ActEV Evaluation JSON Formats

Version: 2.0

Date: 2020-08-25

ActEV Team

NIST

Table of Contents

Background

The ActEV Independent evaluations in 2019 will use the Multiview Extended Video with Activities (MEVA *http://mevadata.org*) test video dataset for the activities as defined in the annotation guide [1]. NIST is encouraging the crowd sourcing of annotations to greatly enhance the amount of annotated MEVA training data for the activities and share them with the ActEV research community. This document describes the requirements a "sharer" must adhere to and the common data files for exchanging data. By now, you should be familiar with:

- The MEVA video data
- Reviewed the annotation exemplars
- Reviewed the annotation guidelines for both the tracked object types and the activity definitions.

To annotate a video, one must temporally localize each activity instance (in terms of the start/end the performed activity) and spatially annotate bounding boxes of object(s) associated with the activity instance.

Annotators can use any of the video annotation tools used by the video analytics community, some of the commonly used ones are described in [2][3][4][5][6][7][8][9]. Since most video annotation tools have their own output format, annotation sharers are responsible for translating their annotations into a common format, where all the data can be aggregated, compared and merged to create a training dataset.

The common annotation data format is a set of three, JSON-formatted data files that describe the video files annotated, the activities annotated within the videos, and activity instance annotations themselves.

By convention, the names of three files are:

```
file-index.json
activity-index.json
activities.json
```

Section 1 describes each of the three data file formats and the content they hold.

1. Annotation Files

Annotations are represented by three file types. (1) The **file-index** defines the list of video files annotated and metadata about the video files including the annotated frames. (2) The **activity-index** enumerates the list of annotated activities and metadata about the activities including the annotated objects. The textual description of the activities and objects are contained in the "ActEV Annotation Definitions for MEVA Data"[1]. (3) the **activities** file contains the annotations for each instance of an activity.

Each triplet of files forms a coherent set of annotations meaning a triplet could represent a single file of a single activity or multiple files for multiple activities.

These definitions will be used throughout the document:

- "**Activity Instance"**: an observed instance of activity. It could be visibly present within a single camera view or across multiple camera views (if an annotation regime supports multiview annotation which is optional).

- "**Frame State Signal**": a signal-based representation of a given variable's "state" at frame X which continues until the "state" changes. The data structure is a dictionary with keys being a frame number and the value being the state. The value can be either a simple data type or another dictionary.
- **"<value>": a dictionary key that is a name, e.g., a file name, that must be unique within the given dictionary**

1.1. File Index

The file index JSON is a three-level dictionary with first being indexed by the video source file's name and the second level representing metadata about the file. An example, along with an explanation of the fields is included below. The file JSON contains several pieces of metadata from the MEVA camera calibration data found in the MEVA data repository [10].

```
{
   "2018-03-07.16-55-00.17-00-00.admin.G329.avi": {
       "facility_id": "KF1",
        "begin_time": "16-55-00",
        "camera_id": "G329",
        "camera_model": "reolink",
        "camera_resolution": "1920x1072",
        "camera_set_id": "3-329",
        "camera_type": "EO_NIR",
        "clip_id": "2018-03-07.16-55-00.17-00-00.admin.G329",
        "date": "2018-03-07",
        "end_time": "17-00-00",
        "filename": "./2018-03-07.16-55-00.17-00-00.admin.G329.avi",
        "frame_offset_precision_to_reference_clip_id": 0,
        "frame_offset_to_reference_clip_id": 0,
        "framerate": 30,
        "krtd_filename": "",
        "ply": {
               "hi-res": "./mutc-3d-model/mutc.ply",
               "low-res": "./mutc-3d-model/mutc_coarse.ply"
        },
        "recording_site": "indoor",
        "reference_camera_id": "G329",
        "reference_clip_id": "2018-03-07.16-55-00.17-00-00.admin.G329",
        "reference_file_index_key": "2018-03-07.16-55-00.17-00-00.admin.G329.avi",
        "selected": {
               "1": 1,
               "9001": 0
        },
        "slot_date_time": "2018-03-07.16-55-00"
   },
    "2018-03-07.16-55-00.17-00-00.school.G422.avi": {
       "facility_id": "KF1",
        "begin_time": "16-55-00",
        "camera_id": "G422",
        "camera_model": "reolink",
        "camera_resolution": "1920x1080",
        "camera_set_id": "3-421",
        "camera_type": "EO_NIR",
        "clip_id": "2018-03-07.16-55-00.17-00-00.school.G422",
        "date": "2018-03-07",
        "end_time": "17-00-00",
        "filename": "./2018-03-07.16-55-00.17-00-00.school.G422.avi",
        "frame_offset_precision_to_reference_clip_id": 30,
        "frame_offset_to_reference_clip_id": 0,
        "framerate": 30,
        "krtd_filename": "",
        "ply": {
                "hi-res": "./mutc-3d-model/mutc.ply",
                "low-res": "./mutc-3d-model/mutc_coarse.ply"
        },
        "recording_site": "indoor",
        "reference_camera_id": "G421",
        "reference_clip_id": "2018-03-07.16-55-00.17-00-00.school.G421",
        "reference_file_index_key": "./2018-03-07.16-55-00.17-00-00.school.G421.avi",
        "selected": {
                "1": 0,
                “11”: 1,
                "9002": 0
        },
        "slot_date_time": "2018-03-07.16-55-00"
 }
}
```

