Enabling the Processing of Sentinel-1 TOPS Data with the Open-Source DORIS Software (PPT)

van Leijen, Freek; Iannini, Lorenzo; Maljaars, Hanno; Hanssen, Ramon; Caro Cuenca, Miguel; Wenhao, Wu; Aguilera, Esteban

Publication date
2014

Document Version
Final published version

Citation (APA)
van Leijen, F., Iannini, L., Maljaars, H., Hanssen, R., Caro Cuenca, M., Wenhao, W., & Aguilera, E. (2014). *Enabling the Processing of Sentinel-1 TOPS Data with the Open-Source DORIS Software (PPT)*. INSARAP Workshop, Frascati (rome), Italy.

Enabling the Processing of Sentinel-1 TOPS Data with the Open-Source DORIS Software

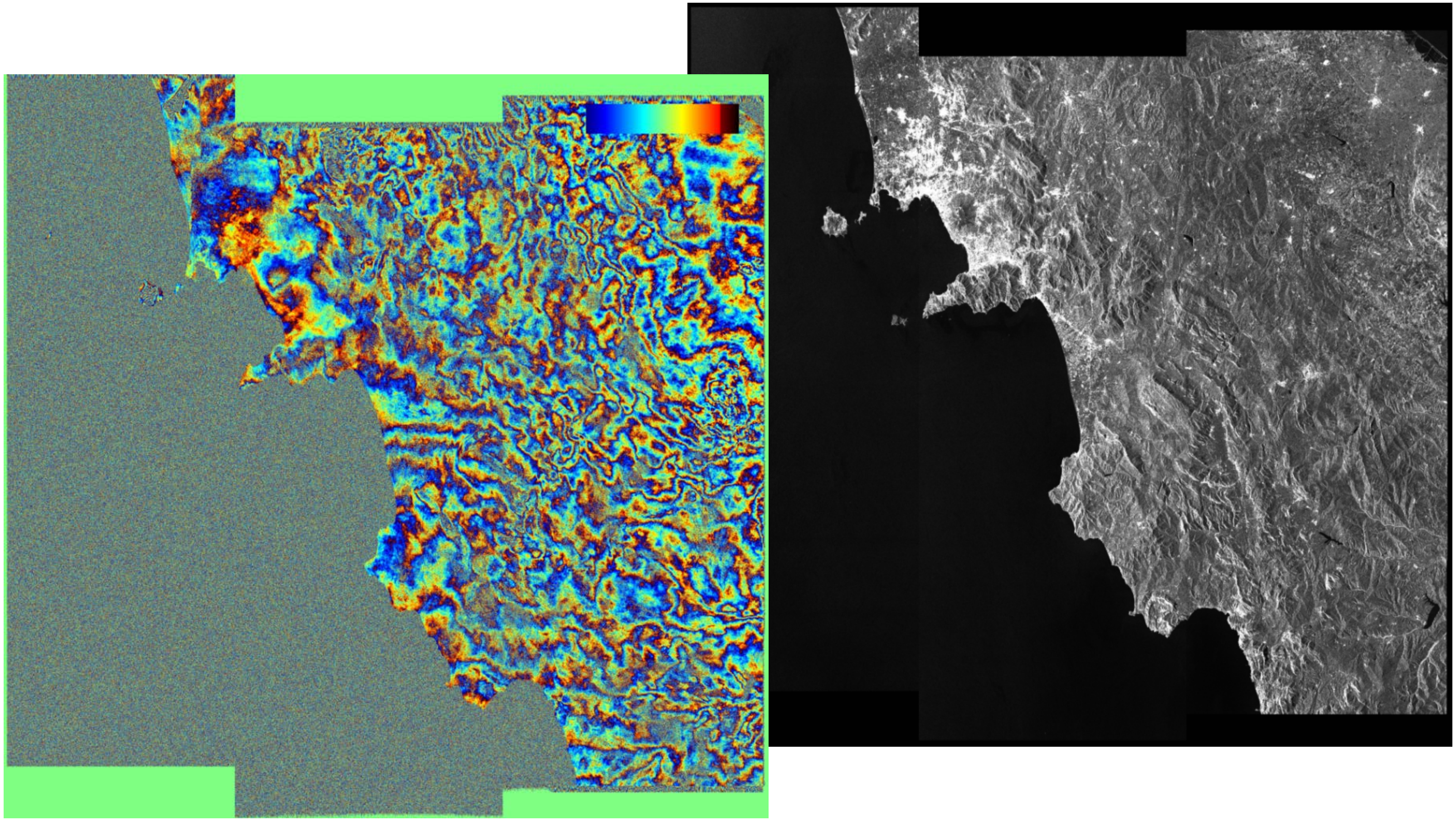
Freek van Leijen, Lorenzo Iannini, Hanno Maljaars, Ramon Hanssen,
Delft University of Technology

Miguel Caro Cuenca, TNO

Wu Wenhao, Wuhan University

Esteban Aguilera, SkyGeo/Hansje Brinker BV

Sentinel-1 IW processing with DORIS: Naples



DORIS open-source software

- Enabled interferometric applications in the last 15 years (ERS-1/2, Envisat, Radarsat-1/2, ALOS, TerraSAR-X, Cosmo-Skymed)
- Implemented in C++
- Based on a modular structure
- Designed for single master-slave combinations
- Various users created a custom-made shell for stack processing (in-house or open-source, e.g., STAMPS, ADORE)

DORIS for Sentinel-1

Development in 3 stages:

