MEMcaf

MEteorological Metadata CombinAtion Framework

Second mid-term report

DIVA Group, University of Fribourg, Switzerland
and
Federal Office of Meteorology and Climatology MeteoSwiss

Master thesis of Lorenzo Clementi

February 8, 2008

1 Introduction

According to the planning proposed in [1] and [2], this document represents the second milestone of my master thesis. Since the first mid-term presentation (December 14, 2007) we have progressed in the development of the framework prototype and we implemented a real end-to-end use case.
After this brief introduction, Section 2 will discuss the advancement in the framework implementation, whereas Section 3 will present a concrete use case. Section 4 discusses some interesting outlooks and, finally, Section 5 describes the final phase of the project.

2 Status of work

The main advancements implemented since the first mid-term presentation are the following:

Parameters perturbation In order to determine the best value for a given parameter, it is now possible to *perturbate* it. The perturbation process leads to a set of slightly different experiments; the evaluation of the results quality gives an indication on which is the best method configuration.
The parameter perturbation works for integer type parameters: the user can select either a single value (no perturbation) or a range of values (lower bound, upper bound, step width).

Multi threading The two main operations executed by the framework are the metadata combination and the data combination (see steps 3 and 4 of Figure 1 in [2]). These operations – and particularly the second one – may require a considerable amount of time depending on number of CPUs and their speed, as well as on the amount of information to be treated. In order not to block the application, both these operations are executed in newly created, dedicated threads.

Sessions It is often interesting to execute the same experiment by slightly modifying the configuration. For this reason, we implemented the concept of *session* by offering the possibility to serialize the frame that gathers the method configuration. It is therefore possible to save an experiment parameters and, later, edit them and execute the experiment again.

GUI and other minor improvements In parallel to the previous points, we adapted the graphical user interface in order to ease the use of the framework.

3 A real use case

As a proof-of-concept, it showed natural to combine the two data types directly available at the RASA Team[1]: radar and satellite images. The goal of this use case is to apply an *atmospheric motion field* derived from satellite images to a radar detected *precipitation field* in order to evaluate its displacement and, in addition, to determine the set of parameters that yield the best result for our region. The interest of this use case is clearly highlighted in [3]: "*nowcast validation and comparison with extrapolation is incomplete*".
Note that our application does not compute the vector field and the extrapolated images, but it uses some specific software to perform these operations. Our tool focuses on the way the set of experiments is organized and it aims to provide a reusable architecture for other tests.

The combination method described above is composed by the following steps:

1. **Method configuration:** selection of a specific event to investigate (time information) and parameter value assigning (see Figure 2). These parameter values will then be passed to the algorithm that computes the atmospheric motion field.
2. Verification of the availability of data (satellite and radar images).
3. **Metadata combination:** generation of an XML procedure file containing all the sequentially ordered operations needed to consistently combine the data (e.g. image calibration, projection).
4. **Data combination:** execution of the combination on real data and generation of the extrapolated images.
5. **Result evaluation:** comparison of the extrapolated images with the measured data; error evaluation (see below).

In order to evaluate the quality of the extrapolated images we defined a simple **quality flag** that is computed by comparing the extrapolated radar image with the actually measured data. Figure 1 shows the result of a linear combination that subtracts the measured image from the extrapolated one.

Given the notation P_Y = number of yellow pixels, P_R = number of reddish pixels, P_G = number of greenish pixels, the quality index is computed as follows: $\frac{P_Y}{P_Y+P_R+P_G}$. If this value is close to 1 the quality is very good, if it is close to 0 the quality is poor.
A more refined verification method is called **CRA** and it is illustrated in [4]; this method could be implemented as a further development of the use case described in this paragraph.

Based on this quality index, several sets of experiments are currently being analyzed. From the meteorological point of view, the goal is now to point out the interesting trends in the results quality in order to quantify the **range of optimization**.

3.1 First results

The first results show interesting trends arising from the **pattern size** perturbation. This parameter has a big impact on the running time of the AMF algorithm and its optimization is a trade-off between **cost** (running time) and **benefit** (result quality). As shown in Figure 3, the most efficient pattern size for the extrapolation of the OMC radar product seems to be around 60 - 80 pixels.
In the 6 cases under analysis, we also pointed out that the extrapolation quality decreases logarithmically over time.

[1] Radar and Satellites Team at MeteoSwiss Locarno-Monti.