- <file>: A string for the name of the video file without a path and including the extension.
  - begin_time: The beginning time of the recording in the form “HH-MM-SS”,
  - camera_id: The camera id of the video,
  - camera_model: The camera model of the file. See the MEVA Data Repo [10] for values,
  - camera_resolution: The resolution of the file in the form "<width>x<height>",
  - camera_set_id: The camera set the camera belongs in the form “[3|5]_<camera_id>”. See the MEVA Data Repo [10] for values.
  - camera_type: The type of video recording with value “EO”, “IR”, or "EO_NIR",
  - clip_id: The clip_id of the file which is the name of file without the extension,
  - date: The collection date of the file in the form “YYYY-MM-DD”,
  - end_time: The end time of the recording in the form “HH-MM-SS”,
  - filename: The path and filename for the file relative to the main data set directory,
  - facility_id: The ID if the facility. Presently defined IDs are “UF1” (for “Unknown Facility” 1) or “KF1” (for “known Facility 1” aka the MEVA 1 data collection),
  - frame_offset_precision_to_reference_clip_id: The frame precision of the offset estimation from the reference camera,
  - frame_offset_to_reference_clip_id: The number of frames the camera is offset from the reference camera,
  - framerate: The frame rate of the video,
  - krtd_filename: The filename of the KRTD file for the video. MEVA Data Repo [10] for details. KRTD files may be available for UF1,
  - ply: A dictionary listing the location of the PLY files for the MEVA data. There are several PLY files defined on mevadata.org. The dictionary maps the name of the PLY file to the filename of the PLY file relative to the main data set directory. PLY files are not available for UF1,
  - recording_site: A generalization of the camera's recording site. Values “indoor” | “outdoor”,
  - reference_camera_id: The camera id of the reference camera for estimating frame offsets,
  - reference_clip_id: The clip ID of the video file used for estimating frame offsets,
  - reference_file_index_key: Contains the dictionary key WITHIN the current JSON for the reference file. Note: if the reference file is the file itself, then the key self-referencial.

- selected: a Frame State Signal with keys representing a frame number and the values being 1 (for annotated frames) and 0 (for unannotated frames) within the file <file>.
    - <framenumber>: 1 or 0, indicating whether or not the activity will be annotated for the given frame. Note that records are only added here when the value changes. For example in the above sample, frames 1 through 9001 in file "`2018-03-07.16-55-00.17-00-00.admin.G329.avi`" are selected for annotation. The default signal value is 0 (not-selected), and the frame index begins at 1, so for file "`2018-03-07.16-55-00.17-00-00.school.G422.avi`", frames 1 through 10 are not selected. Also note that the signal must be turned off at some point after it's been turned on.
- slot_date_time: The Slot Date Time value is a five-minute division of the recording day. The video file times are accurate to the second but rarely begin on these time boundaries so this value links together files that, for the most part, occur during the 5-minute time slice. The values are of the form "YYYY-MM-DD.HH-MM-SS".