1. Design and prototyping of new processing chain – ~DONE
2. Testing and evaluation of processing settings – ONGOING
3. Final implementation – JUST STARTED

DORIS for Sentinel-1

- Requires an integration module around the DORIS core to merge the different bursts/sub-swaths
- DORIS core for processing on burst level

DORIS for Sentinel-1 implementation

- Extension of the existing DORIS core to enable TOPS mode
  - C++
  - New modules (de-ramping spectrum, re-ramping spectrum, spectral diversity)
  - For processing on burst level
- Integration module around the DORIS core
  - Python, using GDAL libraries
  - Stack processing, merging of bursts/sub-swaths

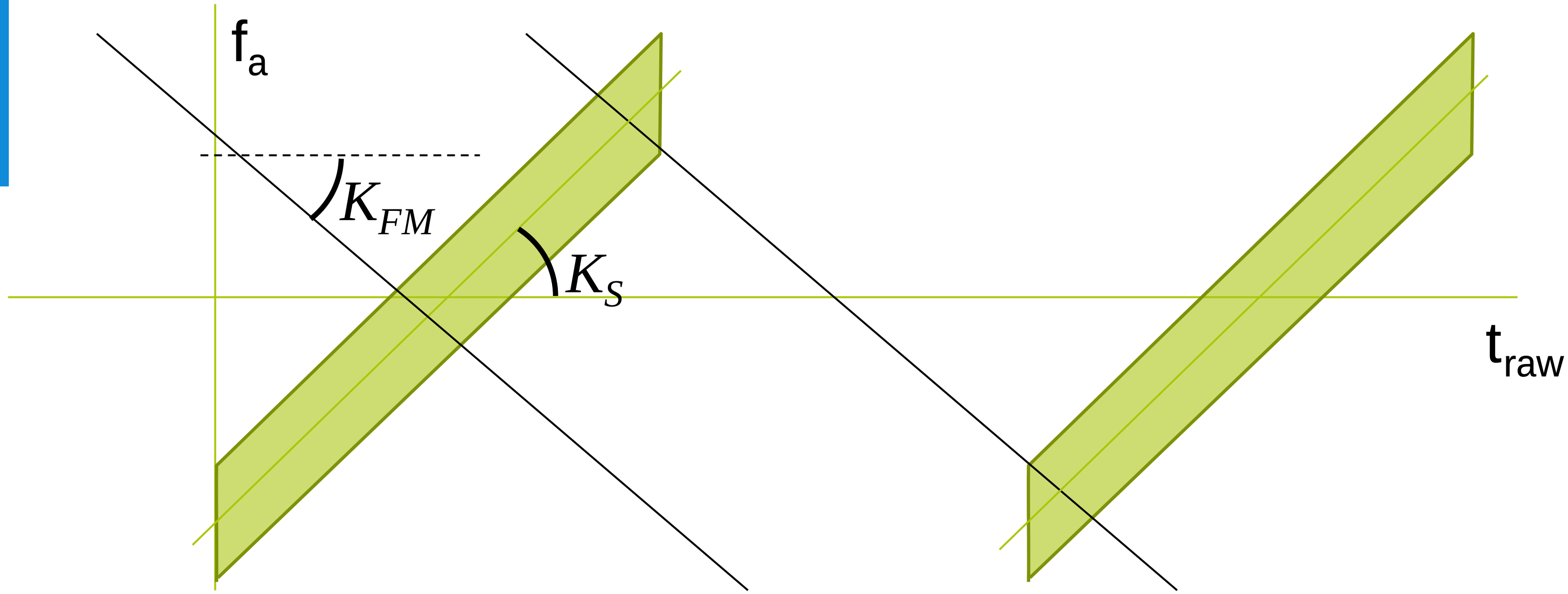
New processing flow

1. Reading of data
2. Deramping of spectrum
3. Coregistration
4. Resampling of slave
5. Reramping of spectrum
6. Computation of interferograms
7. Estimation of phase offset/azimuth shift on sub-swath/full-swath level
8. Phase correction per burst
9. Merging of bursts/sub-swaths

Data Reader

- Python, based on GDAL library
- Extraction of valid pixels

Deramping/Reramping: Azimuth FM



- Frequency modulation is the Doppler rate experienced by targets in azimuth raw times. Second order model with range:

$$K_{FM}(t_r) = c_2\left(t_r - t_r^{REF}\right)^2 + c_1\left(t_r - t_r^{REF}\right) + c_0$$

generalAnnotation/AzimuthFmRate/t0
generalAnnotation/AzimuthFmRate/c0
generalAnnotation/AzimuthFmRate/c1
generalAnnotation/AzimuthFmRate/c2

- Different from effective rate K_{AZ} in the focused image. Conversion from raw time to focused time need to be performed

