ClassX Slide Synchronization User Guide (v.1.1)

Ngai-Man Cheung
ncheung@stanford.edu
16 March, 2012

I. Overview

This user guide discusses the ClassX slide synchronization program. Automatic slide synchronization makes possible display of electronic slides alongside relevant sections of the lecture video. When a slide transition occurs in the lecture video, the slide image is updated accordingly. Slide synchronization also enables the user to select a slide and access the time segment of the video where the selected slide is discussed.

ClassX slide synchronization is based on key-frame extraction and local feature matching. Automatic slide recognition is performed offline and involves two steps: 1) extraction of keyframes from the video and 2) matching keyframes to a deck of slides using pairwise image comparison.

II. Required software

ClassX slide synchronization requires the following software:

- Cmake. This is also required to build OpenCV. Available at http://cmake.org/cmake/resources/software.html, or by:
  ```
  $sudo apt-get install cmake
  ```
- OpenCV. Available at http://sourceforge.net/projects/opencvlibrary/. Installation OpenCV instruction can be found here: http://opencv.itseez.com/doc/tutorials/introduction/linux_install/linux_install.html#linux-installation
- ImageMagick. Available at http://www.imagemagick.org/script/download.php, or by:
  ```
  $sudo apt-get install imagemagick
  ```
- FFmpeg. Available along with ClassX Encoder code
- VLFeat SIFT. Distributed along with ClassX slide synchronization code
- ANN Approximate Nearest Neighbors Library. Distributed along with ClassX slide synchronization code

III. Building the programs

Slide synchronization code resides in the `<ClassX>/slideRecognition` directory. There are four executables: `changeDetect`, `slideMatch`, `image_matching`, `sift`. To build the executables, simply change into the slideRecognition directory and compile the programs:

```
$ cd <ClassX>/slideRecognition
$ make clean
$ make all
```

IV. Usage

As discussed, slide synchronization involves two steps:

(i) Key-frame extraction:

```
changeDetect <para_file>
```

This extracts video frames at slide transition points. The extracted video frames reside in the working directory specified in the parameter file `<para_file>`. See format of `<para_file>` below.

(ii) Slide matching:

```
slideMatch <para_file>
```

This matches the extracted key-frames with the slide deck. See format of the matching results below.

V. Parameter file format

`<para_file>` is the same in key-frame extraction and slide matching, with the following parameters in each row of the file:

```
<Frame_width>
<Frame_height>
<SIFT_location>
<Image_matching_location>
<Slide_images_folder>
<Number_slide>
<Thres_early_termination>
<Slide_image_width>
<Slide_image_height>
<SlideMatch_location>
<Input_video>
<Thres_slide_change>
<Working_dir>
<Slide_number_no_match>
<Number_frame>
```

```
<Slide_loc_top_left_x>
<Slide_loc_top_left_y>
<Slide_loc_bot_right_x>
<Slide_loc_bot_right_y>
```

The meanings of the parameters are as follow:

Parameter	Meaning
`<Frame_width>`	Input video frame width, e.g., 1920 for 1080p video.
`<Frame_height>`	Input video frame height, e.g., 1080 for 1080p video.
`<SIFT_location>`	SIFT executable location; SIFT will be called by `changeDetect` and `slideMatch`, and `<SIFT_location>` indicates the location.
`<Image_matching_location>`	Program `image_matching` executable location.
`<Slide_images_folder>`	Folder where slide images and features reside. Slide feature files follow this convention: `000.sift, 001.sift, …`
`<Number_slide>`	Number of slide images in the slide deck.
`<Thres_early_termination>`	Threshold for early termination in `slideMatch`. Program `slideMatch` performs a two-step search for slide matching, and this parameter specifies the condition to terminate the search after the first step. A small value can speed up the search but could affect the accuracy. Suggested value: 50
`<Slide_image_width>`	Slide image width
`<Slide_image_height>`	Slide image height
`<SlideMatch_location>`	Program `slideMatch` location
`<Input_video>`	Input video bitstream
`<Thres_slide_change>`	Threshold for slide change detection in video. A large value can speed up the search but may result in missing some slide change event. Suggested value: 12
`<Working_dir>`	Working directory. Intermediate files reside here.
`<Slide_number_no_match>`	Optional: Slide number in the result file if no matching slide is found for the corresponding key-frames. Potentially there is no slide showing in the video during this duration. Default value: 0

`<Number_frame>`	Optional: Number of frame to be processed. Default: Process the entire video.
`<Slide_loc_top_left_x>`	Optional: Slide area in video, top-left x. Will be determined automatically if not specified.
`<Slide_loc_top_left_y>`	Optional: Slide area in video, top-left y. Will be determined automatically if not specified.
`<Slide_loc_bot_right_x>`	Optional: Slide area in video, bottom-right x. Will be determined automatically if not specified.
`<Slide_loc_bot_right_y>`	Optional: Slide area in video, bottom-right y. Will be determined automatically if not specified.

Below is an example of a parameter file:

```
1920
1080
/home/ubuntu/Pipeline_Resources/SlideSynchronization/sift
/home/ubuntu/Pipeline_Resources/SlideSynchronization/image_matching
/home/ubuntu/Demo/encoded/SlideSyncWorkDir/SlideDeck/
4
50
776
582
/home/ubuntu/Pipeline_Resources/SlideSynchronization/slideMatch
/home/ubuntu/Demo/Content/00068.mp4
12
/home/ubuntu/Demo/encoded/SlideSyncWorkDir/
```

VI. Output result format

Output in `resultsOut.txt` resides in the working directory `<Working_dir>` specified in the parameter file, with the following format in every row:

`<Frame_num> <Slide_num> <Num_Post_RANSAC_Matches> <Match_Ratio>`

`<Frame_num>`	Frame numbers of the key-frames extracted from the input video.
`<Slide_num>`	Matching slide of the key-frames.
`<Num_Post_RANSAC_Matches>`	Number of correspondences between key-frames and matching slide images. The more is the number of correspondence, the more accurate is the matching result.
`<Match_Ratio>`	The ratio of the number of correspondences between the second best matching slide to the best matching slide. The smaller is the ratio, the

		more accurate is the matching.	

Below is an example of `resultsOut.txt`:

```
0           0           118         0.262712
2010        1           164         0.000000
2130        1           145         0.000000
```

In this example, frame 0 matches slide 0, frame 2010 and 2130 match slide 1.