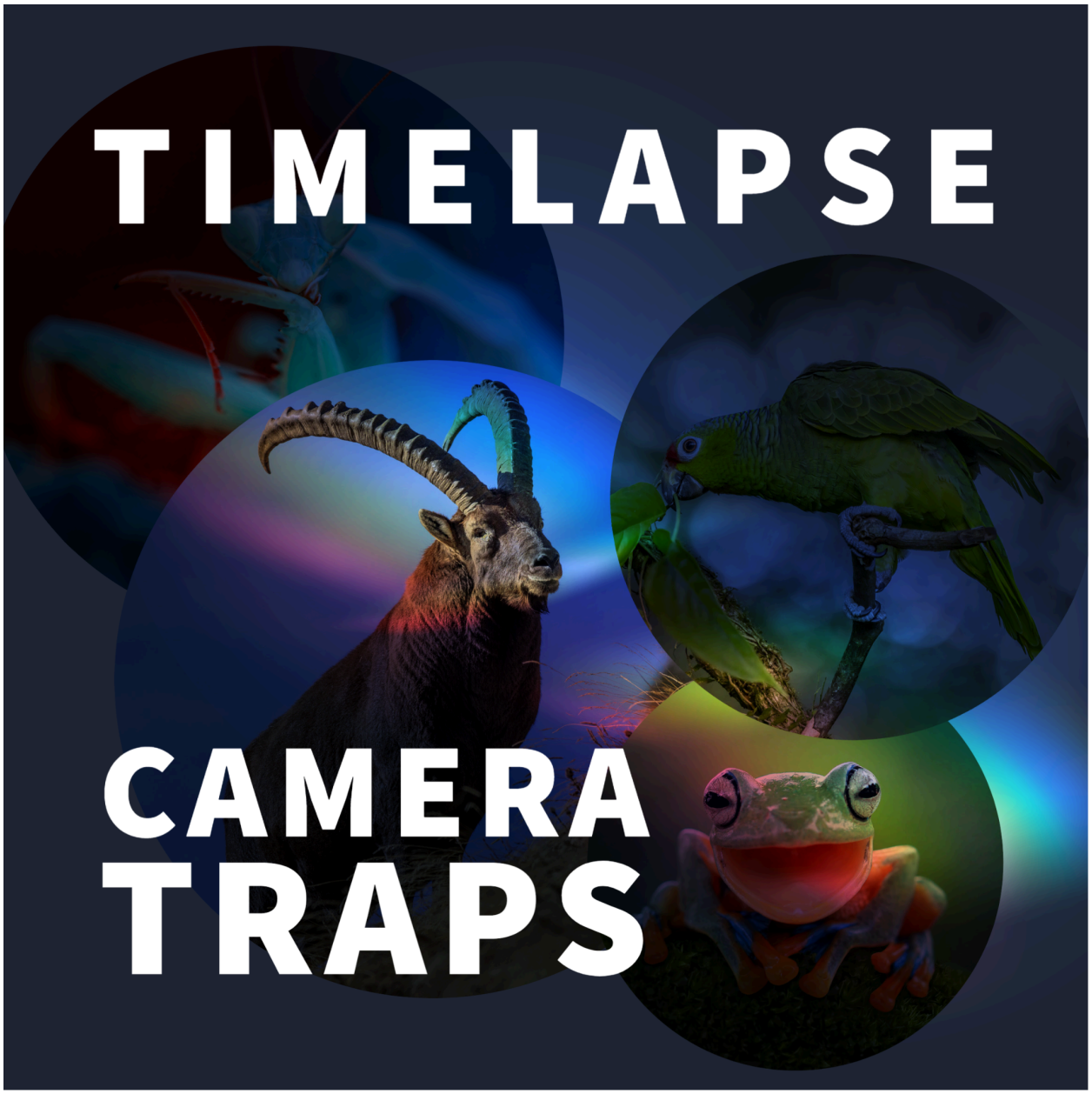
Camera Traps and Timelapse

Francis J. Adair, 2022



Contents

Introduction

The following guide will demonstrate a workflow that can be used with Timelapse by Saul Greenberg software to annotate imagery and/ or video.

- Process steps are denoted by pipeline symbols | process > then… |

Figure 1 - Example camera trap snapshot with metadata labels.

