PrimateFaces

Tool for running face detection and recognition on images and videos of non-human primates

Claire Witham
Institute of Neuroscience, Newcastle University
c.l.witham@ncl.ac.uk

Version 1.0
13th December 2016

Contents

1. Terminology

Face Detection: Find a face or faces in an image or video frame
Face Recognition: Identifying a face as belonging to a certain individual, species or class

2. Background

This is a Matlab based application designed to allow you to train and run face detection and recognition for non-human primates. This was initially developed for use with rhesus macaques but can be tested for any non-human primate species.

3. Requirements

Computer running Microsoft Windows 7* or later and an internet connection.

*If you have a machine running a recent version of Matlab and the toolboxes listed in step 4 then it may be possible to run the tool on Linux or Mac (this has not been tested).

4. System Installation

Two installation options are provided depending on whether you own a Matlab licence or not.

Matlab users (tested with versions 2015b and 2016a; requires image processing, computer vision and image processing toolboxes):

a) Download zip file (PrimateFaces_program.zip)
b) Unzip folder and navigate to mfiles folder in Matlab.
c) Run primatefaces_main.m

Other users: PrimateFaces_Matlab_Installer.exe

a) Download zip file (PrimateFaces_program.zip)
b) Open PrimateFaces_Matlab_Installer.exe file
c) Under installation options choose the location in which to install the application
d) You will then be prompted to download and install the Matlab runtime environment version 2015b

5. Getting Started

The tool has four main components:

1) Train a face detection model.
2) Run face detection on images and/or videos using either newly trained model (1) or one of the models provided (see section 7).
3) Train a face recognition model
4) Run face recognition on images and/or videos. Requires both a face detection model (2) and a face recognition model (3).

5.1 Face Detection

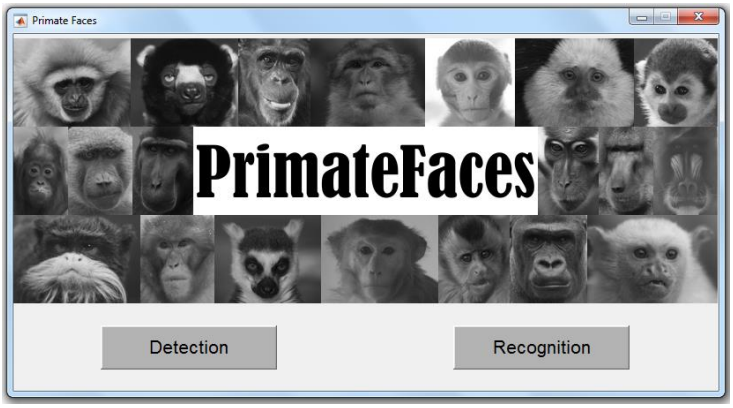
Depending on the species of interest you can either prepare your own facial images and train a new detector (Section 6.1.1; use a standard image editing package to crop faces from larger images) or use one of the provided models and try running your images through the tool (6.1.2).

5.2 Face Recognition

Before starting with face recognition make sure that the face detection component is running well on your images/videos. This is also a good way of producing images for training a new recognition model. Face recognition models can be for trained to distinguish between different animals, different classes such as gender and different species. Arrange the facial images according to the specific folder structure detailed in section 6.2.1. Once a model has been trained it can be applied to images/videos alongside a suitable face detector.

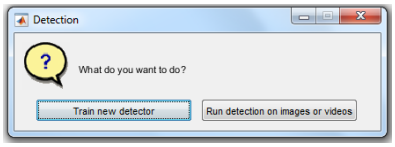
6. Using PrimateFaces

Upon start up the following welcome screen will display:



6.1 Detection

Clicking on Detection will bring up two choices:



6.1.1 Train new detector

Requirements:

1) Folder containing positive images of object you want to detect (e.g. faces, eyes or other features). These can be prepared using standard image editing software.
2) Folder containing negative images (require at least two times as many negative images as positive).

The program will ask you to select the folder containing the positive images. A wait bar will then appear while the program loads and checks the positive images.

You will then be asked to select the folder containing negative images and then input a name to save the detector as. We recommend you save the models in a different folder to the main program folder.

An information screen will then display the number of images, the size of the detector (this is automatically set by the program) and the training. The program will then train the new detector. This may take some time (amount of time depends on number of factors including amount of images and

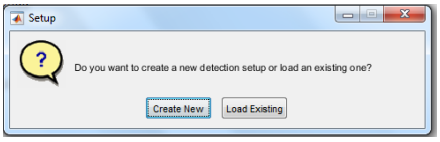
computer processing power). During this time a temporary folder will appear containing the intermediate training stages with the suffix "trainCascadeTemp". During training you can open up this folder to see how many stages have been completed (each stage is saved as a separated xml file). Do not add or remove any files from this folder. The model will try and train for a maximum of 40 stages. In many cases it will run out of images at an earlier stage and produce a model with fewer stages. When the training is finished the information screen will display "Model trained with ** stages". An xml file should now appear with your chosen save name.

