KWANT Documentation

Release 1

Kitware, Inc.

Oct 08, 2018

Contents

Contents:

CHAPTER 1

Introduction

The Kitware Analytics Toolkit (KWANT) is an open source C++ toolkit for computing scores and other metrics for object tracking systems.

Visit the repository on how to get and build the KWANT code base.

1.1 The Scoring Framework

The scoring code essentially consists of two executables, `score_tracks` and `score_events`. Other executables such as `hadwav_score_human_events`, etc., are derived from one or the other of these (or both); their functionality is planned to be merged into the baseline executables.

All the code uses [track_oracle](https://github.com/Kitware/kwiver/blob/master/track_oracle/README.markdown) in an attempt to be agnostic to the source file format.

The options for [score_tracks](README.scoretracks.markdown) and [score_events](README.scoreevents.markdown) are list on their respective pages; the general concepts behind their operation are described below.

Here, we use the term "event" to describe the association of a label with a track; the label typically describes the action of the actor whose track is being observed. The label vocabulary currently recognized is that of the VIRAT program plus generic "PersonMoving" and "VehicleMoving". The list of valid events can be found by running `score_events` with no arguments.

1.1.1 Basic scoring pipeline

Both `score_tracks` and `score_events` have the same basic sequence of steps:

1. The initial set of ground-truth tracks G0 and computed tracks C0 are loaded. The assertion is that all tracks will have a timestamp associated with each frame; if the track format does not support timestamps, various options are available to synthesize them. This code mostly lives in `score_tracks_loader.h` and associated classes.

2. Various optional spatio-temporal and event filters are applied to produce the final set of ground-truth and computed tracks, G and C. See `matching_args_type.h`.

3. An association matrix is computed by comparing the frames associated with each track in G with each track in C. Each pair of tracks is aligned based on timestamps and then checked for overlapping detections. Overlapping criteria are discussed below. The association matrix is computed by code rooted at `score_phase.h` and `phase1_parameters.h`.

4. Once the association matrix is available, various metrics are computed; see below.

1.1.2 Spatial Overlap Detection

Two methods are supported for determining if a pair of time-aligned detections overlap: *image* and *radial* overlap. The default is image-based overlap. Setting the `--radial-overlap` option to a number greater than zero turns on radial overlap.

1.1.2.1 Image-Based Overlap Detection

If detections are described using bounding boxes, which are in turn defined in terms of image coordinates on the image frame, then image-based detection is used. By default, two detections are said to overlap if their boxes overlap by at least one pixel, although options exist to change this.

In general, image-based overlap detection when scoring FMV tracking.

1.1.2.2 Radial Overlap Detection

When the `--radial-overlap N` option is given, where N is a number greater than 0, then two detections are declared to overlap if their centroids are no more than N meters apart. The detections are considered to be points, not boxes, and must be in lat/lon (such as a shapefile with geopoints or a kw18 file with world coordinates given as lat/lon.)

1.1.2.3 Converting Detection Overlaps To Track Overlaps

For any pair of tracks from G and C, once the number of overlaps has been determined, this is converted into a decision as to whether the tracks overlap. By default, they are said to overlap if a single detection is said to overlap, but again there are options to change this behavior.

1.1.3 Scoring events

When scoring an event, we assume that the event label applies to the entire track and that therefore if the event labels match and the tracks overlap, the events match. Some track formats (for example, XGTF) allow events to be defined over subsets of tracks; in these cases, we create new internal tracks for the subsets but record the original source track ID.

Event label matching is (almost) always strict equivalence; there is no concept of an event hierarchy. The exceptions are the PersonMoving and VehicleMoving events, which are specific to VIRAT and derived in a post-hoc fashion from the ground-truth (since the PersonMoving and VehicleMoving events were not annotated.)

1.1.4 Metrics

The track metrics produced by `score_tracks` are single points in Pd/FAR space; score_tracks does not have a concept of "partial match" which would serve as the operating point over which an ROC curve could be swept. Several overlap attributes could be used for this purpose, should the need or desire arise.

The event metrics are computed by generating the same track overlap matrix as for track metrics, which establishes the underlying track overlap profile, and then sweeping an operating point based on the event "relevancy" to declare overlapping tracks as hits or misses, and thus generate an ROC curve and/or a P/R curve. The relevancy measure is typically either the event probability, when scoring detectors, or the retrieval rank, when scoring retrieval.

Note that although the track association matrix is fixed when sweeping out an ROC curve for any particular event, the tracks making up the matrix can change depending on the event being scored. For example, an XGTF file can contain ground-truth for multiple events; when scoring "VehicleUTurn" events, the set of ground-truth UTurn tracks from the XGTF file will be different than those used when scoring (say) "PersonRunning."

1.1.4.1 Track Metrics

1.1.4.2 What the output means

The scoring code is moderately verbose. You may see warnings regarding zero-area boxes and timebases, such as this:

> [. . .] WARN Zero-area-box: file lair-gt.csv track 8776 frame 0 box <vgl_box_2d (empty)> WARN Zero-area-box: file lair-gt.csv track 8776 frame 0 box <vgl_box_2d (empty)> [. . .] WARN Timebases heading in different directions: file lair-gt.csv track 8776 . . . WARN Timebases heading in different directions: file lair-gt.csv track 8776 . . . [..]

They may be ignored.

The metrics are output at the end:

> **HADWAV Scoring Results:** Detection-Pd: 0.205511 Detection-FA: 697 Detection-PFA: 0.0591631 Frame-NFAR: 74.937 Track-Pd: 0.876833 Track-FA: 87 Computed-track-PFA: 0.136364 Track-NFAR: 21.9512 Avg track (continuity, purity): 1.56443, 0.986677 Avg target (continuity, purity): 2.52786, 0.365779 Track-frame-precision: 0.626884

The metrics are:

- ***Detection-Pd*: The ratio D/Td, where D is the number of computed** detections associated with a true detection, and Td is the number of true detections.
- ***Detection-FA*: The number of computed detections which were not** associated with any true detection.
- ***Detection-PFA*: The ratio Fd/Cd, where Fd is the Detection-FA count** above and Cd is the total number of computed detections.
- *Frame-NFAR*: Deprecated.
- ***Track-Pd*: The ratio Ct/Tt, where Ct is the number of computed** tracks associated with a true track; Tt is the number of true tracks.
- ***Track-FA*: The number of computed tracks which were not associated** with any true track.
- ***Computed-track-PFA*: The ratio Ft/C, where Ft is the Track-FA count** above and C is the total number of computed tracks.
- ***Track-NFAR*: The track false alarm rate, normalized to (by default)** tracks per minute per km^2.
- ***Avg track continuity*: The track continuity of a** computed track C measures the number of ground-truth tracks associated with C. Ideal value is 1.
- ***Avg track purity*: The track purity of a computed track C is the** percentage of detections in the lifetime of C which are associated with the "dominant" matching ground-truth track (if any.) The "dominant" matching ground-truth track is that ground-truth track which has the greatest number of associations with C. Ideal value is 100%.
- ***Avg target continuity*: Target continuity measures the number of** computed tracks associated with the ground-truth track.

- ***Avg target purity*: Symmetric to track purity; measures the** percentage of detections comprising a ground-truth track G which are associated with its dominant computed-track (if any).
- *Track-frame-precision*: Deprecated.

1.1.4.3 Event Metrics

CHAPTER 2

Indices and tables

- genindex
- modindex
- search