Camera Traps

Imagine being sat motionless in the middle of nowhere, camera in hand, ready to snap a photo at the slightest of movement. Now imagine doing that for a month or more without any breaks. It is easy to see why camera traps have become indispensable for ecological field studies.

Using an Infra-Red beam, the camera trap detects a passing animal or very often just a rustle of the wind. Once a detection has been made, a sequence of photos or a video of defined length is recorded alongside metadata e.g. date and time of day, temperature, moon cycle, etc…

Setting up ‘CTinfo’

- Each camera trap should have a named SD card and be placed in a named location (folder), e.g. cam26 and loc42.

How you choose to set up your camera trap project is beyond the scope of this document. A typical camera trap layout survey may record key details in a Comma Separated Values (.csv) file titled CTinfo, leaving us with this summary of the camera trap project:

site	array	location	camera	model	start date	end date	latitude	longitude	waypoint
A1	east	loc01	cam01	Reconyx	xx/xx/xxxx	xx/xx/xxxx	0.0000	0.0000	wp01

Table 1 - CTinfo layout example.

Compiling

Once all the SD cards from each camera have been collected, the footage must be compiled. This will require you to create a folder on your hard drive called ‘*Footage’*, that can contain further sub-folders if ‘*Arrays’* have been used. Name each sub-sub-folder after the camera, and its location in the format of *locXXcamXX*. Then extract the contents of each SD card to their corresponding folder so that your *whole camera trap project is contained in the Footage* folder. (Array here refers to having more than one site within a camera trap project).

Reviewing

Now that the Footage folder has been compiled with a *folder* for each of the retrieved camera traps, we can quality assess them against the CTinfo.csv. First, create a new *folder* within the Footage folder titled as ‘Backups’ where any *non-concordant* data can be discarded, as it is important to *never delete any data*.

Non-concordant Cameras

- May not contain any data at all.
- Might contain the wrong/ misplaced data:
- Are named differently or absent from the array.
- Have been duplicated from a different camera or project.

Example, part 1

Opening the folder *loc04cam13* and looking at the footage itself, you can see that the camera name has been set to *cam22*, not *cam13* as the *folder* you named implies. By referring to the CTinfo, you can confirm that *cam13* has been located at *loc04* throughout its deployment, whereas, *cam22* has been deployed elsewhere at *loc09.*

Possible causes for the mix-up might include

- The SD card has not been updated with the new camera name; i.e. cam22 has not been renamed to cam13 prior to deployment.
- cam22 has been placed in the wrong folder, i.e. loc04cam13.
- The data from cam22 has become mixed with that of cam13.
- cam22 is from a separate project and had been misplaced.

Steps that could be taken in resolving the issue

- Refer to the footage and locate the beginning and end photos, where a whiteboard with the date, camera name, and location should be clearly displayed by whoever is deploying/ collecting the camera.
- Look at the rest of the SD cards/ folders to try and locate one called cam22, and refer to CTinfo to check it is in the correct location.
- Check for duplication (discussed further later) by finding cameras containing near or the same number of files and/ or comparing the surrounding environment.

Example, part 2

You found that there were two whiteboards being shown at separate occasions within the footage for the SD card, *cam13*. One stated *cam13 loc04,* and another stated *cam22 loc09* with a different environment. Clearly, part or all of *cam22* has been mixed up with *cam13*.

Thankfully, all the *cam13* footage is made up of MP4 file types, while *cam22* is made up of JPEGs. Therefore, we can easily sort by Type and move the *cam22* footage into the correct folder titled *loc09cam22.*

But wait…There is little way of knowing whether the *cam22* footage mistakenly placed within *loc04cam13* comprises the complete set. Therefore, it would be most advisable to move the entire folder into the **BackUps** folder you created earlier. Then, locate the original SD cards for *cam22* and *cam13* to create the folders *loc04cam13* and *loc09cam22* from scratch.

Corrupted Footage

Data can sometimes become corrupted and cease to display proper timings on all or some of the files. A quick check to see if your data is affected is by sorting each folder by their date. Corrupted files clump together as their date and time often default to something like 1970 at 00:00:00, but for files to be quantitatively analysed they require accurate dates and times.

Therefore, separate any randomly corrupted files into a copy of their folder distinguished by a suffix, e.g. *locXXcamXX*_corrupted, then move that folder into the BackUps folder to avoid deleting it altogether. But, if the camera has instead experienced a systematic error in timings then don't remove the data as it can be fixed later using Timelapse (see Timelapse > Date and Time).

