

# MASSIVE LINKING OF PS-IN SAR DEFORMATIONS TO A NATIONAL AIRBORNE LASER POINT CLOUD

A. L. van Natijs\*, R.C. Lindenbergh, R.F. Hanssen

Delft University of Technology, Department of Geoscience and Remote Sensing.  
Stevinweg 1, 2628CN Delft, The Netherlands - A.L.vanNatijs@student.tudelft.nl

Commission II, WG II/10

**KEY WORDS:** Airborne Laser Scanning, PS-InSAR, geolocation, deformation monitoring, local surface reconstruction, data fusion

## ABSTRACT:

Build on soft soil, close to sea level the Netherlands is at high risk for the effects of subsidence and deformation. Interferometric Synthetic Aperture Radar (InSAR) is successfully used to monitor the deformation trends at millimetre level. Unfortunately the InSAR deformation trends suffer from poor geolocation estimates, limiting the ability to link deformation behaviour to objects, such as buildings, streets or bridges. A nationwide, high resolution, airborne LiDAR point cloud is available in the Netherlands. Although the position accuracy of this LiDAR point cloud is too low for deformation estimates, linking the InSAR location to the geometries outlined by the LiDAR point can improve the geolocation estimates of the InSAR trends. To our knowledge no such integration is available as of yet. In this article we outline methods to link deformation estimates to the LiDAR point cloud and give an outlook of possible improvements. As a test we link 3.1 million TerraSAR-X InSAR Persistent Scatterers to 3 billion LiDAR points, covering the city of Delft and surroundings.

## 1. INTRODUCTION

### 1.1 Subsidence

Subsidence in the Netherlands, and deformation in general, is threatening building integrity, damaging infrastructure and lowering the land with respect to sea-level. Deformation occurs at all scales, from single pillar failure at the 't Loon shopping mall in 2011 (Chang and Hanssen, 2014) to complete regions suffering from effects such as subsidence relative to the water table (Boersma, 2015).

Deformation processes include: peat compaction in the west and north of the country (Boersma, 2015); induced subsidence (and seismic activity) due to gas extraction in the Groningen area (Ketelaar et al., 2006); land uplift and cavity formation due to the flooding of old mine works in Limburg (Bekendam and Pottgens, 1995).

### 1.2 PS-InSAR deformation monitoring

Interferometric Synthetic Aperture Radar (InSAR) is used to monitor deformation from satellites. Millimetre per year accuracy can be achieved in deformation trend estimation. Unfortunately the source of the deformation signal is, in general, less accurately known: geolocation estimates of PS-InSAR are known with metres precision at best, depending on the sensor (Dheenathayalan et al., 2016).

Although this will allow for deformation estimates up to street level (Ketelaar et al., 2006), the deformation signal can not be attributed to a single geometric feature. Persistent scatterer InSAR (PS-InSAR) measurements are commonly dominated by a single scatterer. The location of this scatterer is of key importance in the understanding and interpretation of the deformation behaviour: a subsiding garden house or street will require different precautions than a subsiding bridge pillar.

### 1.3 Combination with LiDAR

To find and improve the estimated location of the dominant scatterer it is beneficial to combine radar measurements with a (high resolution) point cloud. This will allow for linking scattering behaviour to a geometric feature in the scene.

An example of a traditional (2D) InSAR deformation map can be seen in figure 2. A 3D visualisation aids this interpretation of the PS-InSAR signal, over the classical 2D mapped interpretation. Geometric 3D linking of PS-InSAR geolocation estimates to 3D LiDAR point cloud data will give a quantifiable improvement of the geolocation.

Geometric (3D) linking of the datasets will provide:

- Assessment of differential deformation, as a deformation signal can be *attached* to a building geometry.
- Linking of the deformation signal to specific parts of the infrastructure, for maintenance planning and early warning.
- Detection and mitigation of (regional) errors and trends in the radar processing.

This article explains how to create this missing link by truly integrating both data sources. Given the (free) availability of a nationwide airborne LiDAR dataset (Actueel Hoogtebestand Nederland), and the available TerraSAR-X InSAR data, the Netherlands form a perfect test bed for this integration of datasets. Furthermore existing online point cloud viewers, such as Potree (Schütz, 2018), can be extended to visualise this link between the laser point cloud and radar data.

Currently no such combination of datasets is known to us. The combination of optical images and SAR is more common and

\*Corresponding author

aimed at the texturing, classification or 3D reconstruction of SAR point clouds (Tupin, 2010; Schmitt et al., 2017). Although successful, (Wang et al., 2017) suffered from poor InSAR geolocation accuracy.

## 2. DATA

### 2.1 TerraSAR-X

The German (DLR) TerraSAR-X Synthetic Aperture Radar (SAR) mission, was launched in 2007 and delivers high resolution radar imagery ever since. As a Public-Private Partnership the mission combines scientific and commercial interests of X-band, land oriented monitoring applications. With an 11-day repeat cycle and resolutions up to  $3 \times 3$  m the mission can provide deformation data of high spatial and temporal resolution (Werninghaus and Buckreuss, 2010).

For this article radar data from two TerraSAR-X orbits is available, descending and ascending orbits over the same region, covering Delft, surrounding fields, Rijswijk and parts of The Hague (the Netherlands), marked in red on figure 3 and shown in more detail in figure 1.

A total of 72 radar images were acquired between 2016-01-06 and 2018-01-01, 36 for each orbit. SAR Interferometry (InSAR) was applied to extract deformation signals by analysing the time series of phase changes. Pixels that can be tracked consistently over multiple acquisitions are Persistent Scatterers (PS). These *coherent* pixels denote the deformation behaviour of the same scatterer over longer periods of time. (Hanssen, 2001) A linear deformation trend (in time,  $\frac{\text{mm}}{\text{yr}}$ ) is estimated for those points.