6.1.2 Run detection on images or videos

Requirements:

1) Face detection model (xml file; use existing or train own)
2) Folder containing images or videos. The following formats are accepted JPEG, TIF and PNG for images and AVI, MP4, WMV and MOV for videos.

The program will ask you to choose between creating a new setup and loading an existing setup.

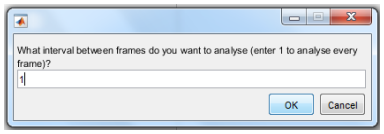


Clicking on "Create new" will take you through the following steps:

a) Select face detector model file (xml file; e.g. RhesusFaceModel.xml).
b) Set threshold for face detection. Some experimenting will be required here to determine the best threshold for your model and images/videos. A very low threshold will have higher sensitivity but also a high false positive rate. A higher threshold will reduce the false detection rate but also reduce the sensitivity.
c) Set minimum size for face detection. To speed up processing (in particular if the faces you are trying to detect are large) it can help to set a minimum face size (the default and absolute minimum size permitted is the size of the face detection model – this is usually around 24x24 pixels).
d) Add additional feature detectors. Once a face has been detected it will be cropped from the original image and resized (to ~100x100 pixels). It is then possible to detect features such as eyes and noses on this cropped image (with suitable detection models) to increase the accuracy of the face detection. If you to choose to add additional detectors you will be asked for the file name of the detection model (e.g. RhesusEyePairModel.xml) and then asked to give a threshold for the model.
e) Finally you will be asked to save the setup file (this makes it quicker to load in future).

Choosing "Load existing setup" instead and a dialog screen will appear asking you for the setup file (this will have a .mat extension).

Once the setup is finished you will be asked for the folder containing the images/videos to be analysed (the folder can contain images, videos or both) and the folder location to save the output to. If there are videos to be analysed you will also be asked how many frames you want to analyse (if 1 is entered here every frame will be analysed).



The program will then process the images followed by the videos. A wait bar will appear showing the progress. If a face is detected in an image or a video still the program will save both the cropped facial image and the original image/still as jpeg files in the output folder. For video the jpeg file names include both the video filename and the frame number.

Cropped facial image files

Processed image file

After all the images have been processed all faces detected will be listed in a .csv file labelled image_results. Each face detected appears on a separate line with the image filename and the x-y coordinates of the face. For video files each video file will generate its own .csv file labelled filename_video_results. Again each face appears on a separate line but this time the frame number is given along with the x-y coordinates.

	A	B	C
1	ImageFile	X	Y
2	testpic_10_3.jpg	1262	506
3	testpic_10_3.jpg	1264	174
4	testpic_10_3.jpg	662	290
5	testpic_11_2.jpg	745	506
6	testpic_11_2.jpg	1140	198
7	testpic_12_2.jpg	1038	195
8	testpic_13_2.jpg	1238	475
9	testpic_13_2.jpg	1249	157
10	testpic_14_2.jpg	1120	108
11	testpic_14_2.jpg	1608	120
12	testpic_15_2.jpg	1229	136
13	testpic_15_2.jpg	707	203
14	testpic_16_1.jpg	979	265
15	testpic_17_1.jpg	1070	319
16	testpic_18_2.jpg	635	623
17	testpic_18_2.jpg	1377	328
18	testpic_1_1.jpg	951	249
19	testpic_20_2.jpg	1264	91
20	testpic_21_2.jpg	1137	268
21	testpic_21_2.jpg	956	667
22	testpic_22_2.jpg	1098	144
23	testpic_22_2.jpg	1539	167

Image_results

Image CSV table

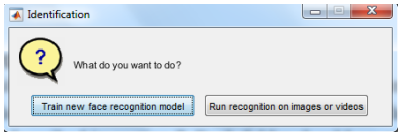
	A	B	C	D
1	FrameNumber	X	Y	
2	1	463	318	
3	2	811	156	
4	2	461	319	
5	3	459	318	
6	4	458	318	
7	5	454	317	
8	6	439	316	
9	6	804	147	
10	7	430	316	
11	7	807	147	
12	8	811	148	
13	8	423	314	
14	9	813	146	
15	9	421	313	
16	10	817	147	
17	10	417	313	
18	11	821	148	
19	11	416	314	
20	12	826	149	
21	12	414	314	
22	13	830	149	
23	13	414	315	

