Automating the Detection and Classification of Active Deformation Areas—A Sentinel-Based Toolset †

José A. Navarro [1,*], María Cuevas [1], Roberto Tomás [2], Anna Barra [1] and Michele Crosetto [1]

[1] Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Av. Carl Friedrich Gauss 7, 08860 Castelldefels, Spain; maria.cuevas@cttc.es (M.C.); anna.barra@cttc.es (A.B.); michele.crosetto@cttc.es (M.C.)

[2] Dpto. de Ingeniería Civil. Escuela Politécnica Superior de Alicante. Universidad de Alicante. P.O. Box 99, E-03080 Alicante, Spain; roberto.tomas@ua.es

* Correspondence: jose.navarro@cttc.es

† Presented at the II Congress in Geomatics Engineering, Madrid, Spain, 26–27 June 2019.

Abstract: The H2020 MOMIT project (Multi-scale Observation and Monitoring of railway Infrastructure Threats, http://www.momit-project.eu/) is focused on showing how remote sensing data and techniques may help to monitor railway infrastructures. One of the hazards monitored are the *ground movements* nearby such infrastructures. Two methodologies targeted at the detection of Active Deformation Areas (ADA) and the later classification of these using Persistent Scatterers (PS) derived from Sentinel-1 imagery had been developed prior to the start of MOMIT. Although the validity of these procedures had already been validated, no actual tools automating their execution existed—these were applied manually using Geographic Information Systems (GIS). Such a manual process was slow and error-prone due to human intervention. This work presents two new applications, developed in the context of the MOMIT project, automating the aforementioned methodologies: ADAfinder and ADAclassifier. Their goal was (1) to reduce the possibility of human errors to a minimum and (2) to increase the performance/reduce the time needed to obtain results, thus allowing more room for experimentation.

Keywords: Ground Deformation Analysis; Ground Deformation Classification; Process Automation

1. Introduction

This paper briefly presents the implementation as actual computer tools of two already existing methodologies to (1) detect ADAs ([1]) using PSs obtained by means of Persistent Scatterers Interferometry (PSI) techniques and Sentinel-1 imagery and (2) to classify these as candidates to belong to one or more categories, namely landslides, sinkholes, land subsidences, constructive or consolidation settlements, expansive soils or movements due to thermal effects ([2]). These tools have been developed in the framework and thanks to the funding granted to the aforementioned MOMIT project, but they may be used in any context where the detection and classification of ground movements is required. The results provided by these methodologies have already been checked by their respective authors, so the goal of this paper is (a) to show that the new tools are correct–they do what they have to—and (2) to present the performance boost thus achieved.

2. The Algorithms and Their Implementation

The algorithm—or methodology—on which the ADA detector (ADAfinder) relies is thoroughly described in [1], so the reader interested in how this procedure works should refer to this publication. Nonetheless, and for the sake of completeness, it is briefly sketched here:

- Inputs: (1) Set of PSs in ESRI shapefile format and (2) a few numerical thresholds.
- Procedure:

  1. Filtering. This means removing (1) stable PSs (velocity below some certain threshold) and (2) isolated PSs (distance threshold) or that make groups too small (less than 2 PSs.)
  2. Identification of ADAs, that is, finding the groups of 5 or more PSs that are close enough as to be considered neighbors.
  3. Quality assessment. This means evaluating the likeliness of the groups previously identified being actual ADAs. This is materialized as a value ranging from 1 to 4, where 1 stands for "absolute certainty" and 4 meaning "big uncertainty".

- Output: An ESRI shapefile with polygons defining the contours of the ADAs. The quality index is one of the attributes. Another shapefile contains the PSs; those being part of ADAs are conveniently tagged.

Figure 1 depicts the results of such classification process. The colors green, yellow orange and red correspond respectively to levels 1 to 4 of the quality index described above.

Figure 1. Sample output ADAs and the points defining these. Color coding denotes the quality of the assessment.