1.2. Activity Index

The activity index JSON file lists the activities annotated for ALL files enumerated in the file index. The file is a two-level dictionary with first being indexed by the activity name (as defined in the MEVA Annotation spec and the second level representing metadata about the activity. It is assumed that if an activity is present in this file, then the activity annotation files below have been annotated for that activity (even if no instance of the activity is found.) An optional field "training" can also be found, for training a system on this activity type. An example, along with an explanation of the fields is included below.

```
{
   "person_enters_vehicle": {
      "objectTypeMap": {
         "person": "*person*",
         "vehicle": "*vehicle*"
      },
      "objectTypes": [
         "vehicle",
         "person"
      ]
  },
  "person_exits_scene_through_structure": {
      "objectTypeMap": {
         "person": "*person*",
         "vehicle": "*vehicle*"
      },
      "objectTypes": [
         "vehicle",
         "person"
      ]
  },
  "person_exits_vehicle": {
      "objectTypeMap": {
         "person": "*person*",
         "vehicle": "*vehicle*"
      },
      "objectTypes": [
         "vehicle",
         "person"
      ],
      "training": {
        "description": {
          "type": "text",
          "filename": "activities/annotation/act001.txt"
        },
        "exemplars": [
          {
            "domain": "extended_video",
            "filename": "activities/video/foo.avi",
            "annotation_type": "actev",
            "annotation": {
              "activities":
"activities/annotation/act001.activities.json",
              "file-index":
"activities/annotation/act001.file-index.json"
            }
          }
        ]
      }
  }
}
```

- <activity>: A string for the name of the activity as defined in the MEVA Annotation Spec.
    - objectTypeMap: Many objects are potentially annotated. This dictionary provides, for a given objectType, a regular expression for additional objects that are also considered the given objectType.
    - objectTypes: An array of strings enumerating the set of objects annotated for each instance with respect to the given activity.
    - training: Optional part. This dictionary provides information regarding training on an activity.
        - description:
            - type: The type of the description's content. Presently the only value is 'text' for an unformatted, ASCII text file.
            - filename: The path to the description.
        - exemplars: It has to be one instance per record.
            - domain: The domain of the video recording. Presently the only value is 'extended_video' which is a typical ActEV video.
            - filename: The path to the video containing the activity.
            - annotation_type: The type of annotation files for the instance. Presently, the only defined type is "actev" which is the format described in this document
            - annotation

    - activities: The path to the activities file containing instance annotations.
    - file-index: The path to the file-index file containing video metadata.

1.3. Activity Instance Annotation file

The activities JSON file lists the activity instances annotated for ALL files enumerated in the file index. Those file names of annotated files can be found under the “filesProcessed” key. The file is a two-level dictionary with first being indexed by the activity name (as defined in the MEVA Annotation spec and the second level representing metadata about the activity. It is assumed that if an activity is present in this file, then the activity annotation files below have been annotated for that activity (even if no instance of the activity is found.) An example, along with an explanation of the fields is included below.

The annotation output file should be a JSON file that includes a list of videos annotated, along with a collection of activity instance records with spatio-temporal localization information (depending on the annotation). An activity detection annotation output file is included inline below, followed by a description of each field.

The annotation output is shown for videos annotated, with a collection of activity instance records with temporal localization information only. An activity detection annotation output file is included inline below, followed by a description of each field.