Data of the descending orbit contains 1.7 million PS-InSAR points, the ascending data contains 1.4 million points. For all points a geolocation estimation is provided in WGS84 coordinates and a height above NAP (*Normaal Amsterdams Peil*). Both datasets span the same area of 123 km<sup>2</sup>, of which 60 km<sup>2</sup> is over urban (built) terrain, where the highest density of persistent scatterers is to be expected (Hanssen, 2001, CBS, 2017).

**2.1.1 Error model** The ratio of the error ellipsoid in range (direction of radar signal), azimuth (direction of flight) and cross-range (perpendicular to signal and flight direction) was estimated to be 1/2/22 for typical consistent scatterers (figure 4, not to scale). With an estimated standard deviation of 0.128 m, the axes of the error ellipsoids are 0.128 m (range,  $\sigma_r$ ), 0.256 m (azimuth,  $\sigma_a$ ), 2.816 m (cross-range,  $\sigma_{cr}$ ) (Dheenathayalan et al., 2016).

The direction of flight was defined to be 192° for the descending and 350° for the ascending orbit, with a elevation angle of 65.9° for both orbits.

### 2.2 Actueel Hoogtebestand Nederland

*Actueel Hoogtebestand Nederland* (AHN) is a nationwide LiDAR elevation model. AHN was first recorded in 1996 and is licensed as open data since march 2014. The raw point cloud data is published via *Publieke Dienstverlening Op de Kaart* (PDOK) (Kadaster and Geonovum, 2018).

AHN was acquired from an airborne platform, from which laser pulses were fired at the ground below. Given the known propagation velocity of light in air, the time interval between transmission and receiving the reflected signal (*echo*) is proportional to

the distance from the aircraft to the ground. Multiple returns are possible, for example in vegetated areas, where parts of the pulse reflect on different surfaces in the scene. The position and orientation of the aircraft are recorded simultaneously using GNSS and inertial motion sensors to record the position from where the measurement was acquired and in which direction (Vosselman and Maas, 2010).

New iterations are acquired approximately once every ten years, as can be seen in table 1. Acquisition of both AHN1 and AHN2 is finished: AHN2 supersedes AHN1. AHN3 is yet only partially available, and is used where available (figure 3). A combination is made between AHN2 and AHN3 to create a single dataset with the highest possible point density. The higher point density allows for better detection of small objects and improved reconstruction of facades that are badly aligned with respect to the viewing angle of the laser scanner.

A summary of the data volume involved is given in table 2. The file size increase of AHN3 is due to the addition of extra attributes, such as classification, intensity and acquisition time.

	Recording	Density ( $\frac{\text{pt}}{\text{m}^2}$ )
AHN1	1996 - 2003	0.06 - 1
AHN2	2008 - 2012	16.8
AHN3	2014 - 2018	18.2

Table 1. Acquisition years and average data density of the three available iterations of AHN. Point densities for AHN2 and AHN3 were determined over the study area (9 tiles).

	Area (km <sup>2</sup> )	Points	Size
AHN2	35 997	640 billion	988 GiB
AHN3	16 249	252 billion	1.1 TiB
Combined			1.7 TiB

Table 2. Coverage, point count and dataset size of AHN2, AHN3 and a combination of both.

AHN is defined in *Rijksdriehoekskoördinaten* with height relative to NAP, 'RDNAP', the Dutch national coordinate system (EPSG:7415). This Cartesian coordinate system is used as the basis for this project.

To ease navigation in the web application (section 3.1) the point cloud is coloured based on the publicly available aerial photograph of 2016 (Kadaster and Geonovum, 2018). This photograph shows small differences to the point cloud, as it was not recorded simultaneously.

**2.2.1 Error model** For AHN (iteration 2 and 3) the accuracy is defined as maximum 5 cm systematic ( $1\sigma$ ) and a 5 cm stochastic error ( $1\sigma$ ) in the vertical direction. Requirements for horizontal accuracy are 50 cm ( $1\sigma$ , both  $x$  and  $y$ ) for objects larger than  $2 \times 2$  m. In reality this is often outperformed (van Meijeren, 2017).

The large standard deviation and the infrequent acquisition make AHN itself unsuitable for deformation monitoring at the millimetre level that is obtained by InSAR monitoring (van Meijeren, 2017).

### 2.3 Data preparation

For the massive visualisation a combined dataset of AHN3 and AHN2 is created, AHN3 is used whenever available (figure 3).