Note: a whiteboard or equivalent note should be used to trigger each camera trap at the *beginning and end* of its deployment period indicating the date, time, location, and camera trap ID.

Renaming

Right now, the files contained within your footage folder will have a nondescript name that isn't really all that useful. So before any annotation can begin, we must first re-name each file with the camera location and name. Phil Harvey's ExifTool may be used with *Windows Command Prompt (CMD)* to rename files *en masse* but has a learning curve. Therefore, Saul Greenberg recommends the use of ReNamer Lite, since it is a free and easy to use software.

- Open ReNamer.exe.
- Add a rule Prefix: (Insert Meta Tag) File_FolderName.
- | Ctrl + A | to select all the files contained in the folder.
- Drag all the files from the camera folder into ReNamer.exe
- Rename all files and clear the queue once complete.
- Repeat for each camera folder individually.

Setting up ‘CTindex’

- So, you’ve got a CTinfo, now it’s time to create a CTindex.

The CTindex is the quality controlled version of CTinfo, wherein each camera’s *read* deployment/ retrieval date is stated. First, create CTindex.csv using the columns outlined below. The process of completing the CTindex involves manually looking through each SD card/ folder.

index	project	array	location	camera	start date	end date	fail type	note	exclude
1	Zoo	east	loc01	cam01	xx/xx/xxxx	xx/xx/xxxx			N
2	Zoo	east	loc01	cam01	xx/xx/xxxx	xx/xx/xxxx	Obscured		N
3	Zoo	east	loc01	cam01	xx/xx/xxxx	xx/xx/xxxx			N
4	Zoo	west	loc02	cam02	xx/xx/xxxx	xx/xx/xxxx	No data		Y

Table 2 - CTindex layout example.

- Complete as much of the CTindex as possible before beginning the annotation procedure within Timelapse.

Note: In the above table, *cam01* has been split between three index rows. This is because the camera had become *obscured* part-way through its deployment and introduced occlusion bias. Splitting the camera up like this on the CTindex, means the obscured data can be removed.
(see Failure Criteria below)

You should return to CTindex throughout the annotation procedure. Starting this process early can save you a headache later, as it will be harder to edit folder structures once annotation begins. For now, just fill it in as best that you are able to.

- **Tip:** comparing CTindex side-by-side with CTinfo is a handy way of corroborating dates between the two and illuminating any issues.

Failure Criteria

The camera will be considered as failed if one of the following conditions are met:

- Malfunction – usually the camera has expired over a couple of days.
- Placement – placed too high or inappropriately.
- Misplaced – data does not belong to the project.
- Missing – data is missing from the project.
- Duplicate – data repeated in the project files.
- Obscured – object is blocking the Field of View (FOV).
- FOV – field of vision is too small or not appropriate.

When FOV is reduced over time e.g. by vegetation growth, then split the deployment camera between multiple index rows on CTindex as in Table 2 to isolate the worst of it and mark the ‘fail type’ as ‘Obscured’. Sometimes a camera can be split over three or more index rows.

If a camera has failed it means that the data contained cannot contribute towards the quantitative analysis of the results. In these cases, the CTindex should indicate the type of failure in the fail type column. The Camera folder (or subset of failed images/ videos) can then be moved into the BackUps folder, so as to not be included at the annotation stage.

Timelapse

Saul Greenberg's Timelapse software annotates footage and utilises folder structures to manage the content. Critically, it is as of writing the only annotation software that supports both images *(.jpg)* and videos *(.avi, .mp4, or .asf)*. Its use of editable templates allows for custom workflows guided by a number of helpful in-built features for streamlining the process.

Getting Started

As seen in The Timelapse QuickStart Guide

1. Go to the Timelapse website: https://saul.cpsc.ucalgary.ca/timelapse/
2. Follow the links to the Download and Install page.
3. On the download page, read the instructions on how to download, install and run the software. The download will be a single zip file named: Timelapse-Executables.zip
4. Extract the contents of that file to a convenient location (e.g., your desktop). You should now have a folder called Timelapse.
5. Within that folder, you will see two programs: Timelapse.exe and TimelapseTemplateEditor.exe. As all other files in that folder are required by Timelapse, leave the Timelapse executables in that folder.
6. For convenience, create shortcuts on your desktop to these programs.