ADAclassifier, the application classifying the ADAs, starts where ADAfinder ends, taking its output to find what kind of deformation process(es) is (are) being suffered by the ADAs. A complete description of the algorithms used to accomplish such task may be found in [2]. Again, a brief sketch is presented here for the sake of completeness.

- Inputs: These are: (1) the ADAs output by ADAfinder in ESRI shapefile format and (2) a set of thresholds controlling the classification process; optionally (3) a Digital Terrain Model (DTM), (4) landslides, sinkholes, land subsidences, infrastructures and geologic inventories (ESRI polygons) and (5) a map measuring the horizontal displacements of the PSs. When one or more optional files are absent not all the classification processes will be executed.
- Procedure: there are six different procedures, each one targeted at classifying the ADAs as one of the six different phenomena being tracked. Table 1 briefly describes the one tying to decide if an ADA is a constructive or consolidation process. There are other five algorithms not shown here for the sake of conciseness.
- Output: An ESRI file, which is a copy of the input one including six extra flags stating whether the ADA may correspond or not to any of the six deformation processes investigated.

Table 1. Sketch of the algorithm used to ascertain if an ADA corresponds to a constructive or consolidation process. Source: [2].

Is the whole ADA placed inside the area of a infrastructure?	No	Not a constructive settlement		
	Yes	Time series fit a negative exponential function	Yes	Consolidation process
			No	Not a consolidation process

The applications were developed in C++ for performance reasons. The popular Python language was consciously discarded; in spite of being the language of choice to integrate plug-ins in some GIS

like Quantum GIS (QGIS), its interpreted nature makes this language completely inadequate due to its slowness.

Three different flavors of the software exist. First, the logic of both ADAfinder and ADAclassifier was included in a single library, ready to be used by other, higher level applications willing to include the ability to detect and classify ADAs among their features. Second and third, both ADAfinder and ADAclassifier were delivered in two versions, that is, as command line applications easily integrable in batch procedures as well as full-fledged GUI-based tools, well suited for interactive use and experimentation. These are delivered as executable files.

In spite of avoiding Python, it is still possible to integrate these tools in a GIS, using their GUI executable incarnations. Both ArcGIS and QGIS provide mechanisms to do so; in the case of QGIS a small development in Python is required—which is not necessary in ArcGIS. Figure 2 depicts the integration of ADAfinder (button encircled in red) and ADAclassifier (green circle) in QGIS.
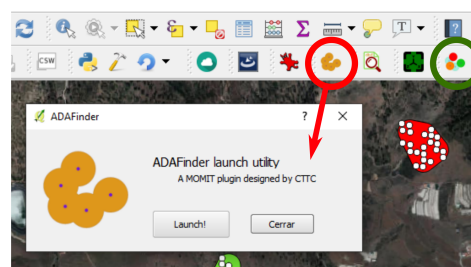

Figure 2. Integrating the tools in Quantum GIS.

3. Results and Conclusions

Since this paper focus on the implementation of two tools automating a couple of validated methodologies that already provided correct outputs *before* such applications existed, the way in which the results justifying the success or failure of ADAfinder and ADAclassifier must be presented is clearly different. In short, it is not necessary to prove that the underlying methodologies do work. This was proven before (see [1,2]). Instead, it is necessary to show that the software is correct (they produce the same results than the methodologies on which they rely) and that they improve the performance and reliability of the manual processes.

Two different methods ([3]) were used to check the correctness of the tools; in the case of ADAfinder, previous datasets processed manually—that is, using a GIS tool—existed so the strategy was to process these usign ADAfinder and compare the results looking for discrepancies. Such comparison was automated as much as possible to avoid the need of always subjective human assessment. This meant exporting the values of the attributes of the ADAs, both for the manual and automated results, and run a difference checker software. No differences higher than the precision of the standard 8-byte double value were detected, so this allowed to validate the correctness of this tool. On the contrary, no former data were available for ADAclassifier. Therefore, synthetic datasets, adhering to easily recognizable geometric patterns were created. Such geometric patters allowed for an absolute certainty about what kind of results had to be obtained—that is, what ADAs had to be tagged as being undergoing this or that deformation process. Figure 3 depicts one of these datasets, prepared specifically to check the settlement/consolidation process described in Table 1. The reader may check by him or herself how the ADAs tagged as settlements in Figure 3d are those matching the conditions stated in the aforementioned table.