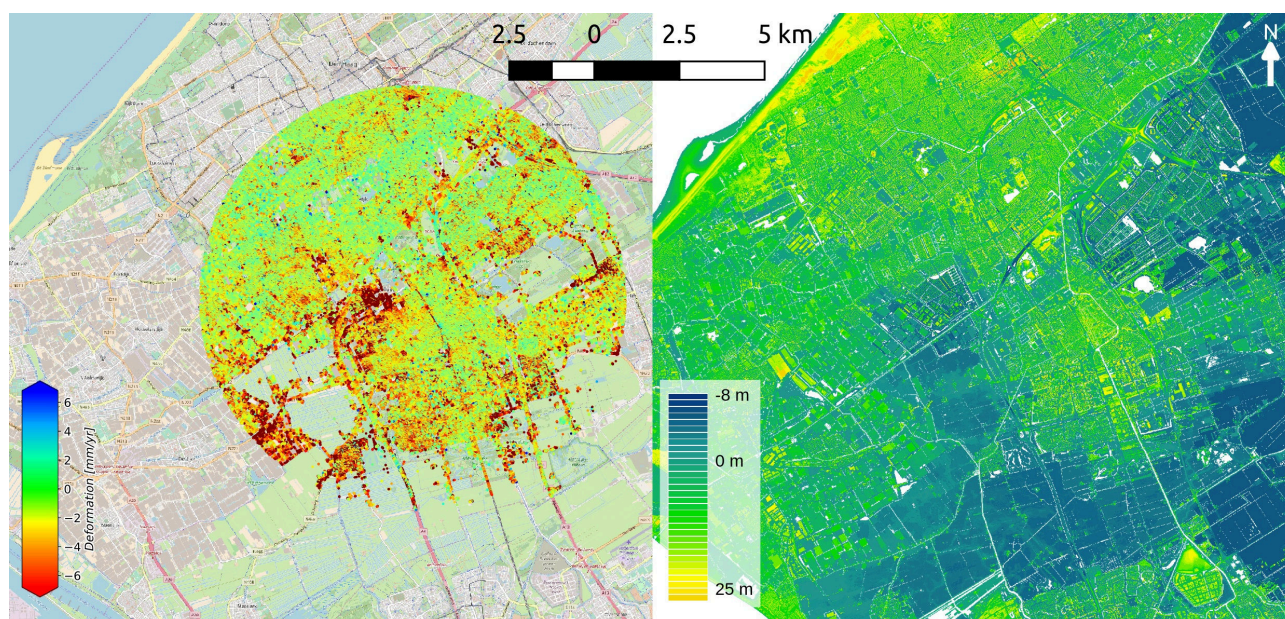


Figure 1. Left: PS-InSAR data from both ascending and descending orbits. The city of Delft is hidden below the points. The Hague can be seen in the north and the northern part of Rotterdam in the south. (Background: OpenStreetMap) Right: digital surface model extracted from AHN3 of the same region.

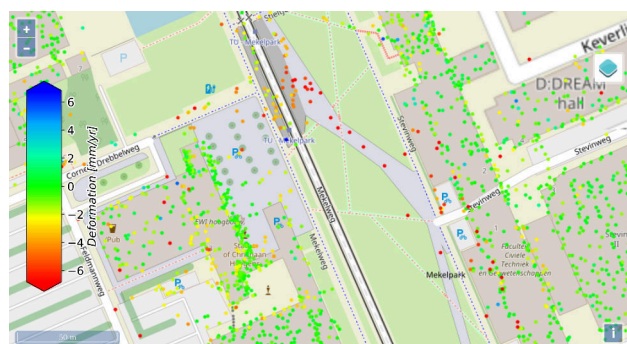


Figure 2. Traditional PS-InSAR deformation map, showing the TU Delft campus. Misalignment can be seen at the building facades. (Background: OpenStreetMap.)

AHN data is delivered in LAZ-tiles of  $5 \times 6.25$  km, based on the standard national tiling scheme. Both datasets are tiled in tiles of approximately  $1 \text{ km}^2$  ( $1 \times 1.25 \text{ km}$ ,  $\frac{1}{25}$  of the original tiles). For each LiDAR tile a buffer of 25 metres on all sides is included, this is to allow radar measurements on the border of two tiles to match. Given the radar error model, such a buffer is large enough to accommodate points on the border between tiles.

This process is done using `pdal` (colouring and clipping) and `las2las` of LASTools (tiling) (Bell et al., 2018, Isenburg, 2018). It created 30137 LAZ-files (tiles), with an average filesize of 58 MiB and 21 million points (average point density of 15 points per square metre). These tiles are small enough to be processed in memory, and large enough for regular file storage. Queries on the LAZ-files are elementary (axes aligned bounding boxes) and do not require a point cloud database (van Oosterom et al., 2015).

Due to the 25 metre overlap between tiles, the method may be run for each tile independently, *horizontal scaling* of the algorithm is possible. That is, each tile can be processed independently, on

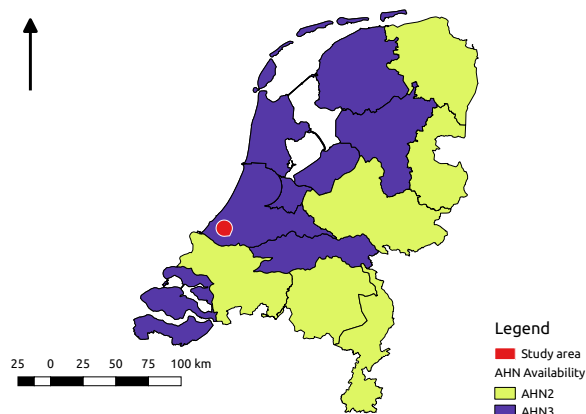


Figure 3. Map of the Netherlands, showing the availability of AHN2 and AHN3 respectively. Shown in red is the extent of the TerraSAR-X data available to this study.

a separate CPU or even separate node. This enables us to combine the PS-InSAR points with *massive* numbers of LiDAR measurements in a distributed manner, reducing the execution time required.

Due to the small size, the radar dataset is not tiled. Small datasets (such as the TerraSAR-X dataset discussed here) are tiled in memory. Larger datasets are converted to HDF5 first, which allows efficient querying during processing.

### 3. METHOD

To aid the interpretation of the deformation signal contained in the PS-InSAR data we want to visualise the available data and link the data to the geometry known from the LiDAR point cloud. The following five steps to achieve this will be discussed here:



### 1. Common visualisation of InSAR and LiDAR data

Interpretation is left to the operator, just as with traditional (online) maps (such as figure 2). Unlike traditional 2D maps the geolocation estimation is shown with the error ellipsoid in 3D. This enables improved interpretation of the radar signal and scene geometry.

### 2. First Nearest Neighbour linking

The Nearest Neighbour in the LiDAR point (single point), with respect to the radar geolocation estimate, is expected to govern the scattering behaviour. This method is computationally efficient, but may overestimate the distance on low density surfaces, as illustrated in figure 4.

