Tracking Active Regions detected by SPoCA

Cis Verbeeck

January 21, 2010

1 Introduction

- An Active Region is defined as a coherent group of corresponding AR blobs. Let's suppose this grouping of blobs into ARs is already done. The surface of the AR is the surface of the whole group, and is confined within a bounding box.

- The goal of tracking is to appoint the same ID to an AR over time, even if these timestamps of the AR are entered into the HEK as separate events. We will also keep track of the parents and children of AR in case of an AR merge or split.

- We assume the output (HEK reporting) cadence is a multiple of the input (tracking) cadence. While we will update the ARs at every input cadence time stamp t_i, we will only write the result to the HEK at output cadence time stamps.

- The HEK needs a separate AR event for every output cadence time stamp. Let's internally keep a separate AR for every input cadence time stamp. In addition to this, I propose to keep a list of AR, the members of which would correspond to a physical AR: one and the same AR entry from the start to the end of its lifetime. Every AR in that list would have the following members: startTime, endTime, ID, parents (= list of pointers to parent ARs in the same list, default = empty list), children (= list of pointers to child ARs in the same list, default = empty list). Every HEK AR event will have as member a pointer to the corresponding AR in the AR list. This list will be useful for our own SPoCA version, where we want just one AR event for the lifetime of an AR. Also, it makes it a lot easier to report, e.g., a split: the parents and children lists need to be adapted just once, and this information will be easily available to all earlier HEK AR events corresponding to this AR.

- Suggestion for the format of the ID: date.time.AR.SPoCA.v1-n, e.g., 20100119.143000.AR.SPoCA.v1-2 would be the second largest (hence -2) AR detected on January 19, 2010 at 14:30:00 UT by version 1 of SPoCA.

- When an AR observed at time t_i seems to have disappeared at time t_{i+1}, but is present again at some time $t_k = t_{i+2}, \ldots, t_{i+n_{NotDeadYet}}$ (where $n_{NotDeadYet}$ is a small number which we need to choose), we consider it to be the same AR (meaning it has the same ID). Similarly, when an AR observed at time t_i splits into pieces at time t_{i+1}, but merges again at some time $t_k = t_{i+2}, \ldots, t_{i+n_{NotSplitYet}}$, then we consider it to have been one AR from t_i till t_k, to avoid segmentation artifacts. I think it's a bad idea to do the same in case of a merged AR which splits again in a certain period, since we can't pretend to have two different ARs between the merge and the split.

2 Tracking algorithm step (from t_{i-1} to t_i)

- Consider all AR $\mathrm{AR}_{t_{i-1},j}$ at time t_{i-1} and all AR $\mathrm{AR}_{t_i,k}$ at time t_i (both ordered from largest to smallest size). For each combination, verify whether their bounding boxes overlap. If this is the case, verify whether the masks overlap, i.e., verify whether $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$.

- Using this information, perform the following actions for every AR $\mathrm{AR}_{t_{i-1},j}$ at time t_{i-1}:

  - If there exists **no AR** $\mathrm{AR}_{t_i,k}$ at time t_i such that $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$, then it seems that $\mathrm{AR}_{t_{i-1},j}$ has **disappeared** at time t_i. Do nothing.
  - If $\mathrm{AR}_{t_i,k}$ is **the only AR** at time t_i such that $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$, then $\mathrm{AR}_{t_{i-1},j}$ and $\mathrm{AR}_{t_i,k}$ are **the same AR, observed at different times**. Create a new AR event at time t_i and let it point to the same AR in the AR list as $\mathrm{AR}_{t_{i-1},j}$. The position, area, etc. for this new AR event are of course these of $\mathrm{AR}_{t_i,k}$. Put the end time of the AR corresponding to $\mathrm{AR}_{t_{i-1},j}$ equal to t_i.
  - If there are **several AR** $\mathrm{AR}_{t_i,k_1}, \ldots, \mathrm{AR}_{t_i,k_p}$ at time t_i such that $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k_1} \neq \emptyset, \ldots, \mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k_p} \neq \emptyset$, then it seems that $\mathrm{AR}_{t_{i-1},j}$ has **split** into p pieces at time t_i. Create p new AR

events at time t_i, and let each of them point to a corresponding new AR in the AR list (the ID of those ARs is the ID of $\mathrm{AR}_{t_{i-1},j}$, appended with the letters 'a', 'b', 'c', etc., respectively; their start time is t_i). The new ARs have the same parent list as the AR corresponding to $\mathrm{AR}_{t_{i-1},j}$, but with a pointer added to this AR corresponding to $\mathrm{AR}_{t_{i-1},j}$, since it is a new parent for all of these new ARs. All new ARs have empy children list. In the AR corresponding to $\mathrm{AR}_{t_{i-1},j}$, add pointers to all of the new ARs to the children list.

