TSBB11 – GAIT

User Guide

Magnus Selin (magse761) Kevin Kjellén (kevkj515)
Rolf Lifvergren (rolli107) Christoffer Malmgren (chrma018)
John Stynsberg (johst529)

December 17, 2017

1 Introduction

This is the user guide to a gait analysis system. The system has several methods to track feet in a video. It uses those tracks to estimate in which frames the feet touch the ground, and the gait parameters: duty factor, stance duration and swing duration. The system is intended to be used for medical statistics or analysis. In particular, it might be useful in the process of fitting a new prosthesis to a patient by detecting asymmetries in the patient's gait. To read more about how the system works read the technical documentation[1].

2 Prerequisites

To run the marker-based tracking method, you need circular markers in bright colors which stand out from the background and analysis subject in the video. Bright blue and pink markers have previously been used successfully. Using different colors for markers on different feet gives more robustness but is not a requirement. The markers that were used in development were paper circles with a diameter of around 6 cm.

To run the OpenPose-based tracking method your computer needs a NVIDIA graphics card with at least 1.6 GB graphics memory available capable of running CUDA 8 and cuDNN. To use the experimental tracking method based on a convolutional neural network, a CUDA-capable graphics card is strongly recommended.

The system uses a video file (or a sequence of images) as input. Thus some device that can record video is necessary in order to generate input, for example a smartphone equipped with a video camera.

2.1 Software requirements

To run the software Python 3.5 or later is required, as well as OpenCV (version 3.3.1 or later is recommended).

In addition to this, CUDA 8.0 and cuDNN required to use the OpenPose-based tracking method. OpenPose[2] then needs to be compiled prior to running the software. The documentation for OpenPose describes how to compile that library, which includes the installation of CUDA and other prerequisites.

[1] TSBB11 -– GAIT Technical Documentation

[2] `https://github.com/CMU-Perceptual-Computing-Lab/openpose`

3 Installation

Installation is done by cloning the the project's git repository[3]. Install the required Python packages in a Python virtual environment by standing in the repository in a terminal and running `pip install -r requirements.txt`.

4 Usage

To start the program run the file `gait_analysis.py` inside the virtual environment in which the required Python packages are installed (see Section 3).

When running the program the you is presented with a window as in Figure 1.

The choice if which video to analyze and if a marker or a markerless tracker should be used. If you have analyzed a video before, there is an option to use cached tracking data. If you are using a marker tracker, you have to specify the number of trackers you want to use.

When pressing `EXECUTE` the tracking starts. If you are using marker-tracking you have to set the thresholds which is described in section 4.1.

4.1 Using the color tracker

The color tracker is a blob detector operating on a image thresholded in the HSV colorspace. You can choose the thresholds for the color tracker in a graphical interface, see figure 2.

Click at the markers in the picture to create a initial threshold. Thresholded parts of the image will be marked in white. Using the frame slider you can verify that the threshold is reasonable throughout the entire video. Individual thresholds might be changed with the sliders. Pressing `q` will fix the selected threshold interval, and start the threshold selection for the next tracker. When all thresholds have been specified, this starts the analysis of the video.

4.2 Annotation tool

There is also a manual annotation tool, which can be used by running one of the files `annotate_updown.py` or `annotate_position.py`, depending on which kind of data you