Folder Structure

Timelapse consists of two file types in this order 1. Template (.tdb), and 2. Database (.ddb). They're in order because first you must create a Template using the Timelapse Template Editor software (see below). Store the template within the folder containing the image and/ or video files. Afterwards, you can open that Template file in Timelapse2.exe from the menu | *File* > Load template, images, and video files… | to begin the annotation project.

Main Project Folder:		
Format	**Name**	**Description**
folder	BackUps	Auto-created upon opening the Template.tdb within Timelapse.
folder	DeletedFiles	Auto-created upon deleting a file within Timelapse.
folder	Footage	Folder (+ sub folders) read by Timelapse to form the project.
TDB	Template.tdb	Designed using Timelapse2TemplateEditor.exe (note: Timelapse has been updated since 2022)
DDB	Database.ddb	Contains the annotations associated with Footage.
CSV	Data.csv	Exported comma separated value (CSV) version of the Database.ddb.

Table 3 - Timelapse folder structure.

- Timelapse uses folder structures to read the footage data.

Templates

If you have ever used something like Microsoft Excel, Google Sheets, or Apple Numbers to edit a CSV file, then using the Timelapse Template Editor will be easy. The default template will start you off with columns as displayed below.

Type	Default Value	Label	Data Label	Tooltip	Width	Copyable	Visible	List
File		File	File	The file name	100	Y/N	Y/N	
RelativePath		RelativePath	RelativePath	Path from the root …	100	Y/N	Y/N	
DateTime	0001-01-01 00:00:00	DateTime	DateTime	Date and time …	160	Y/N	Y/N	
DeleteFlag	false	Delete?	DeleteFlag	Mark a file …	20	Y/N	Y/N	
FixedChoice	example	Example question	ExampleQuestion	Labels appear in …	100	Y/N	Y/N	Item1…

Table 4 - Timelapse Default Template with example row.

There are *five question types* that can be chosen from:

- Choice – choose from a list of predefined values.
- Flag – click the checkbox to apply TRUE/ FALSE.
- Count – enter a number greater or equal to 0.
- Note – write a comment.

Update the Choice List

Templates can be updated after the annotation has begun but a procedure must be followed if any already entered values are changed e.g. change the choice value Human >> Homo sapiens for the whole project.

Open the project on Timelapse2.exe then navigate to | File > Export data in the current selection as a .csv file… | and on the CSV change all the values of Human to Homo sapiens. Save a copy of the CSV file and re-open the project again in Timelapse, navigate to | File > Import data from a .csv file … | and select the new CSV.

Sometimes, re-importing the CSV over the top of already annotated work will cause errors. It's worth using | Select > Custom Selection… | to filter by the new choice list and make sure the old option of Dog has been removed.

Navigation

Even before any annotation has taken place, Timelapse offers a number of ways to quickly navigate through your data. The most convenient is that of the | Select > All files in a folder and its subfolders | from which you can view the folder structure itself. Other methods for selection include;

- | Select > Custom selection… | from which a subset of your images and/ or videos can be viewed by a set of criteria you have defined.
- | Edit > Find file | allows you to search specific files using keywords.

Copying

When annotating thousands of records at a time, there are often streams of files in sequence that will all require the same information. Timelapse provides some very quick ways to annotate lots of records at once. Start by opening your project in Timelapse2.exe and right-clicking on any of the fields;

- | Copy to all | will populate the field value of every selected file with the current entry, even if the entry-box contains a blank value.
- | Copy from forward to end | duplicates the current field value up to and including the final record, but excludes all records preceding it.
- | Propagate from the last non-empty value to here | looks back to the previously filled record under the current field and duplicates that value into all the records between and including the current file.

QuickPaste

Is a menu that can be opened by pressing | Q | on your keyboard. Making a | New QuickPaste… | presents each entry field in your template to have their values customised and saved as a preset. The preset can be named and then used via a macro of | Ctrl + (n+1) | to populate every field entry box at once.

Duplication

There are two methods for duplicating records in Timelapse (see below) e.g. for when there is more than one species present in a media file. It should be noted that Timelapse duplicates records, i.e. rows, and not the file itself so having a 'Flag' question in your template called Duplicate to differentiate between them.

- | Edit > Duplicate this record… | creates a new copy of the current record (not the file itself), populated by blank/ default values (a duplication counter is located at the top-right of the screen).

Remove Duplicates

- | Edit > Delete > Only the data associated with the current file… |

Mass Duplication

Sometimes you might find a long sequence of files that contain two or more species. In these cases, it is not practical to duplicate every single record. Therefore, it is easier to duplicate all the files instead.