Doppler centroid retrieval

- Doppler centroid model
  - t_r: two-way range time
  - t_a: azimuth focused time

$$f_{DC}(t_r, t_a) = f_{DC}^{REF}(t_r) + K_{AZ}(t_r)\left(t_a - t_a^{REF}\right)$$

$$f_{DC}^{REF}(t_r) = d_2\left(t_r - t_r^{REF}\right)^2 + d_1\left(t_r - t_r^{REF}\right) + d_0$$

dopplerCentroid/dcEstimate/AzimuthTime

- Extract platform velocity v_s from orbit
- Convert steering rate K_{sr} in Hz/s

$$K_S = \frac{2v_s}{\lambda} K_{sr} \frac{\pi}{180}$$

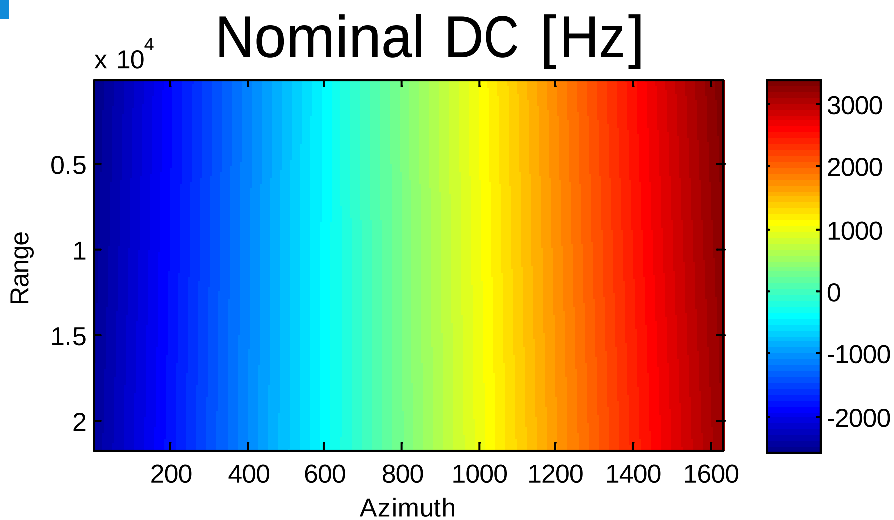
…/productInformation/azimuthSteeringRate

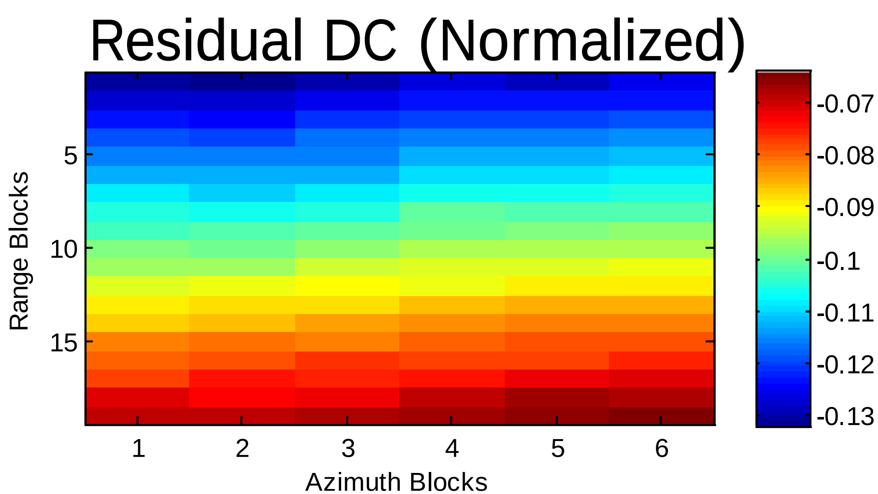
- Raw time -> Focused time

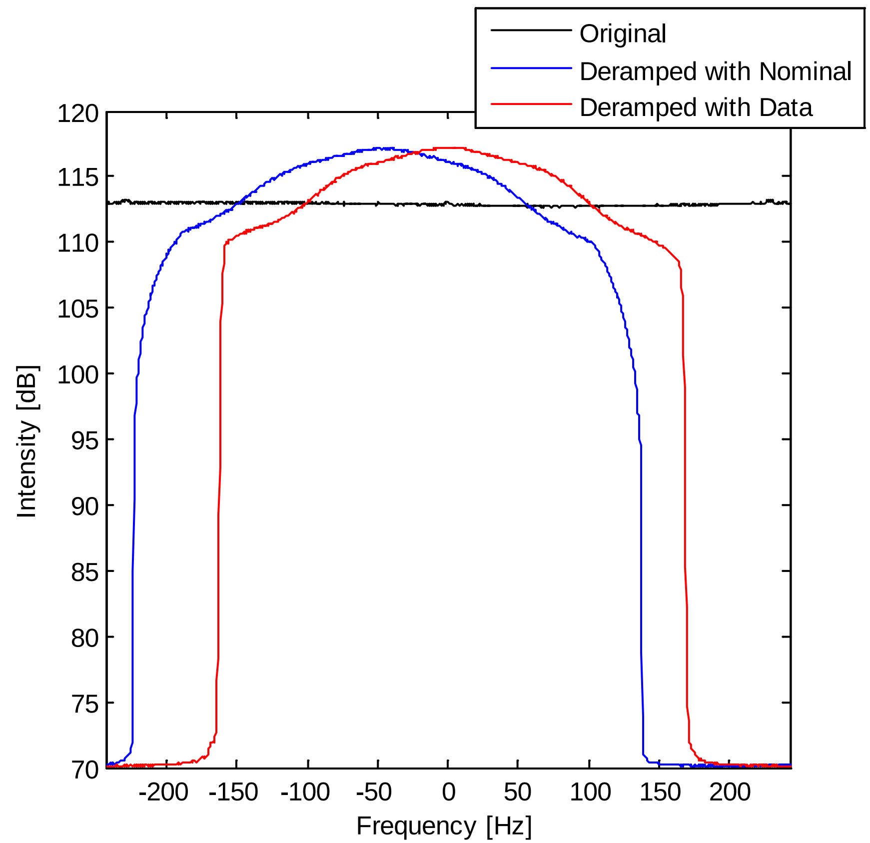
$$\alpha(t_r) = 1 - \frac{K_S}{K_{FM}(t_r)} \rightarrow K_{AZ}(t_r) = \frac{K_S}{\alpha(t_r)}$$