### 3. Linking to a single surface

After Nearest Neighbour search, either up to a predefined number of points or all points up to a maximum radius, a single plane is fitted on the points found. This plane approximation of e.g. a facade makes the matching algorithm more robust in areas of low point density.

### 4. Linking to multiple surfaces

For complex geometries the previous approach can be extended. Multiple locally linear ('flat') surfaces may exist in the neighbourhood of the scatterer.

### 5. Linking to dihedral or trihedral geometries

Dihedral and trihedral surface configurations are known to act as better radar reflectors. These geometric configurations might be extracted from the scene. This is currently not implemented.

The difference between the methods is sketched in figure 4. Approximating locally linear surfaces as 2D planes adds robustness in case of point density differences. On rough surfaces, or non flat surfaces this approximation may not hold. Rough surfaces are approximated by an 'average' surface and it might be possible to approximate non-flat surfaces as linearly close to the point of intersection.

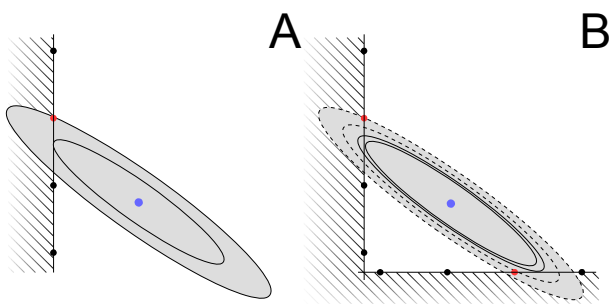


Figure 4. Sketch of the effect of surface reconstruction on matching between a LiDAR point cloud (black/red dots) and a PS-InSAR geolocation estimation (blue dot, including error ellipsoids). The match with Nearest Neighbour in the point cloud, shown in red, is much further away than the actual surface (thin black line). Shown in a *simple* situation A; and a *complex* situation B. Although the horizontal surface is further away, it could be considered a candidate match.

After matching, the resulting matching distances may be used to analyse the geolocation quality, for example detection of biases and trends in the geolocation.

### 3.1 Common visualisation of InSAR and LiDAR data

Web based visualisation is built around Potree (Schütz, 2016, Schütz, 2018) and Three.js. Potree is a WebGL based renderer for large point clouds in the web browser, built on top of Three.js 3D library. Previously the full (nationwide) AHN2 dataset has been successfully converted to be used in the Potree viewer (Martinez-Rubi et al., 2015). Other visual aids (such as the error ellipsoids and plane estimates) can be implemented using Three.js.

Potree ensures a smooth viewing experience by loading the point cloud from a pre-processed octree structure. Only the points in view at the client are downloaded and never more than a user-defined maximum. Due to the  $2\frac{1}{2}$ D nature of the radar dataset this data is distributed using a quadtree tiling scheme, loading only the tiles in view and removing those no longer in view from memory.

### 3.2 First Nearest Neighbour linking

Nearest Neighbour search should take this covariance into account. Use of the *Whitening Transform* will allow any (conventional) Nearest Neighbour algorithm to be used on this problem (Stansbury, 2013).

The viewing geometry of the radar satellite can be expressed as a rotation matrix relative to the world coordinates (RDNAP). This rotation ( $\mathbf{R}_{SAR}$ ) can be combined with the quality model of the radar geolocation ( $\sigma_r, \sigma_a, \sigma_{cr}$ ) to a covariance matrix relative to world coordinates:

$$\mathbf{Q}_{SAR} = \mathbf{R}_{SAR} \begin{bmatrix} \sigma_r & & \\ & \sigma_a & \\ & & \sigma_{cr} \end{bmatrix} \mathbf{R}_{SAR}^T \quad (1)$$

Using the Whitening Transform all points (LiDAR and radar) are projected on the eigenvectors of the covariance matrix of the radar point ( $\mathbf{Q}_{SAR}$ ) and scaled by the eigenvalues. This creates a new coordinate system where the Euclidean metric represents distances in  $\sigma$  rather than metres. All errors are now standard normal distributed, as can be seen in figure 5.

The transformation found works for a single, constant, error model only. As a consequence the transformation has to be calculated and applied for each unique viewing geometry (ie. orbit). This includes construction of a new search structure for each viewing geometry and/or error model. The Multiple Spatial Transformation Technique by (Sakurai et al., 2001), based on pre-processed search structures and approximate transformations may be used to speed up this process if required.

This transformation, based on the eigenvalues ( $\mathbf{E}$ ) and a diagonal matrix of the eigenvectors ( $\mathbf{D}$ ) of the covariance matrix of the radar measurements ( $\mathbf{Q}_{SAR}$ ), can be found using:

$$\mathbf{W} = \mathbf{E}^{-1} \mathbf{D}^{-\frac{1}{2}} \mathbf{E}^T \quad (2)$$

The correctness of this transformation can be checked by transforming the covariance matrix using the Whitening Transform to the identity matrix:

$$\mathbf{W} \mathbf{Q}_{SAR} \mathbf{W}^T = \mathbf{I}_3 \quad (3)$$

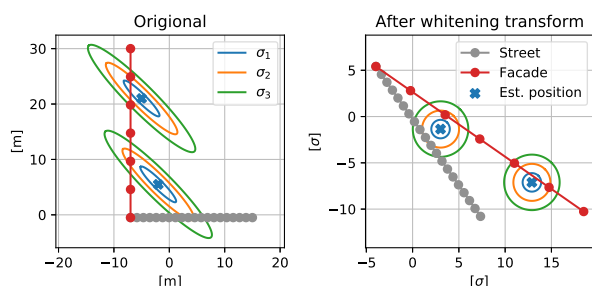


Figure 5. Effect of the Whitening Transform.

In this coordinate system, Nearest Neighbours are nearest in a statistical sense. This search is optimal with respect to the radar, considering AHN the ground truth, without any statistical variability. This assumption is justified due to the (relatively) small error of the LiDAR point cloud.