- Sequence = the selection of files to be duplicated
- Navigate to the first file in the Sequence
- Note down the filename, camera, and location
- Annotate each record in the Sequence with species B
- Note down the filename of the last record in the sequence
- Leave Timelapse and open the footage folders
- Locate the sequence of files within its respective folder
- Copy and paste the Sequence within the same folder
    - e.g. IMG_01.jpg >> IMG_01-copy.jpg
- Reopen Timelapse at the beginning of the Sequence
- |File > Add image and video files to this image set… |
    - Import the folder containing the Sequence of copies
- Annotate each record-copy in the Sequence with species A

Duplication Dilemma

- output.csv = csv that you can export after annotating a camera trap project using Timelapse, via | File > Export data in the current selection as a .csv file… |

There may be time that you have to re-import your annotation data from a CSV e.g. updating a choice list after starting annotation means that you must edit the entries of the output.csv and re-import it into the project.

When importing an output.csv into Timelapse using | File > Import data from a .csv file … | duplicates cannot be attributed to any of the files in the database. This is because Timelapse will try to join the output.csv by the File name column to the footage. Timelapse notifies you of a duplicate entry mismatch and the duplicate annotations don't carry across which will result in a loss of data.

Ensure you have a duplicate flag as an annotation field (see Annotation > Designing the Template). Filter by TRUE, then move all the duplicates to a separate CSV called duplicates.csv. Once annotation is complete, join the duplicate.csv to the new output.csv

Without a Duplication Flag

Without including a flag type question in your template.tdb you won't be able to filter out the existing duplicates very easily. If you don't though, assuming you're using Microsoft Excel, this is how you resolve the problem. First, you'll need to have three things:

1. CSV1: output.csv with duplicates (this would be an older export to csv).
2. CSV2: output.csv without duplicates
3. 1 blank duplicates.csv

Then,

4. Ensure that no annotations will be added after this point.
5. Merge CSV1 and CSV2 to create CSV3 with two new columns called 'Dupliacate_ID' and 'Duplicate_Flag'.
6. For the Dupliacate_ID column, combine the columns for | File & RelativePath & PhotoType & Category & Species | using a code like =A2&B2&M2&Q2&R2 to create a unique ID code for each entry.
7. For the Duplicate_Flag column, mark all repeated entries (from merging CSVs 1 & 2) and duplicates using the code =IF(COUNTIF($Z:$Z,$Z2),"repeated","duplicate") but change $Z:$Z,$Z2 to your Dupliacate_ID.
8. Copy the entries marked as 'duplicate' to the duplicates.csv and then join to CSV2.

Metadata

A file extension (.jpg, .png, .avi, .mp4) is determined by what kind of media is contained in the file and is synonymous with the file type. Metadata is a term that refers to the different attributes that describe the media e.g. author, date, size, duration, resolution… The only thing that isn't metadata is the media itself.

Camera trap footage can contain lots of metadata e.g. datetime, temperature, moon phase, camera make… all of which are useful annotation fields. Timelapse offers two methods for populating fields with metadata;

| Edit > Populate one or more fields with metadata… |

- MetadataExtractor (faster) – fewer attributes to choose from.
- ExifTool (slower, more comprehensive) – created by Phil Harvey.

Date and Time

Sometimes camera traps may be set to the incorrect time or worse, their metadata timings can become corrupted, which usually means returning to a default date and doesn't always corrupt the whole camera equally. Conversely, the erroneous camera trap may still operate using an accurate clock, so the range between the first and last captured media could still be accurate but differ from the recorded deployment duration (systematic error). There are a number of ways you can edit the metadata of your camera trap results for annotation in Timelapse, some of which can be accessed via | Edit > Date correction | and include:

- Reread Dates and Times from Files…
- Correct for Standard and Daylight Savings Time Changes…
- Correct for Cameras Not Set to the Right Date and Time…
- Correct for Cameras whose clock runs fast or slow…
- Check and Correct Ambiguous Dates… (format)
- Read dates and times from a metadata field…

The original CTinfo.csv should contain all the camera trap deployment data and so will be helpful in deciding how to resolve any timings but isn't always enough. A photo of the camera trap Set-up or Pick-up or a photo or hardcopy note like a whiteboard, of the original set up date or time can be used to calculate the deployment duration. These variables can be used to work out the true deployment date and time of a camera to fix erroneous timings.