Deramping

Results on Naples scene - Subswath 1, Burst 01







Deramping

- Problem in ${f_{DC}}^{REF}$ polynomial -> residual spectral shift to be compensated

Current approach:

- A residual polynomial is estimated from the data according to:

$$f_{DC}^{EST}(t_r) = (d_2 + \Delta d_2)(t_r - t_r^{REF})^2 + (d_1 + \Delta d_1)(t_r - t_r^{REF}) + (d_0 + \Delta d_0)$$

$$K_{AZ}^{EST}(t_r) = K_{AZ} + \Delta K_{AZ}$$

NECESSARY *(at least for early S1 images)*

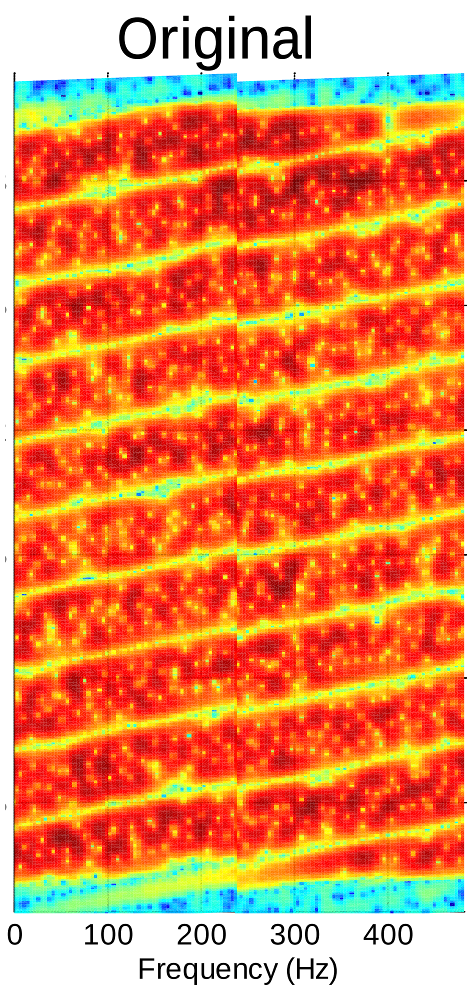
OPTIONAL -> K_{AZ} from the annotation is accurate enough

- The deramping chirp is then computed as:

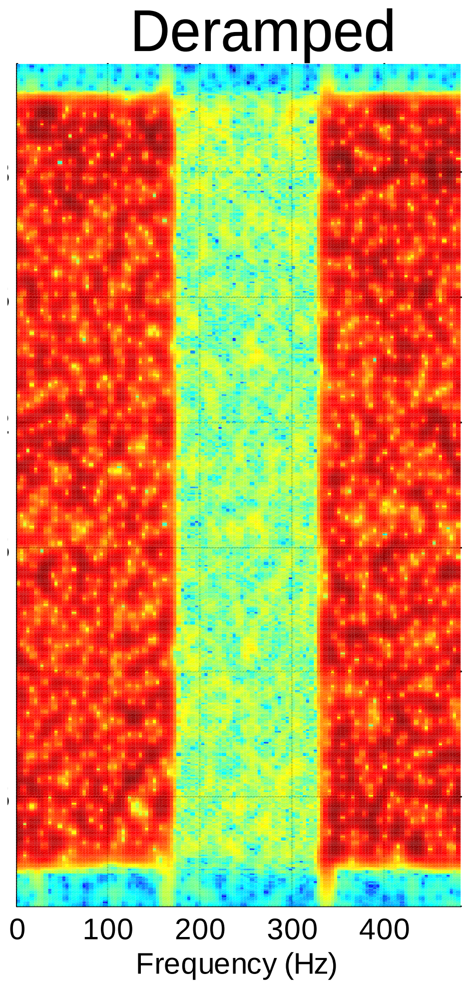
$$C(t_r, t_a) = \exp\left(j\pi\left(K_{AZ}^{EST}(t_r)(t_a - t_a^{REF}) + f_{DC}^{EST}(t_r)\right)(t_a - t_a^{REF})\right)$$

TU Delft

Deramping



Original
0
100
200
300
400
Frequency (Hz)



Deramped
0
100
200
300
400
Frequency (Hz)

Reramping

- Multiplication by inverse chirp

- As resampling is performed on slave image as described by the range and azimuth pixel warping functions/DEM-based offsets:

$$t_a \rightarrow F_a(t_r, t_a)$$
$$t_r \rightarrow F_r(t_r, t_a)$$

the chirp needs to be resampled accordingly, i.e.

