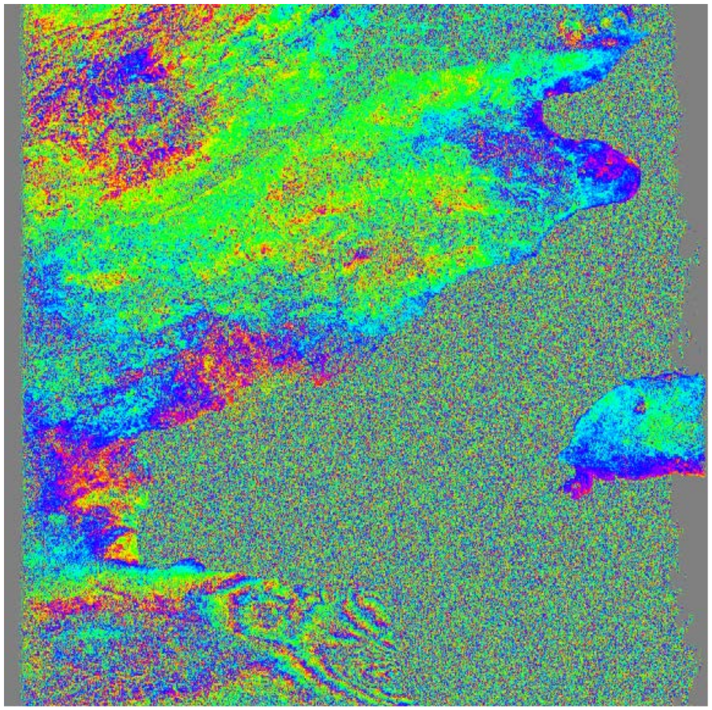
~~Filtering~~, unwrapping~~, and geocoding~~*

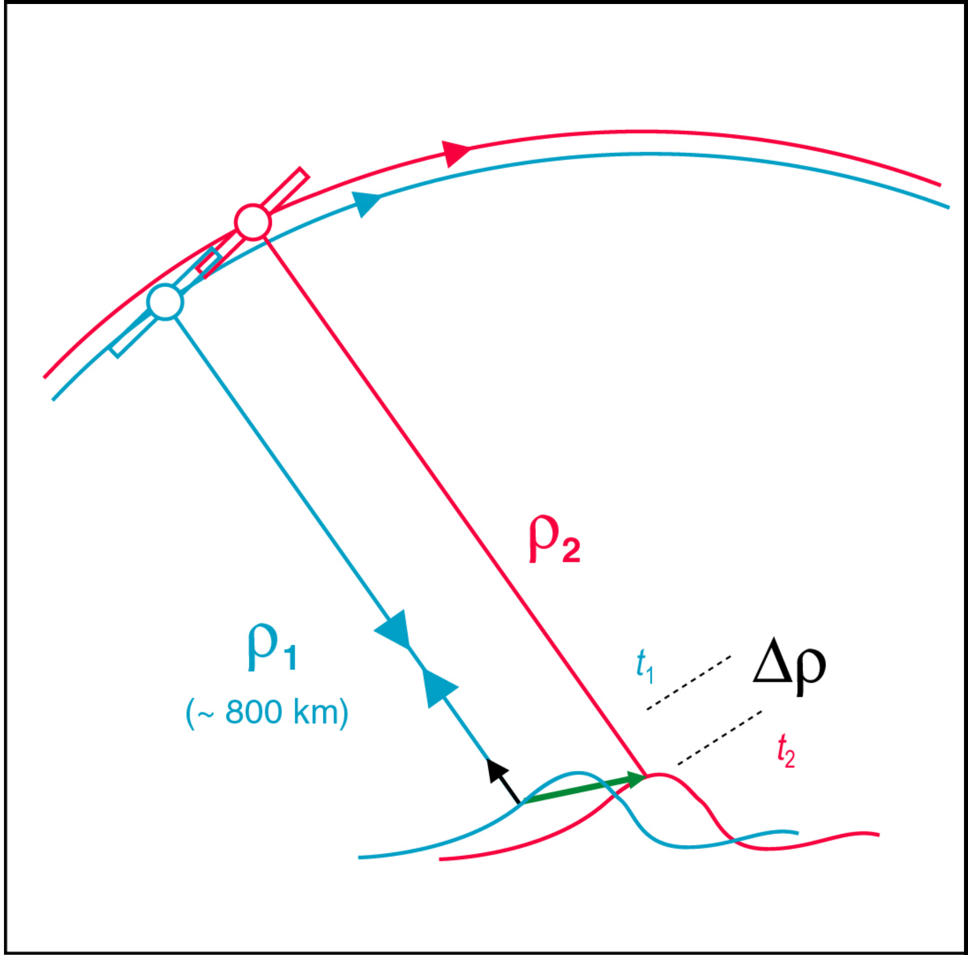
Rob Mellors
Eric Lindsey, Ph.D.
Kurt Feigl