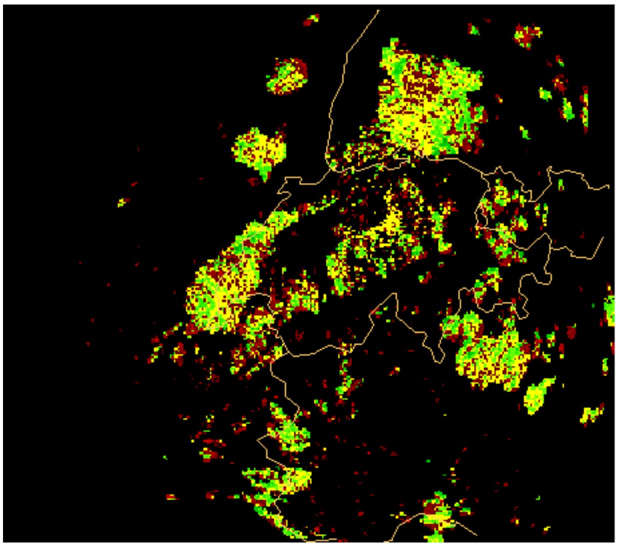

Figure 1: Yellow regions represent correct pixels, reddish regions are under estimated pixels and greenish regions are over estimated pixels.

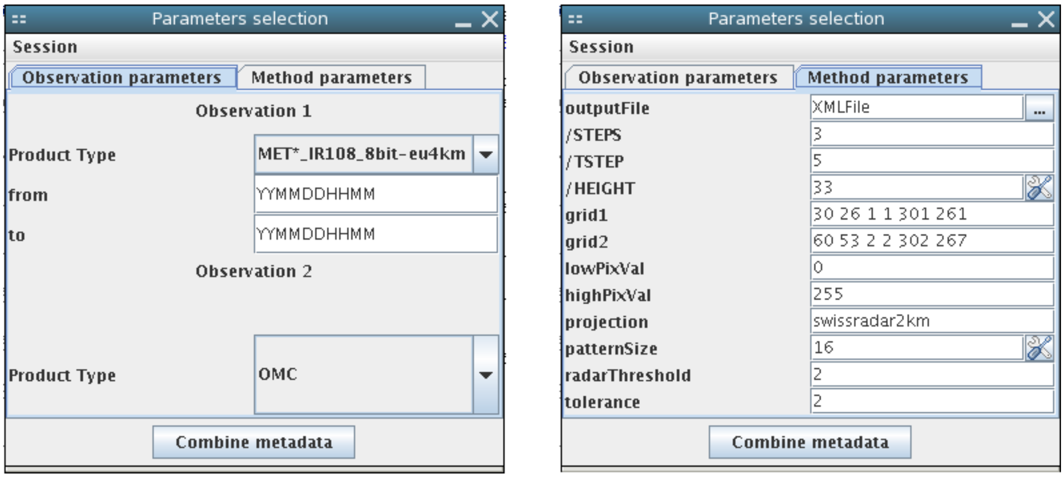

Figure 2: Method configuration.

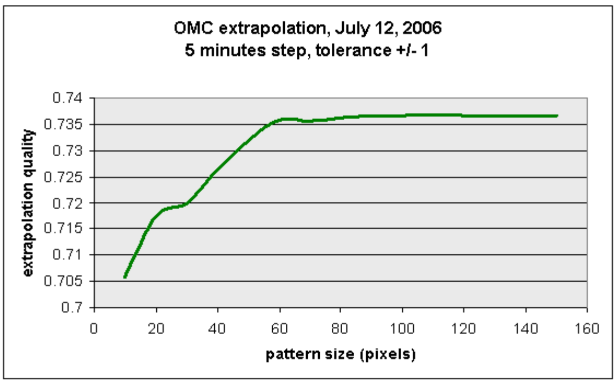

Figure 3: Perturbation of the pattern size parameter.

4 Outlook

In [2] we proposed a list of extensions, some of which could be implemented in the context of this master thesis. In this document, on the other hand, we propose a list of some interesting features whose implementation, however, is outside of the context of this thesis, due to the limited remaining time.

1. Automation of radar and satellite archive queries in order to ease the testing.
2. Analysis of experiments that involve several, correlated parameters.
3. Comparison / combination of TRT[2] extrapolation and AMF extrapolation.
4. Implementation of other combination methods.
5. Machine learning approach: based on the quality flag, implementation of a machine learning technique that automatically determines the best set of parameters.

5 Planned activities

The next planned activities are:

1. From now (beginning of February 2008) until the end of February 2008: use case result analysis, framework consolidation (see Extensions in [2]).
2. From March 2008 until April 4, 2008: thesis writing.
3. April 4, 2008: final deadline (source code and thesis delivery).

References

[1] I. Giunta, G. Galli (2007), *Work proposal for the Master thesis of Lorenzo Clementi*, MeteoSwiss

[2] L. Clementi (December 2007), *Master Thesis first mid-term report*, DIVA Group & RASA Team, MeteoSwiss

[3] J. W. Wilson, *Precipitation nowcasting: past, present and future*, National Center for Atmospheric Research, Boulder Colorado USA

[4] B. Ebert, *CRA (entity-based) verification*, Bureau of Meteorology Research Centre, Melbourne, Australia

[2] Thunderstorm Radar Tracking, an algorithm developed by A. Hering at MeteoSwiss in collaboration with Météo France.