[3] `https://github.com/gait-cdio/gait`

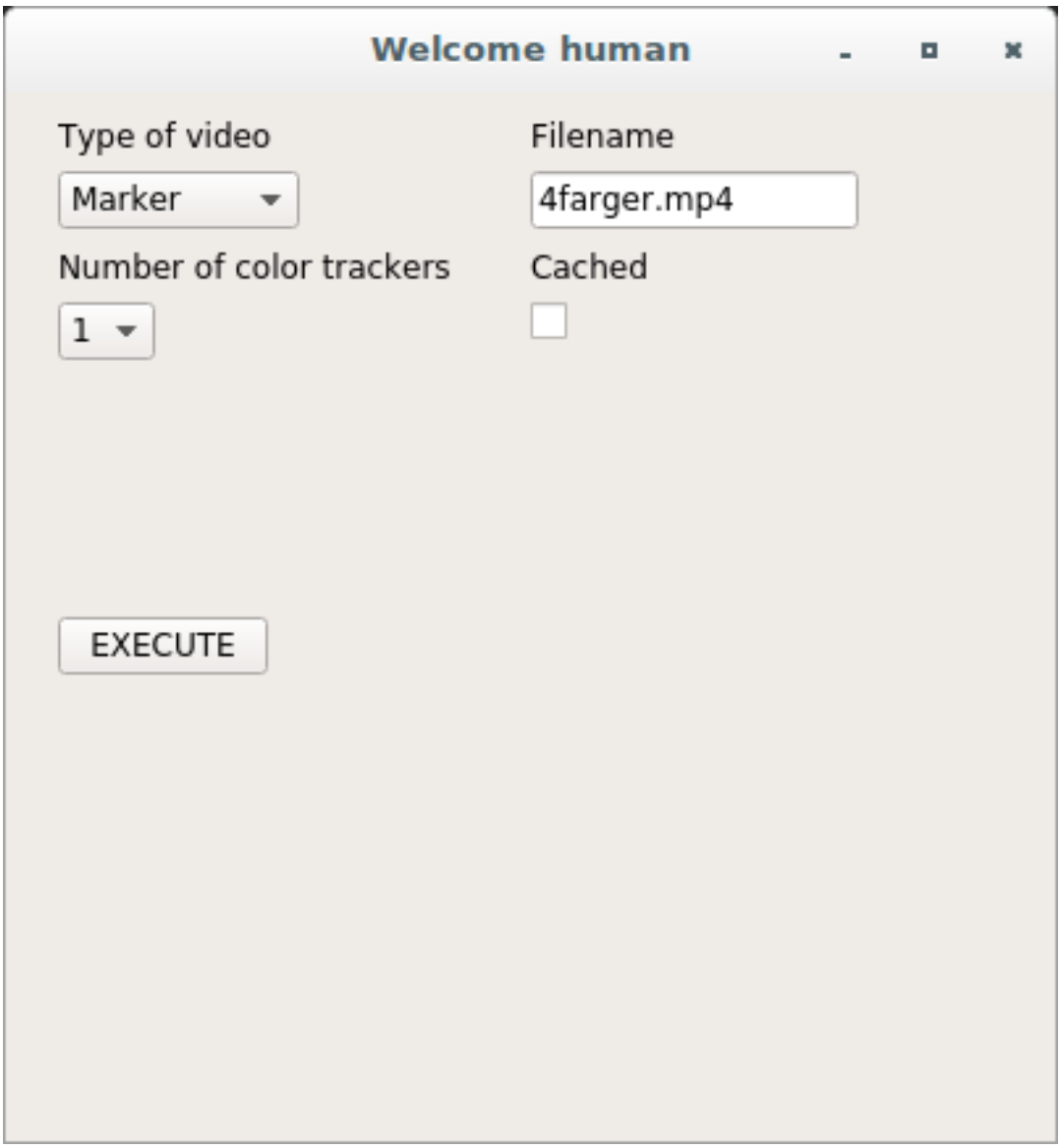

Figure 1: The start window.

want to annotate. It is not part of the end product but might be useful for the interested user. There are instructions in the programs.

5 Output

The system will show a gait diagram, indicating when each foot was in the ground. An example of such a diagram is seen in Figure 3.

The gait parameters can be found in the folder `output-data/`. It is a yaml file with the same name as the input video, for example `subject007.yaml`. The format of the output can be seen in figure 4.