Finally, for each processed file, an optional report dictionary can be added, that is used for auto-reporting the execution statuses.

```
{
  "filesProcessed": [
    "VIRAT_S_000000.mp4"
  ],
  "executionReport": {
    "fileStatuses": {
      "VIRAT_S_000000.mp4”: {
        "status": "success",
        "message": "hello world"
      }
    },
    "siteSpecific": {}
  },
  "activities": [
    {
      "activity": "Talking",
      "activityID": 1,
      "localization": {
        "VIRAT_S_000000.mp4": {
          "1": 1,
          "20": 0,
          "100": 1,
          "112": 0,
        }
      },
      "objects": [
        {
          "objectType": "person",
          "objectID": 1,
          "localization": {
            "VIRAT_S_000000.mp4": {
               “1": { “boundingBox”: { "x": 10, "y": 30, "w": 50, "h":
20 } }
               “20": {}
               “100": { “boundingBox”: { "x": 10, "y": 30, "w": 50, "h":
20 } }
               “104": { “boundingBox”: { "x": 60, "y": 60, "w": 50, "h":
20 } }
               “108": { “boundingBox”: { "x": 30, "y": 90, "w": 50, "h":
20 } }

               “112": {}

            }
          }
        }
      ]
    }
  ]
}
```

- filesProcessed: An array numerating the file names processed (either by annotators or a system) and reported on by this file. Every file, even if the file was unreadable or contained no activities, must be present in the array. The ‘executionReporting” dictionary below can be used to report anomalies.
- activities: An array of annotated activity instances. Each instance is a dictionary with the following fields:
    - activity: The name (e.g. “Talking”) from the MEVA Annotation Spec. [1]
    - activityID: a unique, numeric identifier for the activity instance. The value must be unique within the list of activity detections for all video source files processed (i.e. within a single activities JSON file)
    - localization: The temporal localization of the activity instance encoded as a dictionary of Frame State Signals indexed by the video file id(s) for which the activity instance is witnessed. Each Frame State Signal (for a video) has keys representing a frame number and the value being 1 (the activity instance is present) and 0 (otherwise) within the given file. Multiple Frame State Signals can be used to represent an activity instance being present in multiple video views. In this case, frame numbers are relative with respect to the video file.
    - objects: An array of objects annotated with respect to the activity instance. Each unique object is represented by the following dictionary:
        - objectType: A string identifying the objects type as one of the track types defined in the MEVA Annotation Spec.

        - objectID: unique, numeric identifier for the object.   The value must be unique within a single activities JSON file.
        - Localization: The temporal-spatial localization of the object encoded as a dictionary of Frame State Signals indexed by the video file id for which the object is witnessed. Each Frame State Signal (for a given video) has keys representing a frame number and the value is a dictionary describing the spatial localization of the object. The spatial dictionary has 1 key 'boundingBox' which is itself a dictionary described as a pixel 'x', 'y', 'w', and 'h' for the the x-position, y-position, width and height respectively. The (0,0) (x,y) position is the top left pixel.
- processingReport: An optional dictionary to report success or failures during the processing of the videos.
    - fileStatuses: A dictionary reporting success or failures while processing videos. The keys to the dictionary are the file names used in filesProcessed. All files need not be present.
        - <filename>
            - status: A text string indicating success or failure. The values must be "success" or "fail"
            - message: An additional text string to report additional information. The content is not restricted.
    - siteSpecific: An optional dictionary for which the system can store additional information. The structure and content of this dictionary has no restrictions.

References

1. ActEV Annotation Definitions for MEVA Data
2. Kitware annotation tool
3. The VGG Image Annotator
4. Scalabel (used for annotation of Berkeley DeepDrive project)
5. VATIC - Video Annotation Tool
6. BeaverDam
7. VoTT: Visual Object Tagging Tool
8. Computer Vision Annotation Tool (CVAT)
9. Efficient Annotation of Segmentation Datasets with Polygon-RNN++

10. MEVA Camera Calibration and metadata, MEVA Data Repo, https://gitlab.kitware.com/meva/meva-data-repo/-/tree/master/metadata

Disclaimer

Certain commercial equipment, instruments, software, or materials are identified in this evaluation plan to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by NIST, nor is it intended to imply that the equipment, instruments, software or materials are necessarily the best available for the purpose.