This search is implemented using `pykdtree`, a kD-tree implementation in Python (Python Development Team, 2017, Nielsen et al., 2017).

### 3.3 Linking to a single surface

The surface is locally approximated as a single, three dimensional, plane of the equation:

$$a\hat{x} + b\hat{y} + c\hat{z} = d \quad (4)$$

Coefficients ( $a$ ,  $b$ ,  $c$ ) equal a normal vector of the plane and  $d$  is a constant.

Two approaches were chosen to approximate the surface:

- Using three LiDAR points, the coefficients are defined by the cross-product of the coordinates of these points.
- Using all Nearest Neighbours found, employing Principal Component Analysis (PCA) to find the coefficients of the plane.

The first method (cross-product) is computationally light but the plane is based on three points only and does not exploit the redundancy in the LiDAR point cloud and does not provide a quality metric for the fit. PCA requires more computational effort but exploit the redundancy in the data and provide a quality metric for the fit.

The error model of both datasets is taken as the starting point of the fusion. To estimate the plane, first the covariance matrix of the coordinates is calculated for the LiDAR points found. This covariance matrix is then transformed using the Whitening Transform based on the covariance matrix of AHN ( $\mathbf{Q}_{AHN}$ , formula 2). The eigenvector corresponding to the smallest eigenvalue of the covariance matrix represents the normal vector. The ratio between the eigenvalues is an indicator for the quality of the fit, for flat surfaces the smallest eigenvalue is much smaller than the other two. The constant  $d$  of the plane equation (4) is found by solving the equation for the mean point.

To determine the point of intersection, the PS-InSAR Whitening Transform is applied to the PS-InSAR point and the surface normal found. The PS-InSAR point is projected orthogonal on the

surface, the distance found is the distance in  $\sigma$ . After the application of the inverse Whitening Transform the point of intersection in world coordinates is found.

### 3.4 Linking to multiple surfaces; dihedral and trihedral geometries

Detection of multiple surfaces can generate a *top-3* list of candidate intersections, such as the horizontal surface slightly further away in figure 4B. Detection of surfaces is based on RANSAC (*random sample consensus*) and is only applied if the surface estimated using the single surface estimating technique indicates the area as non-flat.

Furthermore it will allow the detection of more advanced structures, such as dihedral and trihedral configurations of multiple surfaces. To respect the radar scattering behaviour of dihedral and trihedral structures the source of the deformation signal should be placed at the intersection of the surfaces rather than at the surfaces themselves (Richards et al., 2010).

## 4. RESULTS

### 4.1 Webviewer

A screenshot is shown in figure 6. A live demonstration can be found at <http://dev.fwrite.org/radar/>.

### 4.2 Matching

A comparison between the various matching techniques can be seen in the histograms of figure 7. Local reconstruction of the geometry, by surface approximation, leads to lower distances between the original geolocation estimation and the surface found (orange), as sketched in figure 4. The intersection with the surface is on average  $\frac{1}{2}\sigma$  closer than the first Nearest Neighbour (blue).

Over the whole dataset biases are in the order of decimetres, with standard deviations of multiple metres, as can be seen in figure 8. When expressed in radar coordinates the uncertainty in matching corresponds to the expected geolocation error. The expected geolocation standard deviation was 0.128/0.256/2.816 metre in range/azimuth/cross-range (section 2.1.1). As can be seen in figure 9, the standard deviations are of the same order of magnitude as the original estimations. In range and azimuth it is overestimated, while the cross-range estimate is of the same order as the original estimate. This is all under the assumption that the first Nearest Neighbour is the origin of the backscattered signal.

The improvement in location can be seen in figure 10. Compared to figure 2 stable points are now attributed to the facade while subsiding points remain on the street. This subdivision is to be expected given the stable foundations of the building but had to be made by a skilled operator on traditional maps.

Of the total of 3.1 million PS-InSAR points, less than 20% of the points did not match a Nearest Neighbour within  $2\frac{1}{2}\sigma$ . For surfaces the results are slightly better: 85% of the points was linked to a nearby surface within  $2\frac{1}{2}\sigma$ . Results further away are very unlikely, given the validity of the error model. Missing links are generally due to occlusions in the point cloud, for example on facades and in narrow streets, resulting from the different viewing geometries between the sensors. Some of them are due to faulty interpretation of the geometry, leading to plane estimates that do not provide a realistic point of intersection.