$$C(t_r, t_a) \rightarrow C(F_a(t_r, t_a), F_r(t_r, t_a))$$

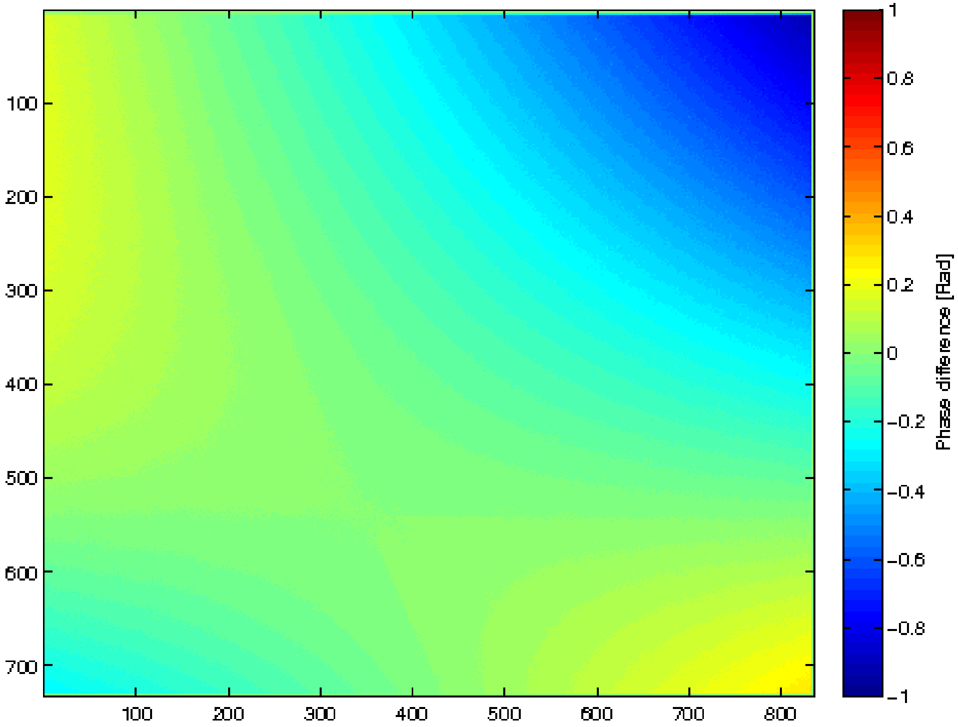
Coregistration

Four methodologies implemented:

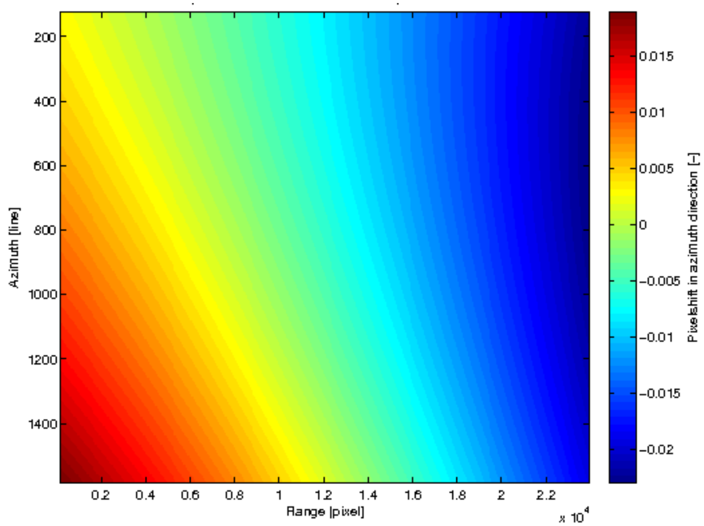
1. Incoherent Cross-Correlation (ICC)
2. Coherent Cross-Correlation (CCC)
3. DEM-based coregistration
4. Spectral Diversity (in combination with one of the other methodologies)

Comparison of methodologies: burst level

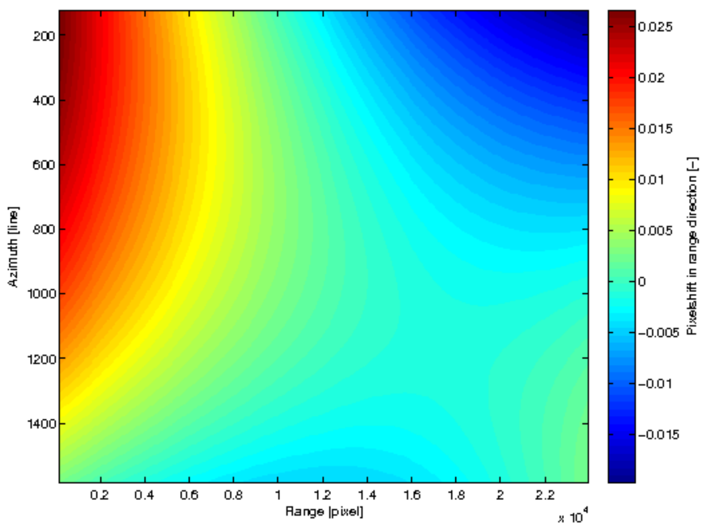
Diffference ICC point scatterers – ICC random points