- Wrapping your head around the problem
- Wrapping operator
- Residues
- using SNAPHU in GMTSAR
  - Eric Lindsey – interpolation
  - Xiaohua (Eric) Xu – masking
- Reduce or avoid the problem
  - Filter
  - Model phase wrapped phase directly
  - Use gradient

*or what to do when your interferogram looks like this

INSAR geometry



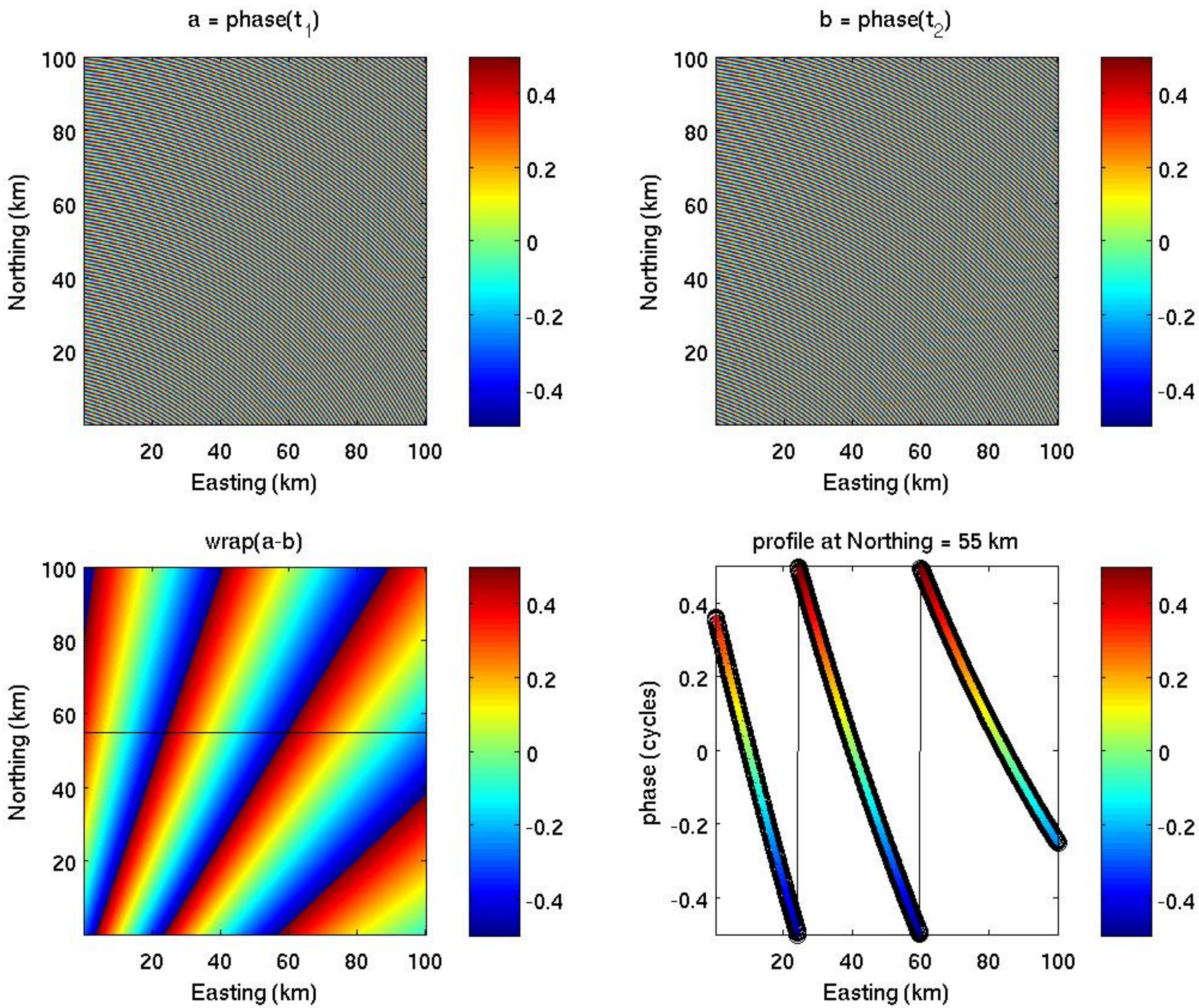
Range Change

$$\Delta\rho = \frac{\lambda}{2}\left[\phi(t_2) - \phi(t_1)\right]$$

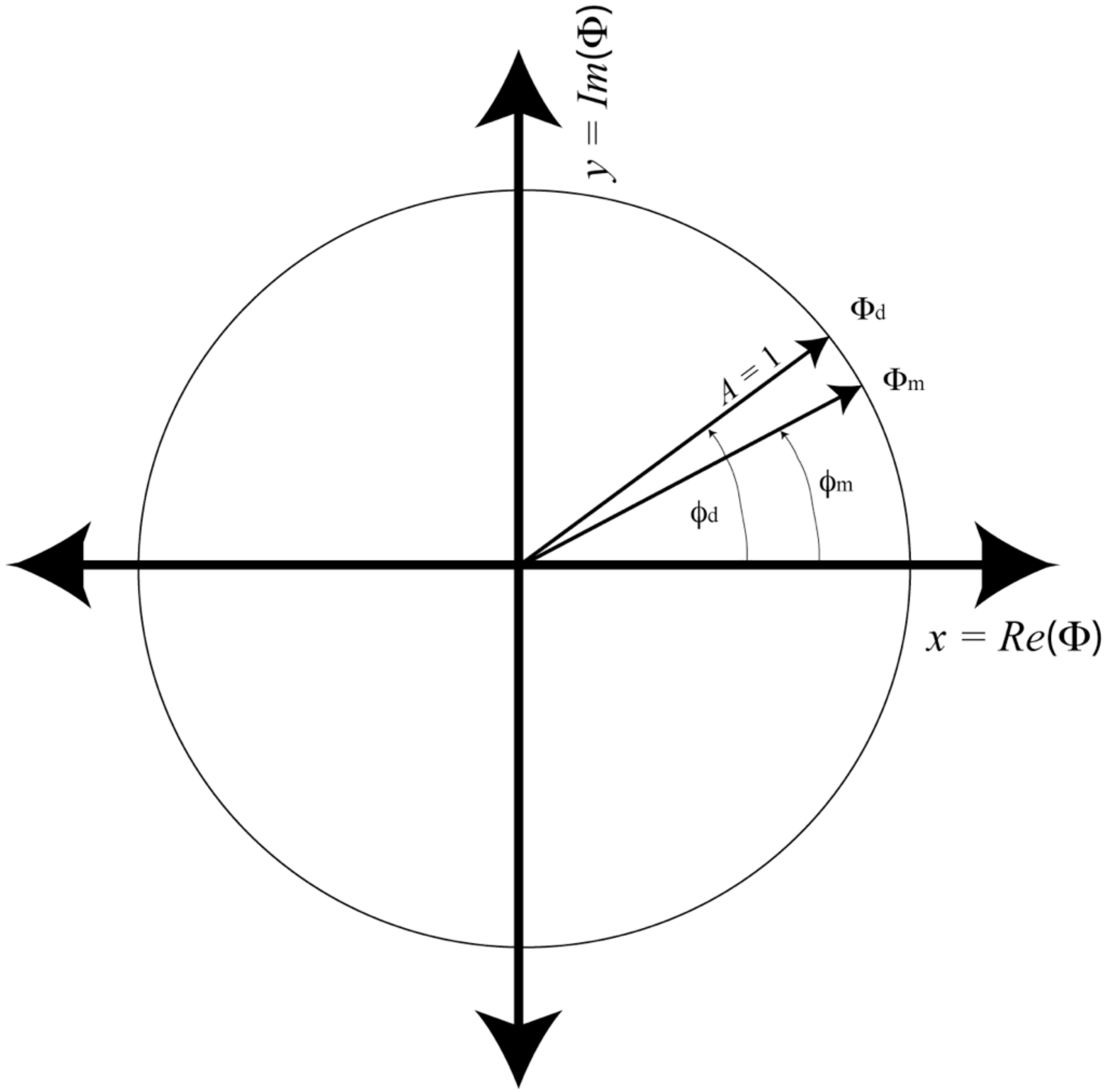
$$\Delta\rho = -\mathbf{u} \bullet \hat{\mathbf{s}}$$

