Appendix A

Week of Science and Techology Lecture Dataset

The data for most of our ongoing experiments have been acquired on the Week of Science and Technology. This festival includes many lectures from various disciplines.

The lectures were recorded by MediaSite device, which enables capture and streaming of audiovisual signal with pictures of currently shown slides. These slides are processed in such a way that major difference in the incoming signal triggers a capture of a new slide as a picture. This way, we gather precise information on presentation timing as well.

File structure of recorded lecture is shown in Figure A.1.

In our pre-processing stage, the `MediasitePresentation_50.xml` file was decoded and timing information was gathered. This way, we have cut the video recording in file `<unique_hash>.wmv` file accordingly for later processing alongside the slide picture.

Sound has been cut with notion of pauses however, always in the places with the lowest loudness. Overhanging pieces of audio were assigned to both slides.

Sound information has been then passed without further modifications to automated speech recognition algorithm of Google, at the time opened for public.

Pictures of slides were thresholded for ease of letter boundary recognition and passed

- `MediasitePresentation_50.xml` ..... metadata, including slide timing
- `Player.html` ..... HTML page with player, not relevant
- `App_Themes` ..... the directory of helper files for player, not relevant
- `Content` ..... the directory of multimedia content
  - `<unique_hash>.wmv` ..... audiovisual recording
  - `slide_<nnnn>_full.jpg` ..... full slide image
  - `thumbnail_<unique_hash>.jpg` ..... thumbnail image shown in catalogue
- `PlayerOptions` ..... the directory of player branding, not relevant
- `Players` ..... the directory of player itself, not relevant

Figure A.1: Data structure of Mediasite Publish2Go package

to system Tesseract for optical character recognition. Acquired words were filtered against a dictionary.

All textual data have been stemmed and removed of stop words.

Visual information from the camera was indexed by a SURF algorithm and a relative histogram have been created.

The resulting information have been stored in a JSON file for later processing and classification in MATLAB.

Example of the JSON file structure follows:

```
{
  title,                    Name of the lecture
  description,              Textual description
  duration,                 Duration in ms
  slides: [                 Set of slides
    number,                   Number of the slide
    start,                    Starting time
    end,                      Ending time
    ocr,                      Text recognized by OCR
    ocr_stemm,                Stemmed version of above
    audio: [                  Set of audio snippets
      result: [                 Transcription result
        alternative: [            Set of alternatives
          transcript,               Actual transcript
          confidence                Confidence of the result
        ],
        final                     If the transcription is final
      ],
      result_index              Which alternative is preferred
    ],
    audio_stemm: [],          Set of stemmed transcriptions
    audio_stemm_alter: [],    Set of stemmed alternatives
    histogram: [],            Histogram of the video, 256 * RGB
    surf                      SURFs detected in the video frame
  ]
}
```