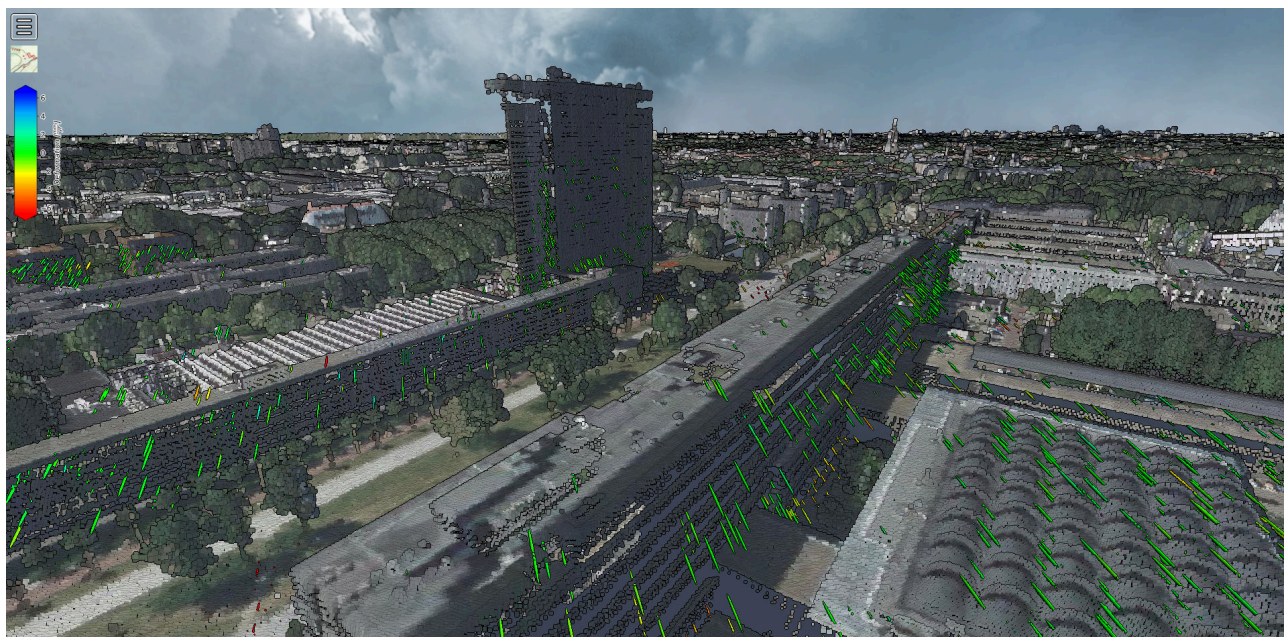


Figure 6. Delft University of Technology campus, shown using an adapted version of Potree, as a coloured point cloud (AHN3) overlaid with TerraSAR-X PS-InSAR trend estimates (shown as  $1\sigma$  error ellipsoids around the expected position, coloured by their deformation signal).

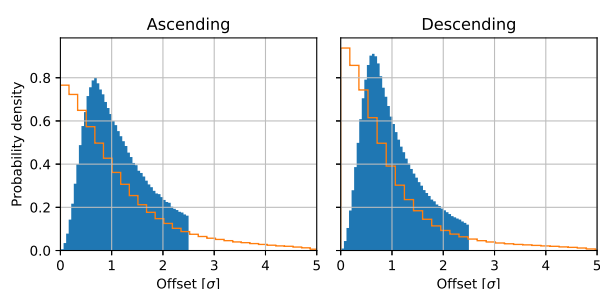


Figure 7. Distance to the closest object using various matching techniques. Blue: first Nearest Neighbour, orange: plane estimation. Nearest Neighbour search is stopped at  $2\frac{1}{2}\sigma$ .

**4.2.1 Quality estimation** Various quality estimates result from the matching procedure. For each point the distance from the original estimation is known, both measured in standard deviation ( $\sigma$ ) and in world coordinates (metres). Furthermore for the surface estimation quality metrics are available, such as the flatness of surface (quality of the approximation). Indicating the quality of the fit for each individual surface.

### 4.3 Regional trends

Although the average offset is almost zero (figure 8), subsets of the data may be biased. By comparing the difference between the point of intersection or Nearest Neighbour and the original geolocation estimate with respect to location, trends become visible.

In figure 11 the median offsets between the geolocation estimation and Nearest Neighbour are shown (NN - estimation), binned in  $100 \times 100$  metre bins. A clear north-south trend can be distinguished in the descending orbit, while in the geolocation of the ascending track the offset increases in the southern part. Due to

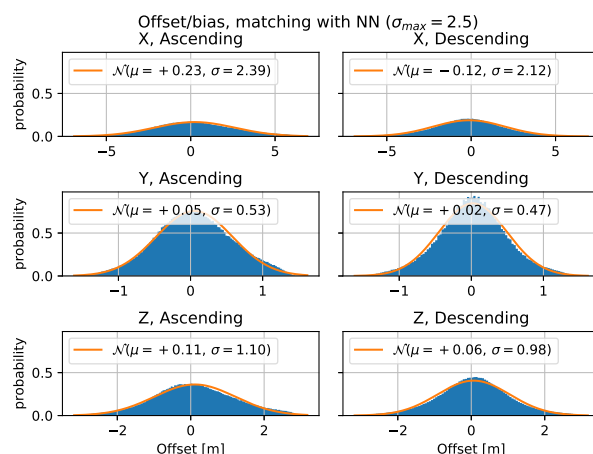


Figure 8. Offsets (biases) in matching, based on the first Nearest Neighbour.

this trend geolocation estimates are off by 2 to 3 metres at the borders of the radar image.

These offsets and trends, that can be converted to radar coordinates too, may help to improve the radar processing.

### 4.4 Processing

Matching 1.4 million SAR points to 3 billion LiDAR points (122 tiles) using the Nearest Neighbour approach takes 15 minutes with three threads on a quad-core Intel i7-3630QM 2.4 GHz laptop computer with 24 GB of RAM. This includes opening the compressed LAZ-files and building of the transformed kD-tree. As the current script is written in python 3.5, higher performance is expected to be achieved by using more optimised programming languages.



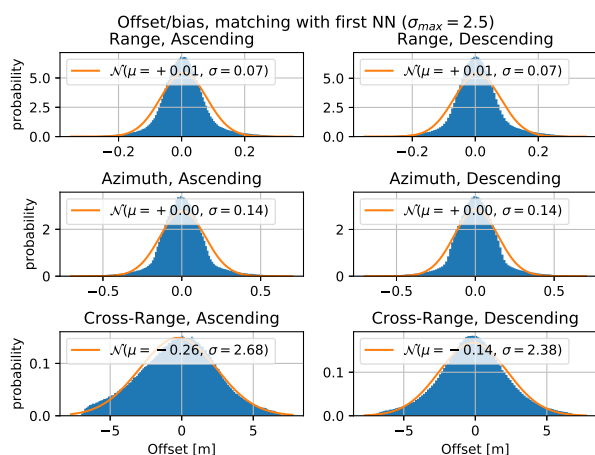


Figure 9. Offsets (biases) in matching, based on the first Nearest Neighbour, in radar coordinates.