Pixel shift Azimuth

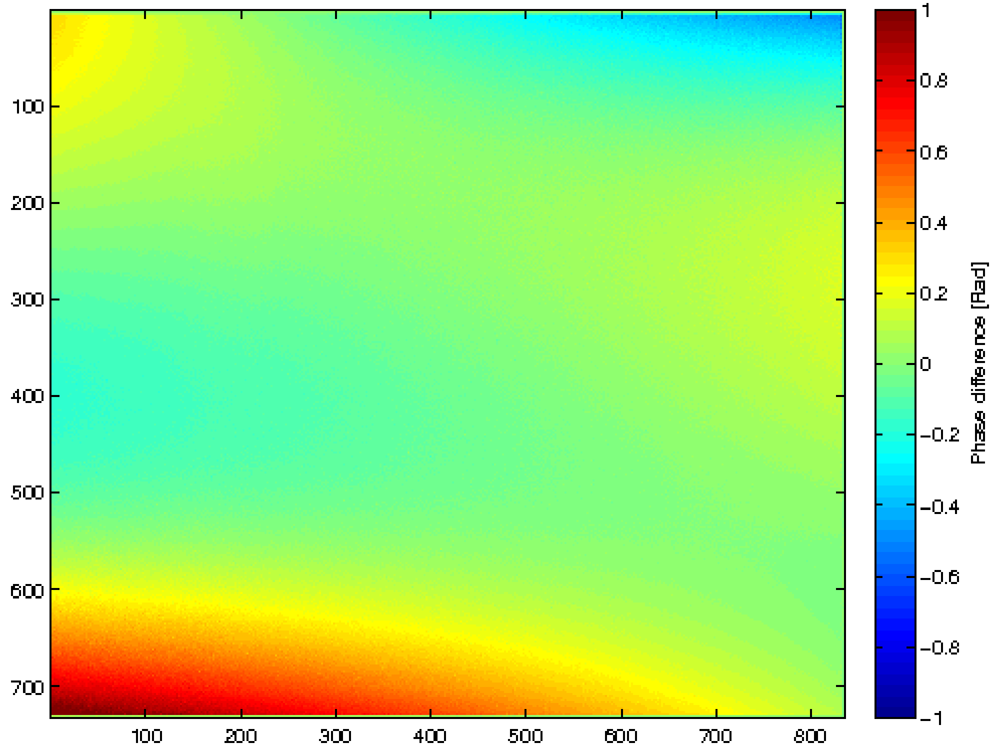


Pixel shift Range

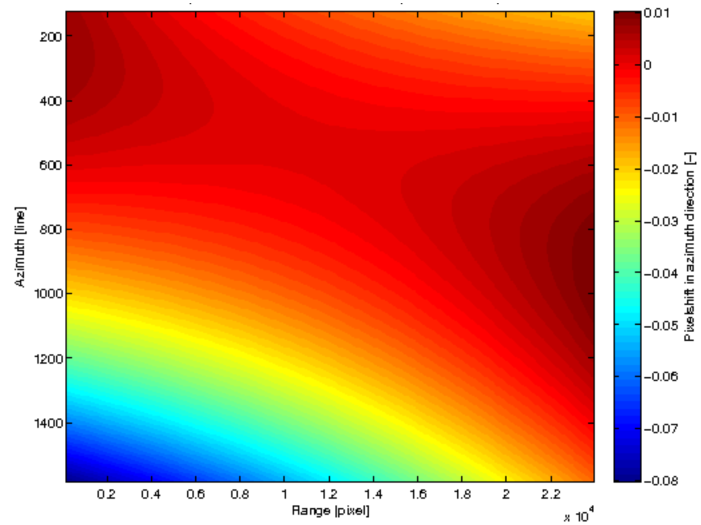


Comparison of methodologies: burst level

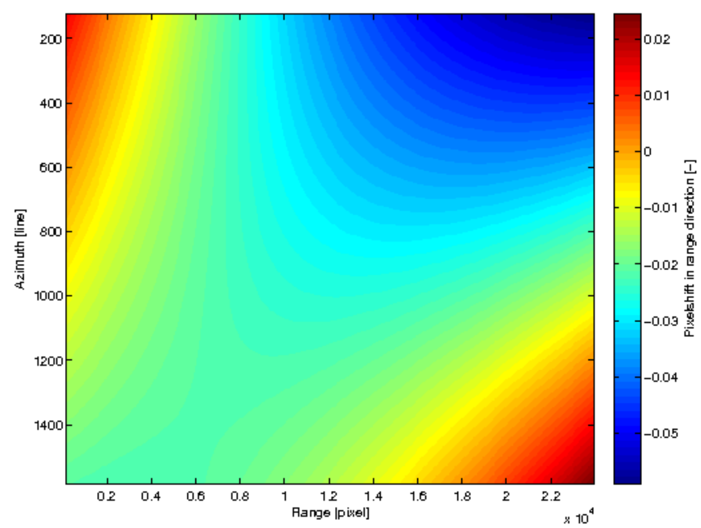
Diffference CCC point scatterers – ICC point scatterers