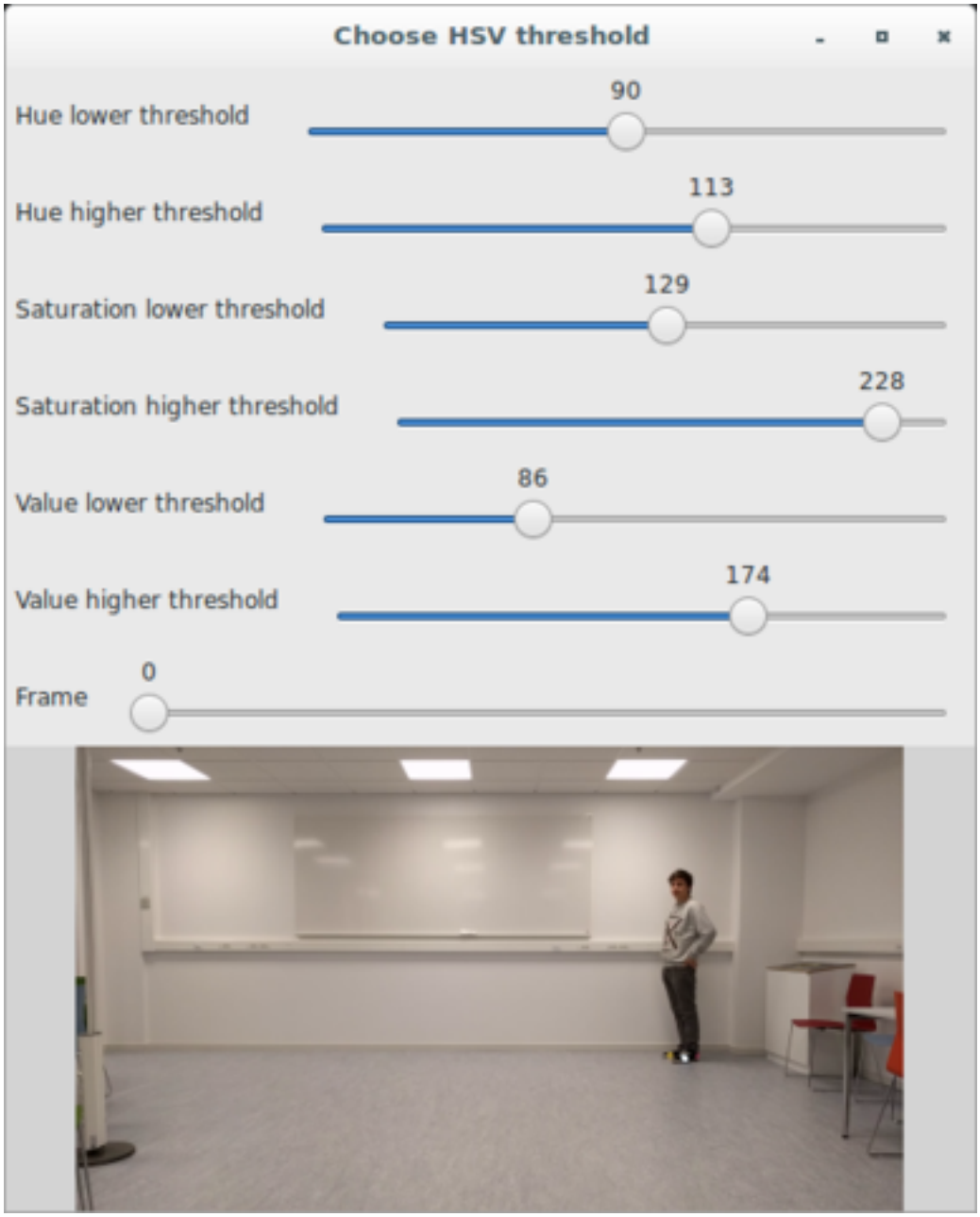

Figure 2: Interface for choosing thresholds for marker-based keypoint tracking.

All estimations of in which frame the foot touches the ground can also be found in this folder. It is a numpy-file with the same name as the input video. The notation for lifted foot is 0 and for foot on ground is 1.

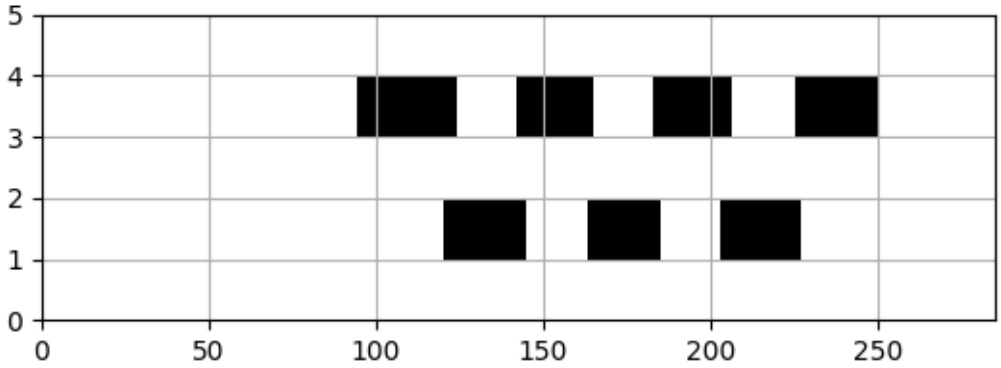

Figure 3: An example of a gait diagram. The left foot is always the lower track.

```
duty_cycle:
  left: 0.43650793650793646
  right: 0.39880952380952384
stance_duration:
  left: 0.6111111111111112
  right: 0.5583333333333333
swing_duration:
  left: 0.7888888888888889
  right: 0.8416666666666667
```

Figure 4: An example of output from the system.