
Re: [ergysr/DeepCC] Does DeepCC auto learn overlapping footages? (#56)

6 messages

ergysr <notifications@github.com> 5 February 2019 at 01:38
Reply-To: ergysr/DeepCC
<reply+00521a7b0e1179487a82ad363c1d20306d7346182bdf664f92cf0000000118703a9392a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>
Cc: Nickson Yap <hi@nickson.me>, Mention <mention@noreply.github.com>

[@NicksonYap](#) The data where this code is evaluated doesn't contain spatially overlapping areas, so this repository sets the correlations to negative infinity for co-occurring detections, for simplicity.

You can replace the motion model to handle overlapping areas. I had previously done it for some soccer sequences with overlapping views, and for a non-public version of the Duke dataset, by setting correlations to positive if both motion, position, and appearance were similar. For overlapping views the motion model code becomes slightly more complicated, so this repository exposes a relatively simplified version.

—
You are receiving this because you were mentioned.
Reply to this email directly, [view it on GitHub](#), or [mute the thread](#).

ergysr <notifications@github.com> 5 February 2019 at 01:48
Reply-To: ergysr/DeepCC
<reply+00521a7b54995006d6818e9f3bccd04ff7e7c9182b6b7c6692cf0000000118703cd692a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>
Cc: Nickson Yap <hi@nickson.me>, Mention <mention@noreply.github.com>

It depends on your method. If it is very generic, your method would automatically handle disjoint or overlapping areas seamlessly. If it is more specific, you would need to take into account the specific areas of overlap, and handle overlapping and disjoint information separately.

Simply replacing the negative infinity correlation with a positive number will not work out of the box. It can happen that appearances look similar but motion is completely different, in which case a negative correlation makes more sense.

Essentially you will just need to replace the function which computes correlations to suit your needs.

[Quoted text hidden]

ergysr <notifications@github.com> 5 February 2019 at 02:05
Reply-To: ergysr/DeepCC
<reply+00521a7bb9c439baf135c1db6b2228d15a4c50dfca03fd0692cf000000011870410292a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>
Cc: Nickson Yap <hi@nickson.me>, Mention <mention@noreply.github.com>

[@NicksonYap](#) Yes, but even more than that, when you have overlapping information, the same person can be seen from several cameras, and this information needs to be fused somehow. When the views are disjoint, the information for a single detection comes only from one camera. If you define a meaningful correlation function, the fusion is done automatically for both cases.

[Quoted text hidden]

ergysr <notifications@github.com> 5 February 2019 at 05:28
Reply-To: ergysr/DeepCC
<reply+00521a7b7dfccb576b28c33accbeb8a93b987453b0ba405092cf000000011870709792a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>

Cc: Nickson Yap <hi@nickson.me>, Comment <comment@noreply.github.com>

Fusion: There can be two detections of the same person in overlapping views. When inferring the predicted ground plane trajectory, these two detections should be fused as one because they are the image projections of a unique real world trajectory.

<https://github.com/ergysr/DeepCC/blob/ab89f0a82a6b603d87c508b2c3ddc217d7075d6e/src/L1-tracklets/createTracklets.m#L50>
<https://github.com/ergysr/DeepCC/blob/c1cbe0e20aa9ef432ee2322625d861103fb07bb8/src/L2-trajectories/solveInGroups.m#L45>
<https://github.com/ergysr/DeepCC/blob/bf83b90d89f239f3bf4540b0bb6053b8048ce45a/src/L3-identities/solveInGroupsIdentities.m#L57>

—
You are receiving this because you commented.

[Quoted text hidden]

ergysr <notifications@github.com> 6 February 2019 at 09:12
Reply-To: ergysr/DeepCC
<reply+00521a7ba5eced797fb31154ea91124bb9eb17011958831092cf000000011871f66992a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>
Cc: Nickson Yap <hi@nickson.me>, Mention <mention@noreply.github.com>

@NicksonYap Tracking computation is done at 4 levels:
L0 computes detections for each frame
L1 computes tracklets (~1 second long)
L2 computes single-camera trajectories
L3 computes multi-camera trajectories

The ReID model is only used to extract appearance descriptors, and a constant velocity motion model is used to compute some motion correlation. These appearance and motion correlations are combined into the final correlation.

Each computation level groups observations from the previous level. For instance L0 takes images and computes detections. L1 takes the L0 detections, computes some features on them (using the reid appearance model and linear prediction motion feature) and runs BIPCC to find groups of detections that should belong to separate people. L2 does the exact same, but groups L1 tracklets into single camera trajectories. This repeats until L3 takes the L2 single camera trajectories and groups them into multi camera trajectories.

You can read the [ACCV 2014 paper](#) for the basic details. Your interpretation sounds more or less right.

—
You are receiving this because you were mentioned.

[Quoted text hidden]

ergysr <notifications@github.com> 21 February 2019 at 03:44
Reply-To: ergysr/DeepCC
<reply+00521a7bf6d3ff484fa905d8877d35bc39dfd407bca2db7092cf000000011885703792a169ce1837e2f8@reply.github.com>
To: ergysr/DeepCC <DeepCC@noreply.github.com>
Cc: Nickson Yap <hi@nickson.me>, Mention <mention@noreply.github.com>

Closed [#56](#).

[Quoted text hidden]