Note: Camera traps are often in American MDY format, Timelapse corrects this to DMY in the DateTime field.

Coordinated Universal Time (UTC)

The base Timelapse2 Template that can be accessed via | Timelapse2TemplateEditor.exe > File > New template… | has a field for 'UtcOffset'. By default this will not be ticked as visible but if the clocks on your cameras or computer have been set for any particular region, you can add it in this field.

Machine Learning

See Timelapse Image Recognition Guide

Using machine learning (ML) in camera trap studies stands to significantly enhance the efficiency of annotation procedures, but is far from replacing it. At this stage ML is best used to identify images containing people, vehicles, animals, and/ or none, which changes the manual annotation workflow (see Annotation).

It is important to remember that ML models, much like humans, are susceptible to errors. These include false positives or false negatives, incorrect identifications, or ambiguities – where several overlapping and possibly conflicting detections occur.

To create an image recognition ML model, it must first be trained by feeding it with a subset of relevant images. The scope of this blog does not include the intricacies of making a model. But, depending on the level of training, either a detector or a classifier type can be used.

Image Recogniser Types

Image recognition models can automatically analyse uploaded files and annotate them with a probabilistic tag that is visible in Timelapse as a bounding box around the subject. The tag corresponds to a category outlined in the training set used e.g. Animal, Primate, Gorilla, that is stored and organised in a separate .JSON file.

- Detectors – identifies the likelihood of coarse groups e.g. Empty, Animal, Human, and Vehicle, which are then enclosed with a bounding box.
- Classifiers – perform fine-grained detections for specific purposes, i.e. the likelihood that each image contains a specific species from a list.

MegaDetector

Microsoft has created a 'detector' type model called MegaDetector. Which has been trained specifically for detecting wildlife and is able to categorise images by Empty, Animal, Human, and Vehicle. Their GitHub page offers instructions on how to set up their API for your own datasets.

Timelapse has officially collaborated with Microsoft's MegaDetector and refers users to the email cameratraps@lila.science for instructions and guidance. Setting up the API on your machine can be time-consuming without computing experience and can take several days to process a large dataset.

Importing ML Output

Once processing is complete and output.json is created:

1. Place the output.json into the same folder as your template.tdb file.
2. Open Timelapse2.exe and load up your project as you normally would.
3. | File > Import image recognition data for this image set. |
4. Import output.json and bounding boxes will be auto-applied.
    - Go to | Options > Preferences… | to adjust settings.

Note: the manual annotation process must now change in order to make best use of the classified data. Whereas before we might annotate camera-by-camera, now we must organise files by their classifications and confidence thresholds. Please refer to Annotation > Image Recognition.

Preferences

| Options > Preferences… |

- Adjusting image recognition settings;
- Changing the exported date format;
- Toggling to auto-populate metadata;
- Change how deleted files are treated;
- Episodes: how many files to check;
- Adjust minimum and maximum zoom level;
- Adjust rapid scrolling performance;
- Adjust image differencing threshold (press Up or Down Arrow on the keyboard).

Annotation

The following section runs through what a typical annotation procedure might look like when using Timelapse2. As previously discussed in the Timelapse > Templates section above, the first step in annotation is to decide what questions you want to ask of the data in | Timelapse2TemplateEditor.exe > File > New template… |

Template Design

Questions you might want to ask of the camera trap data include;

Name	Tooltip	Type
ProjectID	Unique identification title given to the overall project	
Array	Which array (if any) was the camera trap within while deployed	

CamMake	Make of the camera trap being analysed	
CamModel	Model of the camera trap being analysed	
CamSerial	Serial number of the camera trap being analysed	
VidDuration	Video duration	
PhotoType	The category of photo/ video event recorded	Choice
MoonPhase	Moon phase upon photo/ video being taken	
TempC	Temperature in Celsius upon photo/ video being taken	
TempF	Temperature in Fahrenheit upon photo/ video being taken	
Category	Broad cateogry of animal being recorded	Choice
Species	Species level identification of animal being recorded	Choice
Unsure	TICK if you unsure of your selection	Flag
Quantity	Click the counter button, then click on the image to count the entity. Or just type in a count	Count
Injury	TICK if an injury is present within the photo/ video	Flag
Snare	TICK if a snare is present within the photo/ video	Flag
Behaviour	Observed behaviour of the animal when photo/ video was taken	Choice
Author	Person annotating the camera trap dataset within Timelapse [(Forename Initial)(Surname) – one word]	
Comment	Write a comment for this particular photo/ video	
SpeciesPresent	If known populate with Genus species of first species encounter per camera	
HumanType	The apparent occupation of the human/ human activity	Choice
HumanActivity	Populate with unusual human activity/ presence	
DeploymentLocationID	Name of fixed station where the camera is placed	
CamID	Name of camera placed at the fixed station	
BestContent	TICK if photo/ video is noteworthy	Flag
Duplicate	TICK if photo/ video is duplicated from original file in Timelapse.exe	Flag