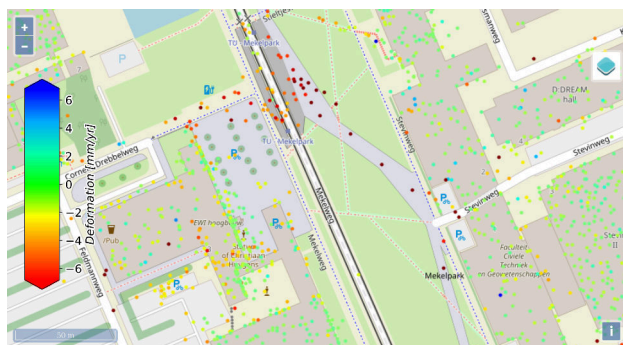


Figure 10. Point locations (mapped in 2D), after matching between the datasets. Compare to figure 2. (Background: OpenStreetMap)

With the research area spanning 122 tiles of a total of 30137 tiles, processing the nationwide dataset will take around sixty hours on this laptop. Given that the program can be scaled over many nodes the computation time can be reduced to the duration desired by adding more computing power.

## 5. DISCUSSION

In this article the geolocation error model is taken as constant over the whole region of the of the radar image and for all different scatterer types. This assumption is likely incorrect, as indicated by (Dheenathayalan et al., 2016). The incidence angle ranges from 20 to 45 degrees over the image and reflectors vary in type: from (round) lamppost to trihedral reflector. Each with different properties and likely different geolocation error estimations. When parameterised, regional variations in the error model can be included.

To improve the matching results the algorithm could be re-run with the biases obtained from the first run after initial matching. After correcting for the bias or trend in the initial geolocation the results of the second geolocation matching algorithm may find a different point of intersection. Quality assessment for this iterative process will be difficult, as the new geolocation will be the result of multiple transformations of the original dataset.

Given the classification provided with the LiDAR point cloud and the free availability of topographic maps in the Netherlands, it is possible to link deformation behaviour to outlines of buildings. Allowing the detection of differential deformation and the conversion from coordinate to addresses (geocoding).

Currently no single error model is implemented. The quality of the surface fit and the quality of the intersection are defined and estimated independently as two separate metrics. An integrated metric could provide, for example, a new quality metric for the point of intersection found.

## 6. CONCLUSION

The techniques introduced enable the efficient attribution of the InSAR deformation signal to real-world objects and features, allowing the next spatial join at a higher scale: linking individual signals to objects. The 3D visualisation will allow for better communication with the greater public, as less interpretation is required with respect to the traditional deformation maps.

Various options exist for the geometric linking of the two datasets. Implemented methods are based on the geometry and may not represent the underlying (physical) radar processes. Nevertheless they improve the geolocation estimate and aid in the attribution of the radar signal to real world objects.

## ACKNOWLEDGEMENTS

We thank SkyGeo (<https://skygeo.com/>) for providing the processed TerraSAR-X data.

## REFERENCES

- Bekendam, R. and Pottgens, J., 1995. Ground movements over the coal mines of southern Limburg, the Netherlands, and their relation to rising mine waters. *IAHS Publications-Series of Proceedings and Reports-Intern Assoc Hydrological Sciences* 234, pp. 3–12.
- Bell, A., Chambers, B., Butler, H., Gerlek, M. et al., 2018. PDAL - point data abstraction library. Hobu inc. <https://www.pdal.io> (March 2018).
- Boersma, C., 2015. The harder we pump, the faster we fall. *Delta Life*, pp. 10–13. Deltares periodical.
- CBS, 2017. Bestand bodembebruik. <https://www.cbs.nl/nl-nl/dossier/nederland-regionaal/geografische%20data/natuur%20en%20milieu/bestand-bodemgebruik-2012> (March 2017).
- Chang, L. and Hanssen, R. F., 2014. Detection of cavity migration and sinkhole risk using radar interferometric time series. *Remote Sensing of Environment* 147, pp. 56–64.
- Dheenathayalan, P., Small, D., Schubert, A. and Hanssen, R. F., 2016. High-precision positioning of radar scatterers. *Journal of Geodesy* 90(5), pp. 403–422.
- Hanssen, R. F., 2001. *Radar interferometry: data interpretation and error analysis*. Vol. 2, Springer Science & Business Media.
- Isenburg, M., 2018. Lastools. rapidlasso GmbH <https://rapidlasso.com/LAStools/> (March 2018).
- Kadaster and Geonovum, 2018. Publieke Dienstverlening Op de Kaart (PDOK). <https://www.pdok.nl/> (March 2018).