Pixel shift Azimuth

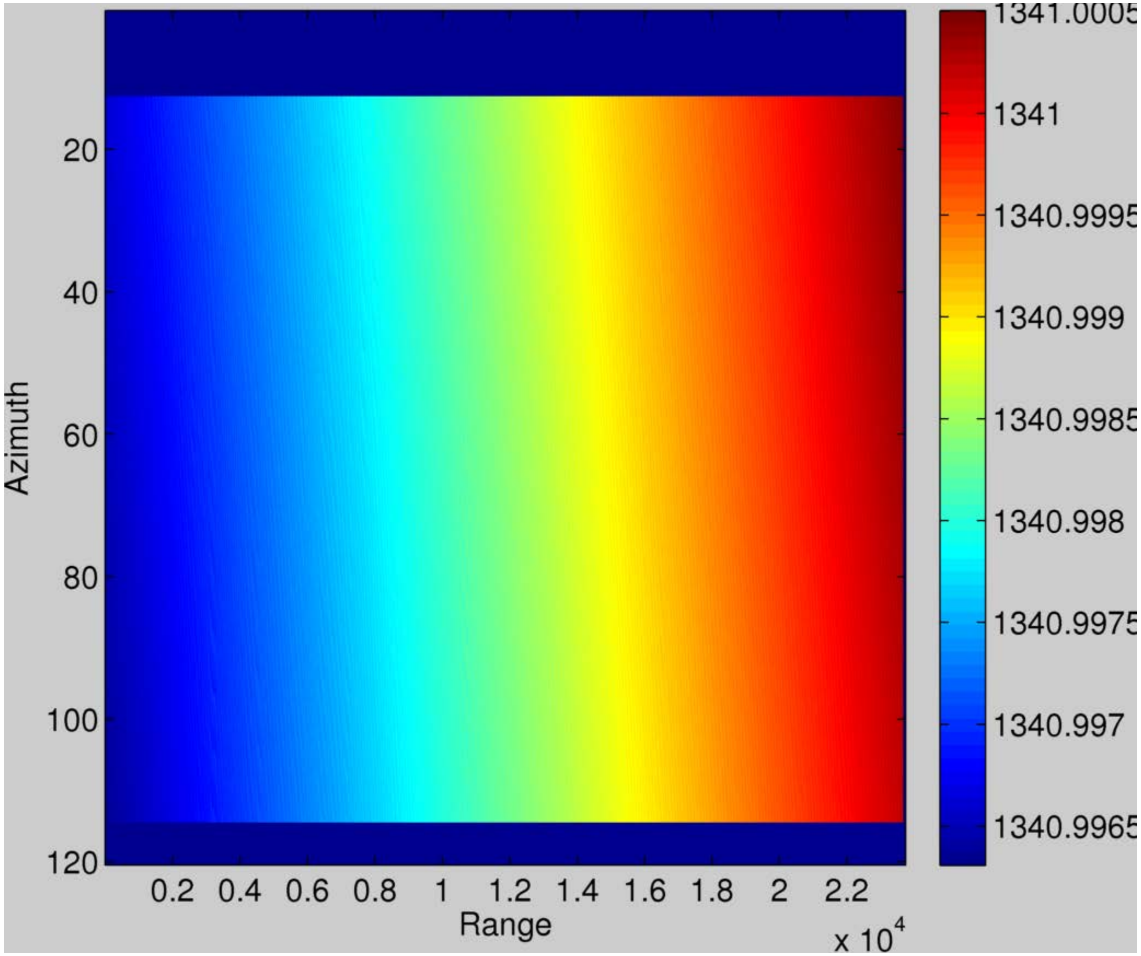


Pixel shift Range



Assessment of consistency: burst overlaps

Diffference range shift burst overlap ICC point scatterers
(1-degree polynomial)



Consistency in coregistration

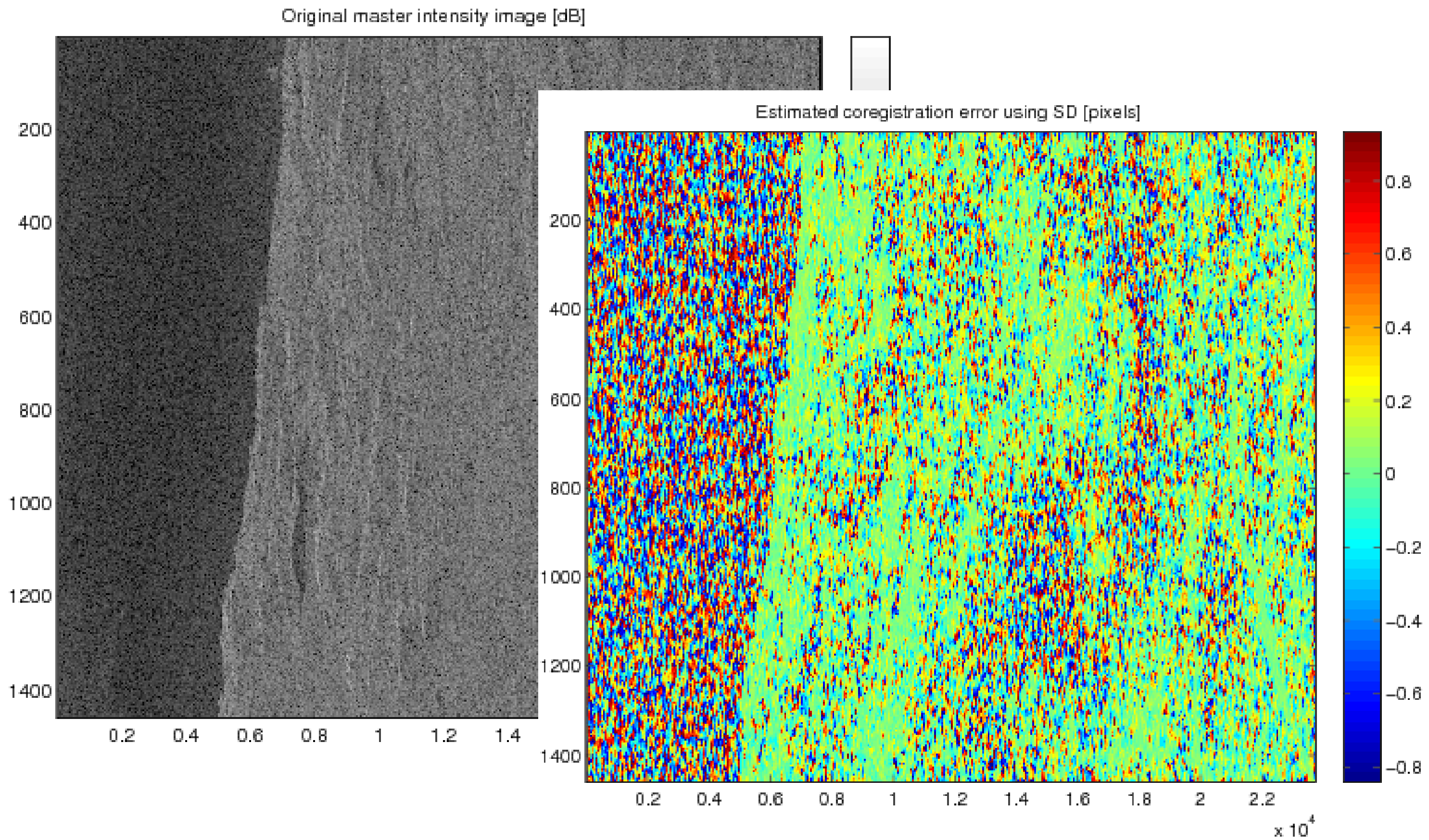
To preserve consistency in the sub-swath/full-swath:

- Single warp function per sub-swath

or

- **DEM-based coregistration**

Spectral Diversity



Currently mean shift is taken. To be changed to pixel-based offsets.

Correction based on burst overlaps

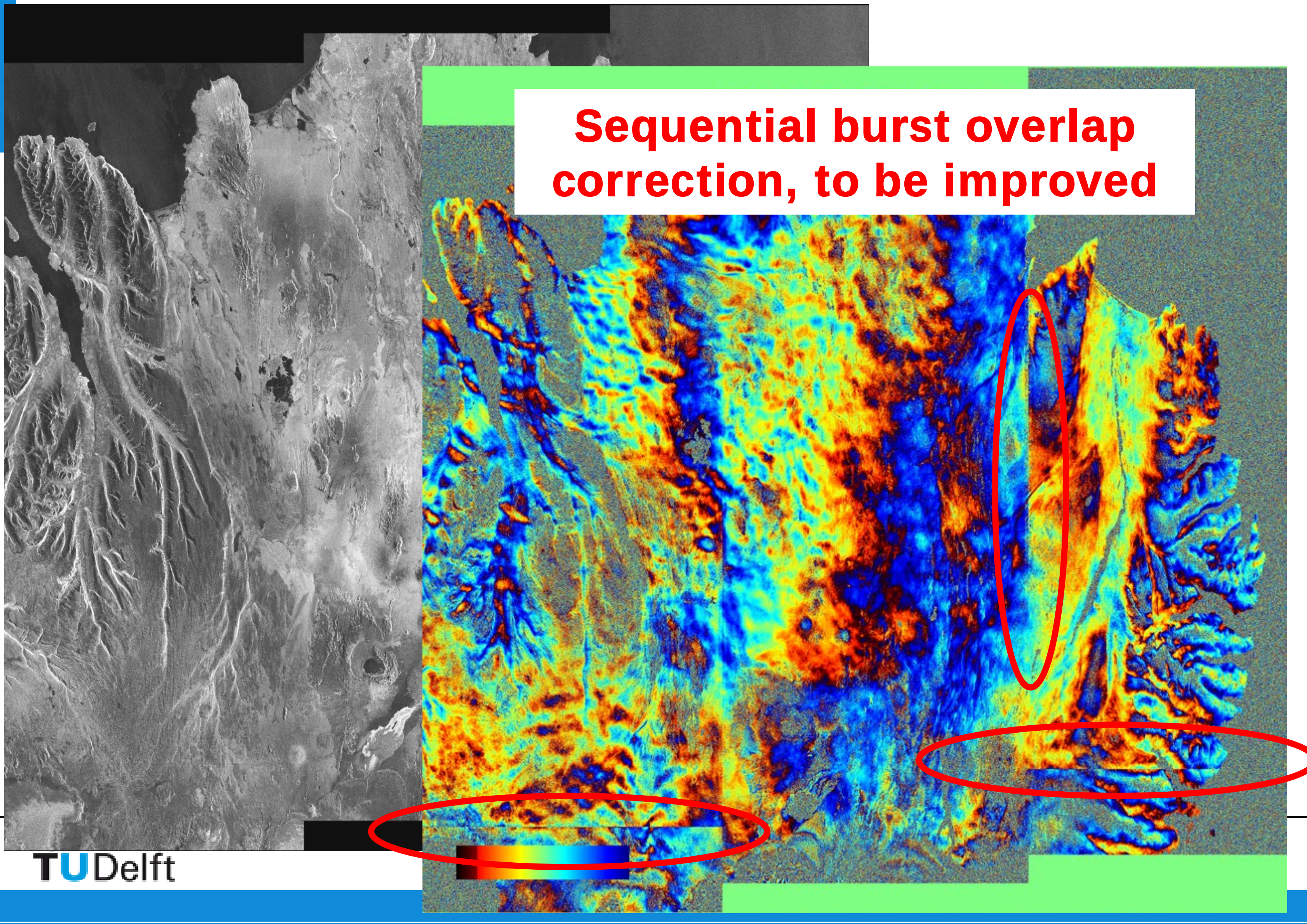
- Currently sequential correction of bursts
- To be changed to integrated correction per sub-swath/full-swath

Merging of bursts/sub-swaths

- Based on GDAL library
- Open question: what to do with burst overlap?
  - Weighted average?
  - Cut at middle of burst overlap?

Iceland

18 Oct 2014 – 30 Oct 2014



Conclusions

- Data with excellent coherence
- TOPS mode forces us to re-assess and improve our coregistration procedures, which is also usefull for other data
- Apart from the technical challenges, significant software adaptions are required for the administration (merging of bursts)
- Correction of azimuth shifts requires further improvement