Table 5 - Suggested template questions

Choices

Some of the examples above are of a choice type (see Update the Choise List), below are some suggestions;

- **Category** - Human, Primate, Carnivore, Hooved, Rodent, Bird, Reptile, Other;
- **PhotoType** - Animal, Empty, Unidentified, Ambiguous, Set Up, Pick Up;
- **HumanType** - Surveyor, Local, Hunter, Logger, Other;
- **Behaviour** - Feeding, Alert, Resting, Grooming, Mating, Digging, Excreting, Adversarial, Vocalising, Other.

The Workflow

SpeciesList.csv – is a list of all the species you identified along the way, ideally in a table with their colloquial names alongside them for reference.

The following workflow is not a rigid procedure but the steps might help to build a foundation for your project (See Camera Traps and Template Design sections):

1. Use | Select > All files in a folder and its subfolders… | to select the first camera within your footage folder.
2. Annotate each PhotoType accordingly e.g. Animal, Empty, Unidentified, Ambiguous, Set Up, Pick Up.
3. If an Animal is spotted, identify it and add the genus and/ or species to the SpeciesList.csv.
4. Annotate the Category accordingly e.g. Human, Primate, Carnivore, Hooved, Rodent, Bird, Reptile, Other.
5. Continue this way, file-by-file till the camera has been fully annotated in this way.
    a. You could make a QuickPaste selection to speed things up.
    b. Remember to flag any Duplicate or BestContent along the way.
6. Repeat steps 1 – 6 for each camera in the footage folder.
7. Close the project and ensure the latest BackUp exists (automatically created by Timelapse).
8. Open your project template.tdb in TimelapseTemplateEditor.exe and add all the scientific names (genus and/ or species) collected on the SpeciesList.csv to the Species field's choice list.
9. Re-open your project in Timelapse2.exe and initiate the template.tdb with | File > Load template, images, and video files… |
10. Use | Select > Custom Selection… | to filter all footage by the categories annotated in step 4.
11. Annotate the Species accordingly for the whole of PhotoType: Animal.
12. Repeat steps 10 - 12 for each camera in the footage folder.

Image Recognition

If using image recognition models then the workflow changes, as outlined below:

1. Before annotation begins use | File > Import image recognition data for this image set. | and select the output.json.
2. Instead of annotating camera-by-camera, you must annotate by the image recognition tags associated with output.json e.g. Empty, Human, Animal, Vehicle…
    a. Use | Options > Preferences… > Automated Image Recognition | to set the bounding box confidence threshold (note: all photos will be tagged).
3. Use | Select > Custom Selection… | and filter by a tag e.g. Animal, between a Confidence of 0.80 to 1.00.
4. Annotate each PhotoType accordingly e.g. Animal, Empty, Unidentified, Ambiguous, Set Up, Pick Up.
    a. If the machine learning has worked correctly, most or all the PhotoTypes will be 'Animal'.
5. If an 'Animal' is spotted, identify it and add the genus and/ or species to the SpeciesList.csv.
6. Repeat steps 2 – 5 for each image recognition tag.
7. Use | Select > Custom Selection… | and filter by 'empty' PhotoType and/ or Category to annotate all ML tags with less than 0.8 Confidence.
    a. If the image recognition is very good, there should be very little media below 80% confidence.
    b. If not, repeat steps 3 - 6 at a lower Confidence e.g. 0.60 to 0.79.

Note: If the confidence threshold for most of the data is low, then it could be faster to manually annotate instead.

