being necessary.

There are currently no alternative applications providing the functionality to convert existing Microsoft Access databases created using ExtractCompare to any other identification software. The aim of this project is to create an application which can automatically do this. Since IBEIS is semi-automatic and requires the user to highlight the area of the picture with the animal, there might be a need to use machine vision to identify the seal in each picture. Ideally, the conversion process should be fully automatic, because the databases in question have several hundred or thousand pictures and would be extremely time consuming if human help is needed. If a tool like this can be developed, it can help animal researchers who have invested time and resources into a discontinued software to avoid losing all of their data and to transition to software which is much better.

5 Methods

IBEIS is written in Python and exposes an API which can be used to access its data management and algorithms. This can be used to automatically import all the seal images. A library for accessing MS Access databases exists for Python called pyodbc. This library can be used to read the ExtractCompare database and get all the information from it. Then, the information can be fed to the IBEIS API to create a new database. If the API requires additional information about the picture such as the position of the animal, this will be implemented using machine vision in python using machine learning libraries like pytorch, or a ready-made solution if one can be found.
Testing will be set up in a way that will verify the resulting converted database in several ways, including integrity of data, accuracy of matching and ease of use of both IBEIS with the new database as well as the conversion tool itself. Heavy focus will be placed on testing to prove that the conversion works smoothly and without errors.

The application will be written in Python due to ease of use and compatibility with IBEIS. The databases to be converted for Zeehondencentrum Pieterburen are around 22 GB total. These are stored both in the seal centre as well as on the student's computer. To transfer them, an encrypted email based service was used. The database will not be made public and will not be stored under version control. Backups of the database have been made to ensure no information is lost in case of an error, since it is not under version control. For version control of the software itself, a repository on GitHub will be created where the code will be stored. This repository will be private during development and possibly published when it is finished.

6 Planning

Dates	Plan	Deliverable
17.4	Project start	
18.4	Planning	Detailed plan
19.4 - 21.4	Read IBEIS API	notes on API usage
24.4 - 26.4	Explore seal database	Database schema diagram
27.4 - 28.4	Read DB into Python	Code
1.5 - 2.5	Test database	Tests, correct database
3.5 - 17.5	Connect app to API	Code
18.5 - 26.5	Test DB conversion & IBEIS	Tests
29.5 - 2.6	Convert and merge DBs	IBEIS database, code
5.6 - 9.6	Test IBEIS database	Testing
12.6 - 13.6	Documentation	Documentation
14.6 - 23.6	Write thesis	Thesis
28.6 - 30.6	Presentation	Slides + Live presentation
3.7 - 7.7	Finish up all deliverables	Submission

Table 2: Project planning

Naturally, unforeseen challenges might arise during the project. If this happens, planning might need to be changed or pushed back to some extent to accommodate time. This will be done dynamically during the project, also with respect to each task's duration since it is unknown at the moment which tasks will be slower or faster than planned for. There also is a safety buffer during July which can be used in extreme circumstances, but this is highly unlikely and should not be necessary.

There is a risk in the fact that IBEIS may not be able to work with the kind of images that are stored in the ExtractCompare database, for example because they may be only a part of the seal's body. In this case, the solution might be to either choose a different application to use or to find some way of processing images that will allow IBEIS to work with them. This would of course require major changes to the planning, in which case the planning will be completely redone to accommodate the new tasks and the changed objective of the project to ensure timely completion of the project.

References

[1] Zach Birenbaum, Hieu Do, Lauren Horstmyer, Hailey Orff, Krista Ingram, and Ahmet Ay. SEALNET: Facial recognition software for ecological studies of harbor seals. *Ecology and Evolution*, 12(5), April 2022.

[2] Jonathan P. Crall, Charles V. Stewart, Tanya Y. Berger-Wolf, Daniel I. Rubenstein, and Siva R. Sundaresan. Hotspotter - patterned species instance recognition. In *2013 IEEE Workshop on Applications of Computer Vision (WACV)*, pages 230–237, 2013.

[3] Izzy Langley, Emily Hague, and Mònica Arso Civil. Assessing the performance of open-source, semi-automated pattern recognition software for harbour seal (p. v. vitulina) photo ID. *Mammalian Biology*, 102(3):973–982, December 2021.

[4] Maximilian Matthé, Marco Sannolo, Kristopher Winiarski, Annemarieke Spitzen van der Sluijs, Daniel Goedbloed, Sebastian Steinfartz, and Ulrich Stachow. Comparison of photo-matching algorithms commonly used for photographic capture-recapture studies. *Ecology and Evolution*, 7(15):5861–5872, July 2017.