- First image at t_1
- Second image at t_2
- Phase shift => range change
- Component of ground displacement along radar line of sight **s**
- Increasing range Δρ away from satellite
- Range is most sensitive to vertical component of displacement
- Motion parallel to ground track of satellite does not change range

Map of phase shift shows fringes

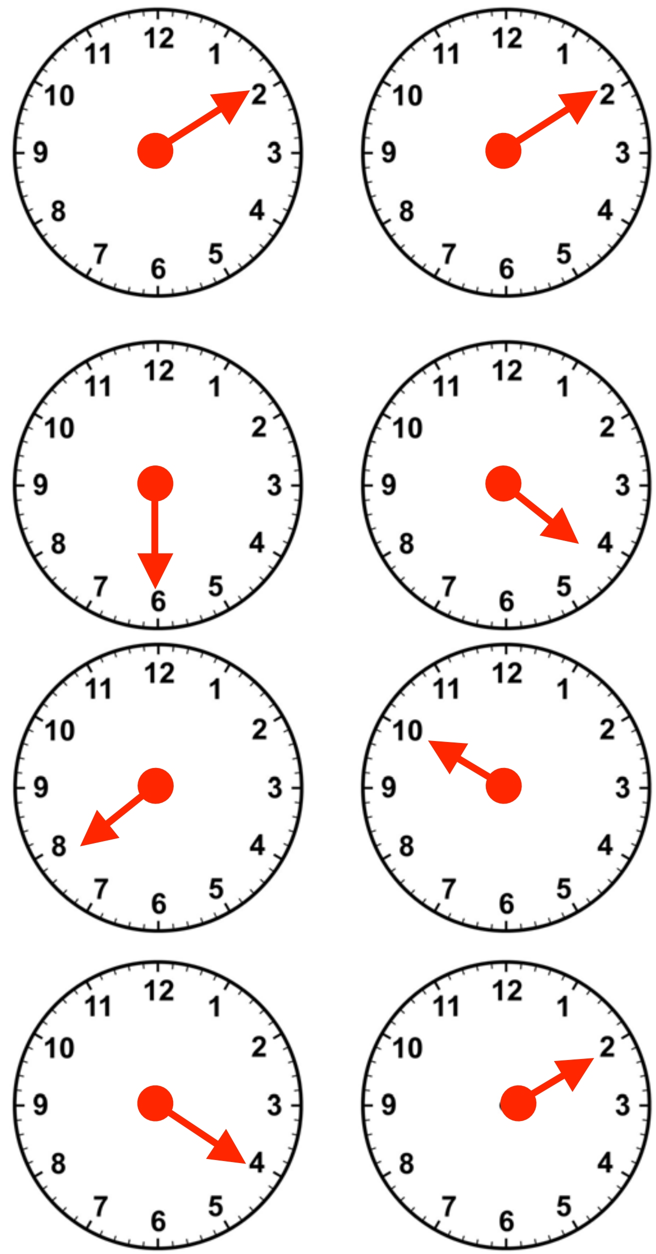


Phase is a vector on unit circle



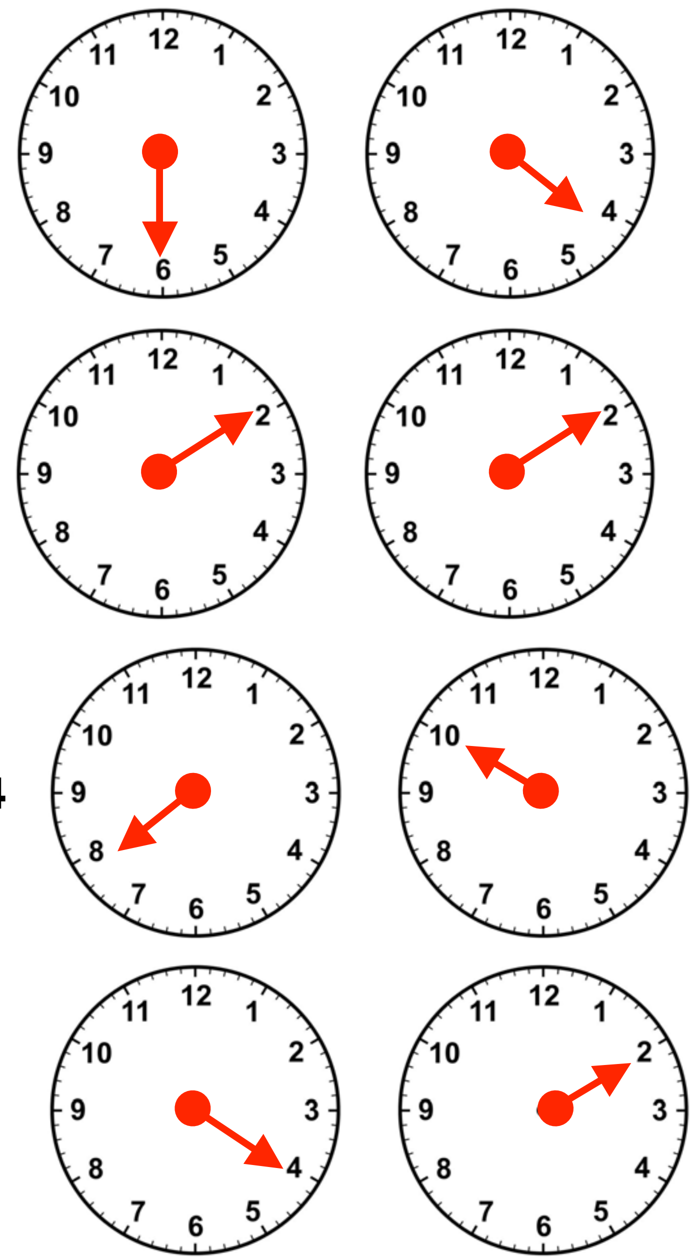
wrap(t)

	Initial	Final