Combined_video_results

Video CSV table

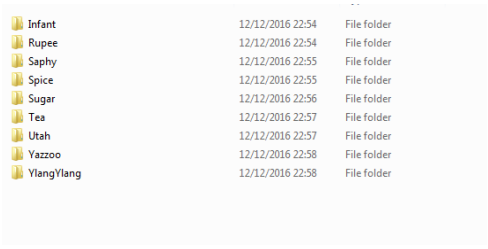
6.2 Recognition

Clicking on “Recognition” will bring up two choices:

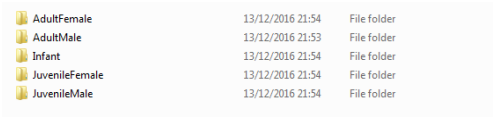


6.2.1 Train new face recognition model

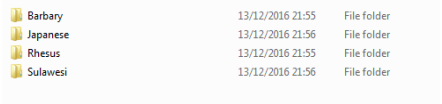
To train a new face recognition model sort your images into different folders based on identity, class or species (depending on the aim of the model). The image below shows the folder structure for the longitudinal_group_model described in section 7. Each folder should contain a minimum of two images. For the best results a minimum of 20 images per class is recommended. The program will ask you for the parent directory and then ask for a file name to save the model as.



Folder structure to train face recognition model on different individuals

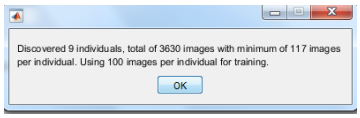


Folder structure to train face recognition model on different classes of animal (gender/maturity)



Folder structure to train face recognition model on different species

You will then be provided with a summary of the images:



Validation – the model can be validated in one of two ways. You will be asked which method you prefer.

1) Separate image set – use this option if you have a second image set of the same individuals/classes/species – for example you may want to train the model on images from one day and validate it on images from another day. The validation image set must have the same folder structure as the training image set.
2) Cross-validation – if there is no separate image set then five-fold cross-validation will be performed on the training image set. Please note that the results of cross-validation are not valid if there is only two images per individual/class/species.

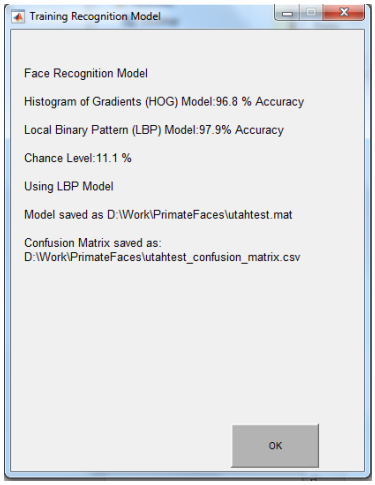
The validation results are given as the % images correctly classified by the model.

The program is set up to create two different face recognition models; the first uses local binary pattern (LBP) features and the second uses Histograms of Orientated Gradients (HOG). The program will choose the most accurate model for your situation based on the validation results. Both models use a support vector machine classification algorithm.

During training an information screen will provide you with details about the current stage of training. The four stages are:

1) Processing images
2) Extracting features from images
3) Training models
4) Running cross-validation/validation

The final screen will give details of the validation accuracy of the two models, the % images expected to be classified correctly by chance and details of the saved model. In addition to the saved model the confusion matrix produced as a result of validation will be saved as a csv file.



Example summary screen

	A	B	C	D	E	F	G	H	I	J	K
1		Infant	Rupee	Saphy	Spice	Sugar	Tea	Utah	Yazzoo	YlangYlang	
2	Infant	100	0	0	0	0	0	0	0	0	
3	Rupee	0	96	2	0	0	0	1	0	1	
4	Saphy	0	0	99	0	0	0	1	0	0	
5	Spice	0	0	0	99	0	0	0	0	1	
6	Sugar	1	0	0	0	98	0	0	1	0	
7	Tea	0	1	0	2	0	96	1	0	0	
8	Utah	0	1	0	0	0	0	99	0	0	
9	Yazzoo	0	2	0	1	0	0	1	96	0	
10	YlangYlan	0	1	0	0	0	1	0	0	98	
11											

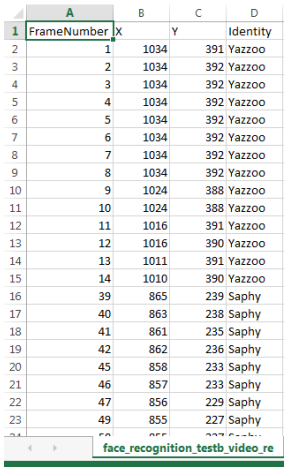
Example confusion matrix

6.2.2 Run recognition on images or videos

Requirements:

1) Face detection model (xml file; use existing or train own)
2) Face recognition model (mat file).
3) Folder containing images or videos. The following formats are accepted JPEG, TIF and PNG for images and AVI, MP4, WMV and MOV for videos.

