# Video Shot and Scene Segmentation

**Version 1.4 Documentation**

---

# 1 Introduction

This is a tool for automatic temporal segmentation of videos into shots and scenes. Shots are sequences of consecutive frames captured without interruption by a single camera. The transition between two successive shots of the video can be abrupt, where one frame belongs to a shot and the following frame belongs to the next shot, or gradual, where, two shots are combined using chromatic, spatial or spatial-chromatic video production effects (e.g., fade in/out, dissolve, wipe), which gradually replace one shot by another. This tool is capable of detecting both types of transitions. Scenes are higher level temporal fragments that correspond to the story-telling parts of the video. They are formed by grouping the detected shots into semantically coherent temporal video fragments. Parallel computing, via multi-threading, is used for making the whole video analysis several times faster than real-time processing.

# 2 Usage instructions

## *2.1 Installing and running the Ubuntu Linux version*

If you received the Ubuntu Linux version of the video shot and scene segmentation software, please follow the installation and execution steps below. The following steps where tested on Ubuntu 16.04 LTS, but should work with other Ubuntu-based distributions and some Debian-based.

**i) Install OpenCV**

To run the software you should have OpenCV library installed. You can do so by installing the libopencv-dev package using the package manager of your preference (apt, synaptic). For example, using apt, open a terminal and run:

```
$ sudo apt-get install libopencv-dev
```

which will install the OpenCV library along with other dependencies. After that you are ready to run the software.

**ii) Run the software**

Navigate with the terminal to the video_segmentation directory and run:

```
$ ./video_segmentation <video_name> [--args]
```

where `<video_name>` is the name of the video you want to process. For example: `./video_segmentation video.mp4` . Please note that the video should be on the same directory with the executable or else you should provide the FULL path to the video. There are sevral optional arguments that can be used, listed below.

Arguments list:

`--kf_path`
: The path to extract the keyframes

`--kf_num`
: The number of keyframes to be extracted for each shot (default 3)

`--trans_type`
: Adds an integer identifier of the transition type to the shots results file, 0 denoting abrupt, 1 dissolve and 2 wipe

`--shots`
: Skip scene segmentation and perform only shot segmentation

`--kf_shot`
: Extract shot keyframes

`--kf_scene`
: Extract scene keyframes

## *2.2 Software output*

After the processing is finished, the folder where the video was placed will contain four newly created results files and folders:

- The file **"<video_name>_shots.txt"** contains information about the automatically detected shots. For each shot the software stores a new line with information about the starting and ending frames of the shot, plus three intermediate frames that can be used as representative keyframes of this shot in other visual analysis tasks (e.g., for video concept detection and video event detection). If the [--trans_type] flag is set, the file will include the transition type identifier (0: abrupt, 1: dissolve, 2: wipe) for each shot in the end of each line.
- The file **"<video_name>_shots.srt"** contains the automatically detected shots in the format of video's subtitles, enabling the user to check visually the produced output (after renaming the file as the name of the video, and using e.g. the VLC player, or other players)
- The file **"<video_name>_scenes.txt"** contains information about the automatically detected scenes. For each scene the software stores a new line with seven (7) numbers. The first two represent the starting and ending frames

of the shot, while the next five numbers are the five most representative keyframes of the scene. If the last two numbers of a scene are zero this means that the scene consists of only one shot, thus we list only three keyframes.

- The file **"<video_name>_scenes.srt"** contains the automatically detected scenes in the format of video's subtitles, enabling the user to check visually the produced output (after renaming the file as the name of the video, and using e.g. the VLC player, or other players).
- If the [-kf_shot] option is selected, the folder **"<video_name>_Shot_Keyframes"** will be created, containing the 3 keyframes of each shot in jpeg image format, named "shotx_y.jpg", where x is the shot number and y is the keyframe number in the shot (1, 2 or 3).
- If the [-kf_scene] option is selected, the folder **"<video_name>_Scene_Keyframes"** will be created, containing the 5 (or 3) keyframes of each scene in jpeg image format, named "scenex_y.jpg", where x is the scene number and y is the keyframe number of the scene (1,2,3,4 or 5).

### *2.3 Software limitations*

The software was tested, in terms of compatibility, using various video formats and codecs, including MPEG, MP4, AVI, WMV, MOV, WEBM, FLV. In addition to these, it may work with the other video formats and codecs supported by FFmpeg, although it should be noted that no such testing has been performed.

# 3 Copyrights

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* The name of the copyright holders may not be used to endorse or promote product derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the Intel Corporation or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.

The implementation of certain components used by this software was based on the Intel OpenMP runtime library. Below is the original copyright.



Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

* Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

* Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.

* Neither the name of Intel Corporation nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

This software is provided by the copyright holders and contributors "as is" and any express or implied warranties, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose are disclaimed. In no event shall the copyright holder or contributors be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption) however caused and on any theory of liability, whether in contract, strict liability, or tort (including negligence or otherwise) arising in any way out of the use of this software, even if advised of the possibility of such damage.