Observed	2	2
Modeled	6	4
Residual	2 – 6 = −4 wrap(2-6)=+8	2 – 4 = −2 wrap(2-4) =+10
Deviation	arc(2,6) = 4	arc(2,4) = 2



wrap(t)

Initial Final



Observed 6 4

Modeled 2 2

Residual
6 – 2 = 4
wrap(6 – 2) = 4
4 - 2= 2
wrap(4 – 2) =4

Deviation
arc(6, 2) = 4
arc(4, 2) = 2

Extracting the phase

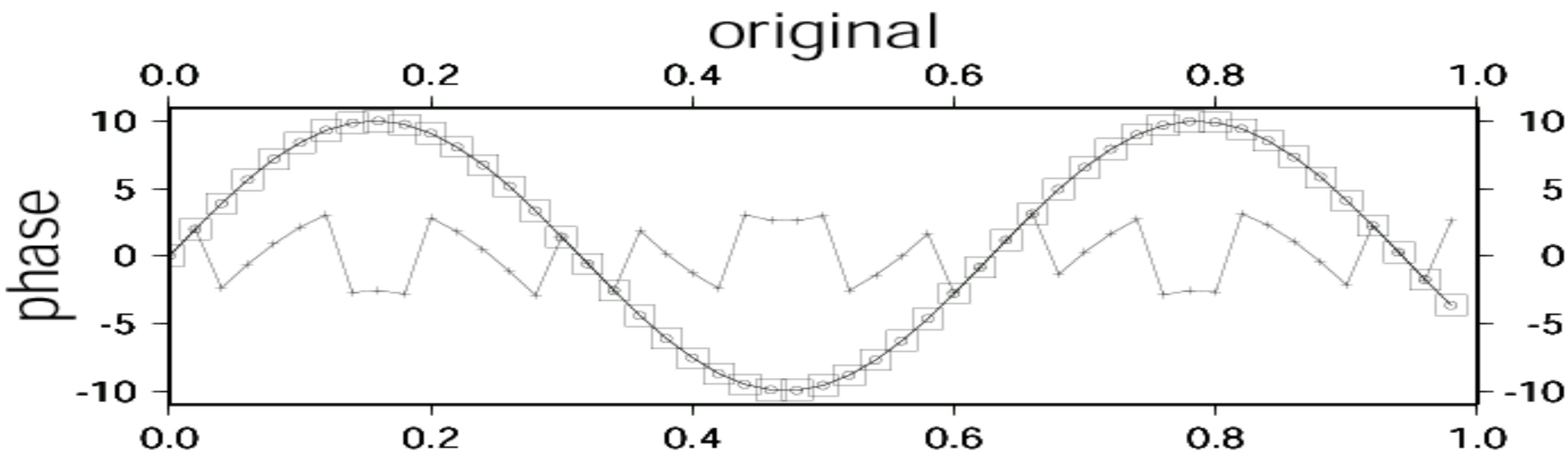
We can get only the wrapped phase*

$$\Phi(t) = \arctan(I(s(t)),R(s(t)))$$

where $-\pi < \Phi(t) \leq \pi$

We would like the continuous phase.