The performance results for ADAfinder fulfilled the expectations. The equipment used was a Windows 10 64-bit computer, with an Intel Core i5-5300U processor including 2 cores and 4 threads at 2.3 GHz, 8 Gb of RAM and a 500 Gb magnetic (non-SSD) hard disk. Processing the dataset, including 926916 PSs, took 179 s, which is ten times faster than performing this task manually.
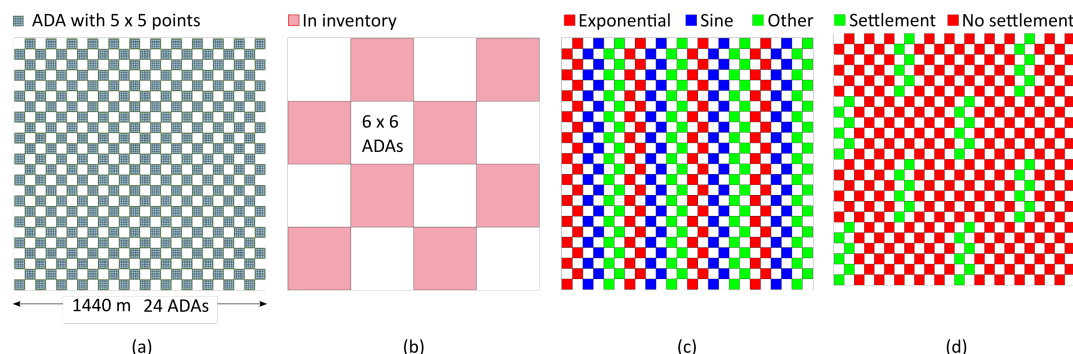
Figure 3. Synthetic dataset to check the consolidation process: (**a**) ADAs distributed in a 24 × 24 checkerboard pattern. (**b**) Infrastructures inventory. (**c**) Kind of deformation time series for each ADA. (**d**) Results of the classification process.

The equipment used to check the performance of ADAclassifier was an Intel Core i7-2670QM at 2.2 GHz, 4 cores, 8 threads, 6 MB cache, 8 GB RAM, 250GB SSD disk and Windows 10, 64 bits. The dataset consisted of a total of 11472 PSs, 411 polygons included in the several inventories and a DTM consisting of 2053 × 1445 elevation values. The computer needed 2 seconds to classify the ADAs checking the six available classification processes. The manual process needs around 5 minutes. This means that ADAclassifier is about 150 times faster—and eliminates chance of human errors.

To conclude, it is possible to state, thanks to the comprehensive and rigorous testing process undergone by these two tools, that ADAfinder and ADAclassifier are two applications that effectively detect and classify ADAs according to state-of-the-art algorithms, reducing significantly the time needed to process data and the chance of introducing human errors in the process, thus incrementing the productivity of the operators. These tools are nowadays very useful in the context of the MOMIT project, but may also be used in any situation where monitoring ground movements is necessary.

References

1. Barra, A.; Solari, L.; Béjar-Pizarro, M.; Montserrat, O.; Bianchini, S.; Herrera, G.; Crosetto, M.; Sarro, R.; González-Alonso, E.; Mateos, R.M.; et al. A methodology to detect and update Active Deformation Areas based on Sentinel-1 SAR images. *Remote Sens.* **2017**, *9*, 1002.
2. Tomás, R.; Cano, M.; Pastor, J.L.; Riquelme, A. Automatic Classification of ADAs. MOMIT Technical Note. 2018. Available online: http://www.momit-project.eu/publications (accessed on 21 February 2019).
3. Navarro, J.A.; Tomás, R.; Cuevas, M.; Crosetto, M. SW Data Processing Components Test Report. MOMIT Deliverable. 2018. Available online: http://www.momit-project.eu/publications (accessed on 21 February 2019).