- For all AR $\mathrm{AR}_{t_i,k}$ at time t_i, perform the following actions:
  - If there exists **no AR** $\mathrm{AR}_{t_{i-1},j}$ at time t_{i-1} such that $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$, then it seems that $\mathrm{AR}_{t_i,k}$ is **a new AR** at time t_i. To be sure, we check whether it was already there at some time between $t_{i-n_{NotDeadYet}}$ and t_{i-2}. So for times $t_{i-p} = t_{i-2}, \ldots, t_{i-n_{NotDeadYet}}$, check whether there exists an AR $\mathrm{AR}_{t_{i-p},j}$ at time t_{i-p} such that $\mathrm{AR}_{t_{i-p},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$.
    1. If there exists one, then $\mathrm{AR}_{t_{i-p},j}$ and $\mathrm{AR}_{t_i,k}$ are **the same AR, observed at different times**, so create one new AR event at time t_i and let it point to the same AR in the AR list as $\mathrm{AR}_{t_{i-1},j}$. Put the end time of this AR equal to t_i.
    2. If no such AR exists, then $\mathrm{AR}_{t_i,k}$ is **a new AR**. In this case, create a new AR event at time t_i, and let it point to a new AR in the AR list (ID: see suggested ID format; begin time and end time t_i; empty parents and children lists).
  - If there is **only one AR** $\mathrm{AR}_{t_{i-1},j}$ at time t_{i-1} such that $\mathrm{AR}_{t_{i-1},j} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$, then $\mathrm{AR}_{t_{i-1},j}$ and $\mathrm{AR}_{t_i,k}$ are **the same AR, observed at different times**, but this case was already handled. Do nothing.
  - If there are **several AR** $\mathrm{AR}_{t_{i-1},j_1}, \ldots, \mathrm{AR}_{t_{i-1},j_p}$ at time t_{i-1} such that $\mathrm{AR}_{t_{i-1},j_1} \cap \mathrm{AR}_{t_i,k} \neq \emptyset, \ldots, \mathrm{AR}_{t_{i-1},j_p} \cap \mathrm{AR}_{t_i,k} \neq \emptyset$, then it seems that the p ARs $\mathrm{AR}_{t_{i-1},j_1}, \ldots, \mathrm{AR}_{t_{i-1},j_p}$ have **merged** into $\mathrm{AR}_{t_i,k}$ at time t_i. To be sure, we check whether $\mathrm{AR}_{t_{i-1},j_1}, \ldots, \mathrm{AR}_{t_{i-1},j_p}$ are fragments of what used to be one AR $\mathrm{AR}_{t_l,j}$ at some time t_l in between $t_{i-n_{NotSplitYet}}$ and t_{i-2}. So we check whether $\mathrm{AR}_{t_{i-1},j_1}, \ldots, \mathrm{AR}_{t_{i-1},j_p}$ have one parent in common at some time t_l in between $t_{i-n_{NotSplitYet}}$ and t_{i-2}.
    1. If there exists such a parent $\mathrm{AR}_{t_l,j}$, then we consider the split at time t_{l+1} and the merge at time t_i to be invalid (so it is **one**

and the same AR all of the time). In this case, create a new AR event for every time between t_{l+1} and t_i and let each of them point to the AR corresponding to $\mathrm{AR}_{t_l,j}$. For the new AR event at time t_q between t_{l+1} and t_{i-1}, the mask should be the union of the masks of the different AR at that time which overlap with $\mathrm{AR}_{t_i,k}$. In the children list of the AR corresponding to $\mathrm{AR}_{t_l,j}$, remove the pointers to each of the children between t_{l+1} and t_{i-1}. Finally, remove all the old AR events of the split between t_{l+1} and t_{i-1}, as well as the corresponding ARs in the AR list. Put the end time of the AR corresponding to $\mathrm{AR}_{t_{i-1},j}$ equal to t_i.

2. If no such a parent $\mathrm{AR}_{t_l,j}$ exists, then the **merge is real**. In this case, create a new AR event at time t_i, and let it point to a new AR in the AR list (ID: see suggested ID format; begin time and end time t_i; parents list: add a pointer to the ARs corresponding to $\mathrm{AR}_{t_{i-1},j_1}, \ldots$, $\mathrm{AR}_{t_{i-1},j_p}$; children list: empty). Add a pointer to the new AR in the children list of the ARs corresponding to $\mathrm{AR}_{t_{i-1},j_1}, \ldots$, $\mathrm{AR}_{t_{i-1},j_p}$.

3 Future work

- Determine optimal input cadence, $n_{NotDeadYet}$, $n_{NotSplitYet}$.
- Take into account Bright Points
- If $n_{NotDeadYet}$ and $n_{NotSplitYet}$ are large, we might need to take into account solar rotation.