Quality Control & Exporting

Once your project has been fully annotated select all and export using | File > Export data in the current selection as a .csv file… | and conclude your annotation. But before you do, it is important to Quality Check your work:

- Ensure all fields are updated with correct metadata (see Timelapse > Metadata).
- Use |Select > Custom Selection… | to filter by each field in turn, leave the field value blank, and if there are any results that indicates missing data so fill them in as required.
    - It is best practice to fill in every field, even if the choice must be null.
    - Only leave fields blank if a decision has not yet been made e.g. a photo marked with a Category of Unidentified should have an empty Species field, whereas a photo marked with a Category of Ambiguous should have a Species of 'Unknown' as ambiguous implies it is unidentifiable.

Extracting Subsets

Once annotation is completed you may want to extract a select subset of data based on your annotation categories. If you have Windows, open the command prompt | Start > Command Prompt (right-click and Run as Administrator) | The following instructions assume you are working with Microsoft Excel:

You will need to create:

- 1 output.csv - for working on.
- 1 blank.csv - for creating the subset.
- 1 text.txt - file for copying over the subset.
- 1 subset folder - to contain the extracted footage.
    - Name the folder after the subset you want to extract and save it to somewhere like; C:\Users\you\Desktop\subsetFolder

Then,

1. Filter the output.csv for the subset you want e.g. filter for 'Meles meles' under the Species column.
2. Copy the filtered columns for RelativePath and File over to the blank.csv and join them into one column separated by a back-slash \
    - Use the Excel expression (Column1 + "\" + Column2) to get RelativePath\File.
3. Copy the new RelativePath\File column over to the text.txt and save it as the filtered subset e.g. Meles_meles, within the same folder as your footage.
4. Open Command Prompt | Start > Command Prompt (right-click and Run as Administrator) |
5. Open the footage folder in Windows Explorer and right-click the address bar to | copy address as text |
6. Change Directory in Command Prompt by typing 'cd ' and then paste the copied address e.g. 'C:\directories\footage', then press Enter on the keyboard.
7. Copy the following code 'FOR /F "delims=" %N in (list.txt) do COPY "%N" C\:Destination'
    - Replace list.txt with the name of your text.txt file created in Step 3.
    - Replace 'C\:Destination' with the file path of the subset folder e.g. for Meles_meles.
8. Paste the code into Command Prompt and press Enter to extract the subset of files to the subset folder.

Hints and Tips

- As a rule it is probably best to mark empties as 'Empty' even if you know an animal is just out of shot. It keeps your annotations more consistent.
- Specific annotation fields and their entries will not be lost if the template.tdb is edited during the annotation procedure e.g. altering the Choice List.
- If stations are removed during the annotation procedure | File > Export data in the current selection as a .csv file… | remove the camera's photo entries (e.g. with excel) then move the old database.ddb to the BackUps folder. The up-to-date .csv can be imported via | File > Import data from a .csv file … |
- Split a project across several databases of below 100,000 photos to prevent import errors.
- New stations may be added during annotation via | File > Add image and video files to this image set… |
- Camera traps are often in American MDY format, Timelapse corrects this to DMY in DateTime.
- Ensure only one database.ddb file exists for the project and place defunct versions into the Backups folder.
    - Or they will be merged into one database.ddb file.
- By default, mark any corrupted data by typing C in the IMAGEQUALITY field.

Look out for …

1. Double-spaces when entering choice options.
2. QuickPaste fields being checked unnecessarily.
3. Forgetting to select 'All Files' before exporting.
4. Keep a copy of the output.csv at all times.

Acknowledgements

I would like to thank Naomi Matthews and Stuart Nixon for their expertise and guidance.

References

- Greenberg S. (2022). Timelapse Database Guide. Retrieved from: https://saul.cpsc.ucalgary.ca/timelapse/pmwiki.php?n=Main.UserGuide
- Greenberg S. (2022). Timelapse Image Recognition Guide. Retrieved from: https://saul.cpsc.ucalgary.ca/timelapse/pmwiki.php?n=Main.UserGuide
- Greenberg S. (2022). Timelapse Quick Start Guide. Retrieved from: https://saul.cpsc.ucalgary.ca/timelapse/pmwiki.php?n=Main.UserGuide
- Greenberg S. (2022). Timelapse Reference Guide. Retrieved from: https://saul.cpsc.ucalgary.ca/timelapse/pmwiki.php?n=Main.UserGuide
- Greenberg S. (2022). Timelapse Template Guide. Retrieved from: https://saul.cpsc.ucalgary.ca/timelapse/pmwiki.php?n=Main.UserGuide