The setup for face recognition is almost identical to that for face detection (see section 6.1.2). After either creating a new setup file (steps a-e) or loading an existing setup file you will be asked for the face recognition model file. The main output from face recognition are csv files as for face detection. These contain an additional fourth column containing identity. You will be asked if you also want to save the processed image/video files.



	A	B	C	D
1	FrameNumber	X	Y	Identity
2	1	1034	391	Yazzoo
3	2	1034	392	Yazzoo
4	3	1034	392	Yazzoo
5	4	1034	392	Yazzoo
6	5	1034	392	Yazzoo
7	6	1034	392	Yazzoo
8	7	1034	392	Yazzoo
9	8	1034	392	Yazzoo
10	9	1024	388	Yazzoo
11	10	1024	388	Yazzoo
12	11	1016	391	Yazzoo
13	12	1016	390	Yazzoo
14	13	1011	391	Yazzoo
15	14	1010	390	Yazzoo
16	39	865	239	Saphy
17	40	863	238	Saphy
18	41	861	235	Saphy
19	42	862	236	Saphy
20	45	858	233	Saphy
21	46	857	233	Saphy
22	47	856	229	Saphy
23	49	855	227	Saphy

face_recognition_testb_video_re

Face recognition csv results for a video file

7. Resources

The PrimateFaces_program.zip folder contains a resources folder that will allow you to test the face detection and recognition on a video of rhesus macaques. The files provided are:

a) Video File: longitudinal_group_video.mp4
b) Face Detection Setup File: detection_setup.mat
c) Face Recognition Model: longitudinal_group_model.mat

The PrimateFaces_program.zip folder contains a models folders which has a variety of different detection models. These have been included to provide users with a starting point for face detection. The files provided are:

a) RhesusFaceModel.xml (rhesus macaque face detector)
b) RhesusEyePairModel.xml (rhesus macaque feature detector – pair of eyes)
c) RhesusSingleEyeModel.xml (rhesus macaque feature detector – single eye)
d) RhesusNoseModel.xml (rhesus macaque feature detector – nose)
e) PrimateFaceModel.xml (primate face detector – trained on wide range of species)
f) LemurFaceModel.xml (lemur face detector – trained on range of lemur species including ring-tailed lemurs)
g) NewWorldFaceModel.xml (new world monkey face detector – trained on range of new world species including marmosets, tamarins and capuchins)*
h) OldWorldFaceModel.xml (old world monkey face detector – trained on range of old world monkey species including macaques, baboons and mandrills)
i) GibbonFaceModel.xml (gibbon face detector – trained on range of gibbon species including Lars gibbons)
j) ApeFaceModel.xml (ape face detector – trained on gorillas, chimpanzees and orang-utans)*

* Neither of these detectors was particularly effective when tested but were included here for completeness.

8. Troubleshooting

In the same folder as the application will be a log file (always labelled PrimateFaces_EventLog). If there are any problems using the application the log file will contain a list of any errors encountered. Please contact claire.witham@ncl.ac.uk for further assistance.

9. Advanced

For users with some programming knowledge we have included the original matlab scripts used for the PrimateFaces tool. These consist of the main function file (primatefaces_main.m) and five other dependent function files. The program can be run from the command line by running the primatefaces_main function. There are a number of settings that can be modified for both detection and recognition. The default settings provided are those we have found work best for rhesus macaques.

9.1 Online face detection and recognition

The basic process for face detection and recognition (in primatefaces_process.m) can be used for online face detection and recognition. An USB camera or webcam can be accessed in Matlab through the image acquisition toolbox (for some models of camera an additional plugin will be required). Stills acquired from the camera can be processed in the same way as offline image and video files. For a USB 2.0 camera with standard definition frame rates of ~15 fps can be achieved.

9.2 Recent advances in face detection and recognition

Face detection and recognition are rapidly evolving fields with new techniques described on a regular basis. Some of these techniques may help with problems such as occlusion and rotation. The software provided here uses fairly basic techniques. If you are interested in newer techniques we recommend consulting a computer vision expert.

10. Citation

Please cite any use of the PrimateFaces tool as ***.