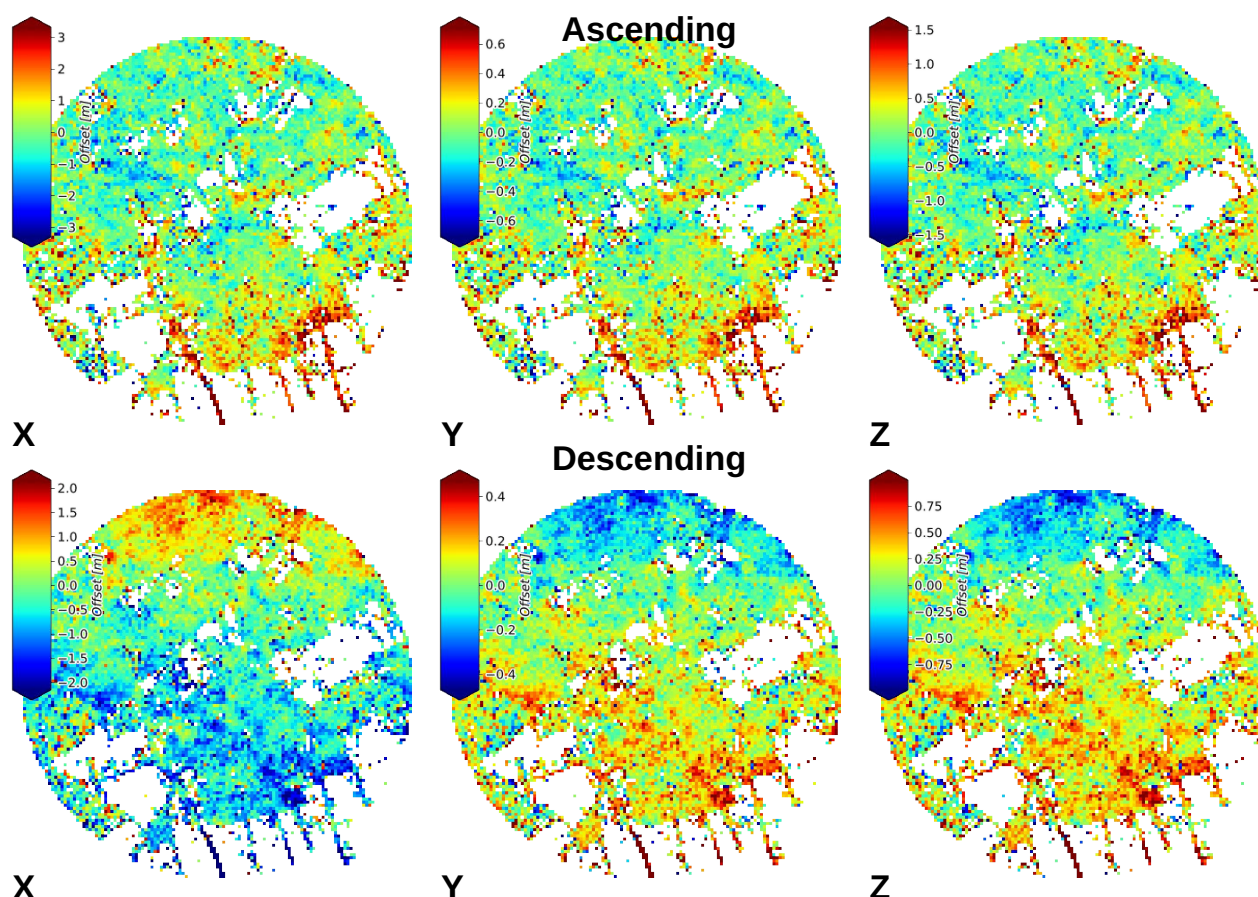


Figure 11. Regional trends (100 × 100 m median) in the offset between the geolocation estimation and the Nearest Neighbour (NN - est.) for both ascending and descending orbits.

Ketelaar, G., van Leijen, F., Marinkovic, P. and Hanssen, R., 2006. On the use of point target characteristics in the estimation of low subsidence rates due to gas extraction in Groningen, the Netherlands. In: *Fringe 2005 Workshop*, Vol. 610.

Martinez-Rubi, O., Verhoeven, S., Van Meersbergen, M., Schütz, M., Van Oosterom, P., Gonçalves, R. and Tijssen, T., 2015. Taming the beast: Free and open-source massive point cloud web visualization. In: *Capturing Reality Forum 2015, 23-25 November 2015, Salzburg, Austria*, The Survey Association.

Nielsen, E., Herb, J. et al., 2017. *pykdtree*: fast kd-tree implementation in Python. <https://github.com/storpipfugl/pykdtree> (March 2018).

Python Development Team, 2017. Python programming language. Python Software Foundation <https://www.python.org> (September 2017).

Richards, M. A., Scheer, J., Holm, W. A., Melvin, W. L. et al., 2010. *Principles of modern radar*. SciTech Publishing.

Sakurai, Y., Yoshikawa, M., Kataoka, R. and Uemura, S., 2001. Similarity search for adaptive ellipsoid queries using spatial transformation. In: *VLDB*, pp. 231–240.

Schmitt, M., Tupin, F. and Zhu, X. X., 2017. Fusion of SAR and optical remote sensing data - challenges and recent trends. In: *Proceedings of IEEE Geoscience and Remote Sensing Symposium*.

Schütz, M., 2016. *Potree: Rendering large point clouds in web browsers*. PhD thesis, Technische Universität Wien, Wien.

Schütz, M., 2018. *Potree*. <http://potree.org/> (January 2018).

Stansbury, D., 2013. The statistical whitening transform. <https://theclevermachine.wordpress.com/tag/eigenvalue-decomposition/> (March 2013).

Tupin, F., 2010. Fusion of optical and SAR images. In: *Radar Remote Sensing of Urban Areas*, Springer, pp. 133–159.

van Meijeren, H., 2017. Assessing the differences between Dutch elevation datasets AHN2 and AHN3. MSc thesis.

van Oosterom, P., Martinez-Rubi, O., Ivanova, M., Horhammer, M., Geringer, D., Ravada, S., Tijssen, T., Kodde, M. and Gonçalves, R., 2015. Massive point cloud data management: Design, implementation and execution of a point cloud benchmark. *Computers & Graphics* 49, pp. 92–125.

Vosselman, G. and Maas, H.-G., 2010. *Airborne and Terrestrial Laser Scanning*. Whittles Publishing.

Wang, Y., Zhu, X. X., Zeisl, B. and Pollefeys, M., 2017. Fusing meter-resolution 4-D InSAR point clouds and optical images for semantic urban infrastructure monitoring. *IEEE Transactions on Geoscience and Remote Sensing* 55(1), pp. 14–26.

Werninghaus, R. and Buckreuss, S., 2010. The TerraSAR-X mission and system design. *IEEE Transactions on Geoscience and Remote Sensing* 48(2), pp. 606–614.