
# Teardrop - a video analysis software

Alexander Isenko

# 1 Description

**trdrop** - pronounced [*'teə(r),drɑp*], is a marvelous video analysis software. It can calculate the frame rate of a raw input video, show frame tears, visualize the result and export it into a youtube friendly format.

**trdrop_lib** is the core library which provides an interface to create a command line and a GUI interface for the provided functionality.

**trdrop_c** is the command line interface which will be configurable through a config file and/or flags. The output can be streamed while being processed from VLC to get a preview.

**trdrop_v** is the GUI interface which will be configurable through user interaction. The output can be shown in a custom window with additional layout options. This will only be done if the time constraint allows it.

# 2 Formal description

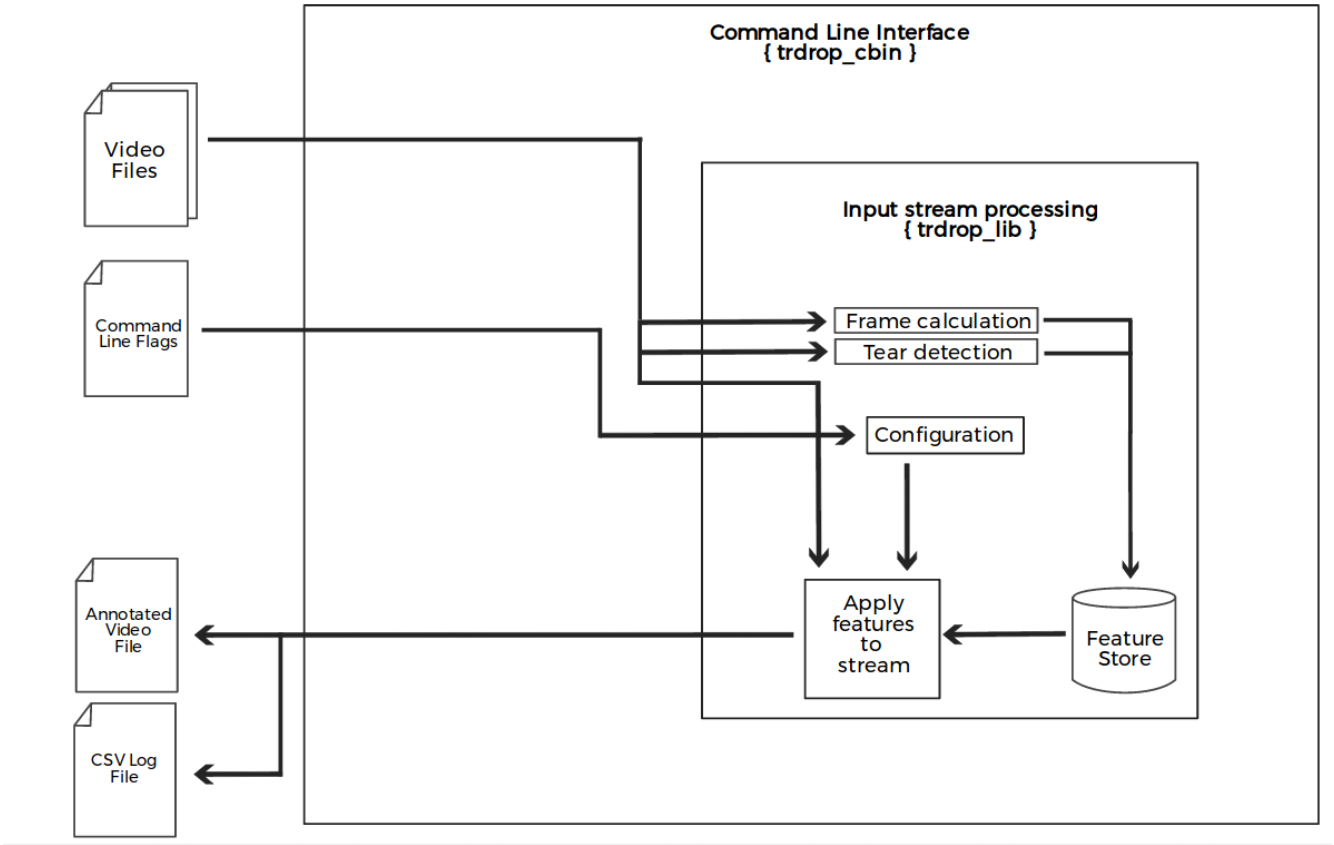
This project covers several themes of C++, mainly defined under the umbrella term *offline feature extraction of big raw video data*. The task consists of creating a streaming interface to being able to process multiple GB sized videos, apply the feature detection and encode everything into a single video in constant space complexitiy.

## 2.1 Functionality

The following features are to be included in **v0.1**:

- determine the real fps of the incoming video files
- show the fps as text in the video
- import up to 4 `.raw` video files with a size greater than 150 `GB`
- export the resulting video into a youtube friendly format (google-terms)
- a log file is created with the framerate and every tear + fps

### 2.2 Program Diagram




## 3 Example usage

### 3.1 Command Line Interface

```
# Creates a new annotated video with defaults
#
$ trdrop_cbin video_01.raw > converted_video.mp4

# Creates a new annotated video and VLC is used to visualize the result
#
$ trdrop_cbin video_01.raw > converted_video.mp4 | vlc

# Creates a new annotated video from multiple inputs
#
$ trdrop_cbin video_01.raw video_02.raw video_03.raw video_04.raw > converted_video.mp4
```

## 4 Future work