This appears simple. Look for 2π jumps and then add the appropriate multiple of 2π.



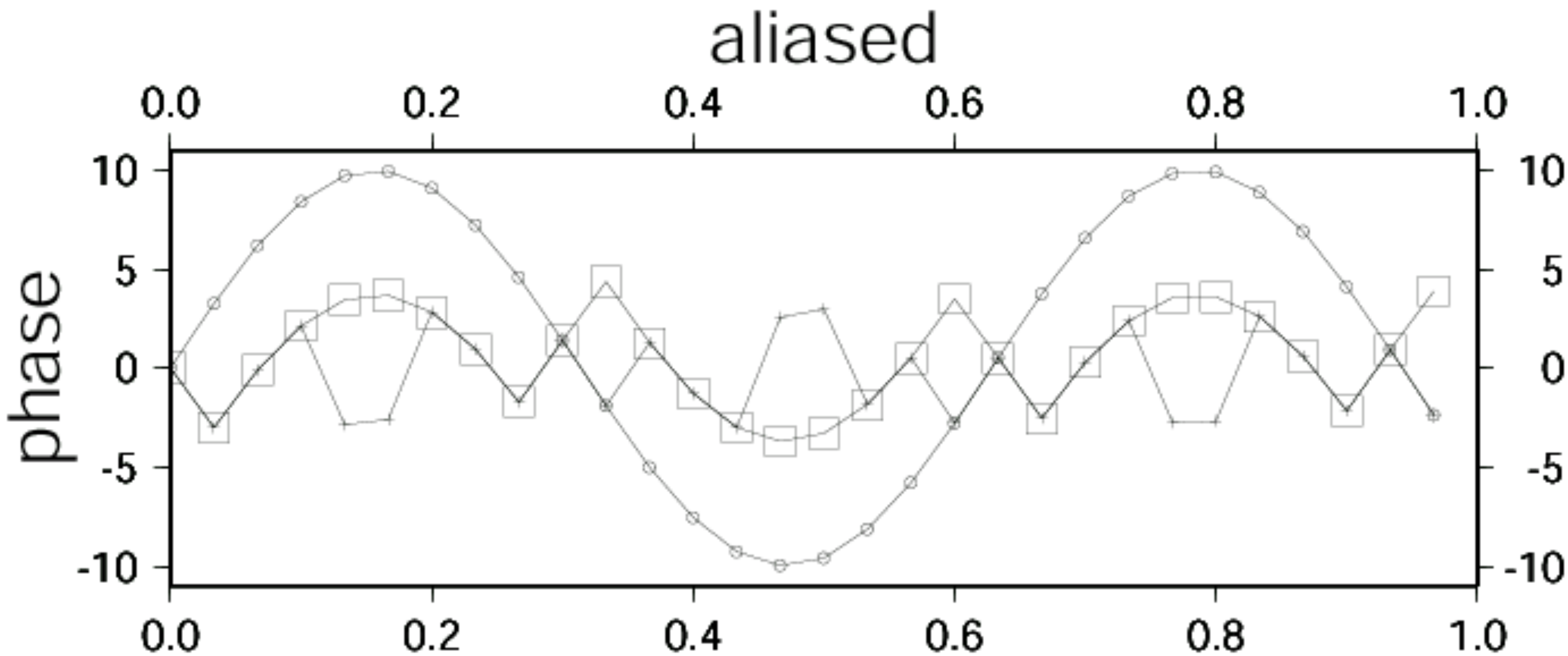
If the data are good, phase unwrapping is straightforward

•We could take derivative in complex version (e.g. Sandwell & Price)

problem number 1: aliasing

True phase changes by more than 1 cycle (2π radians) between samples.
Caused by:

- by large orbital separation,
- steep topography
- large deformation (steep phase gradient)

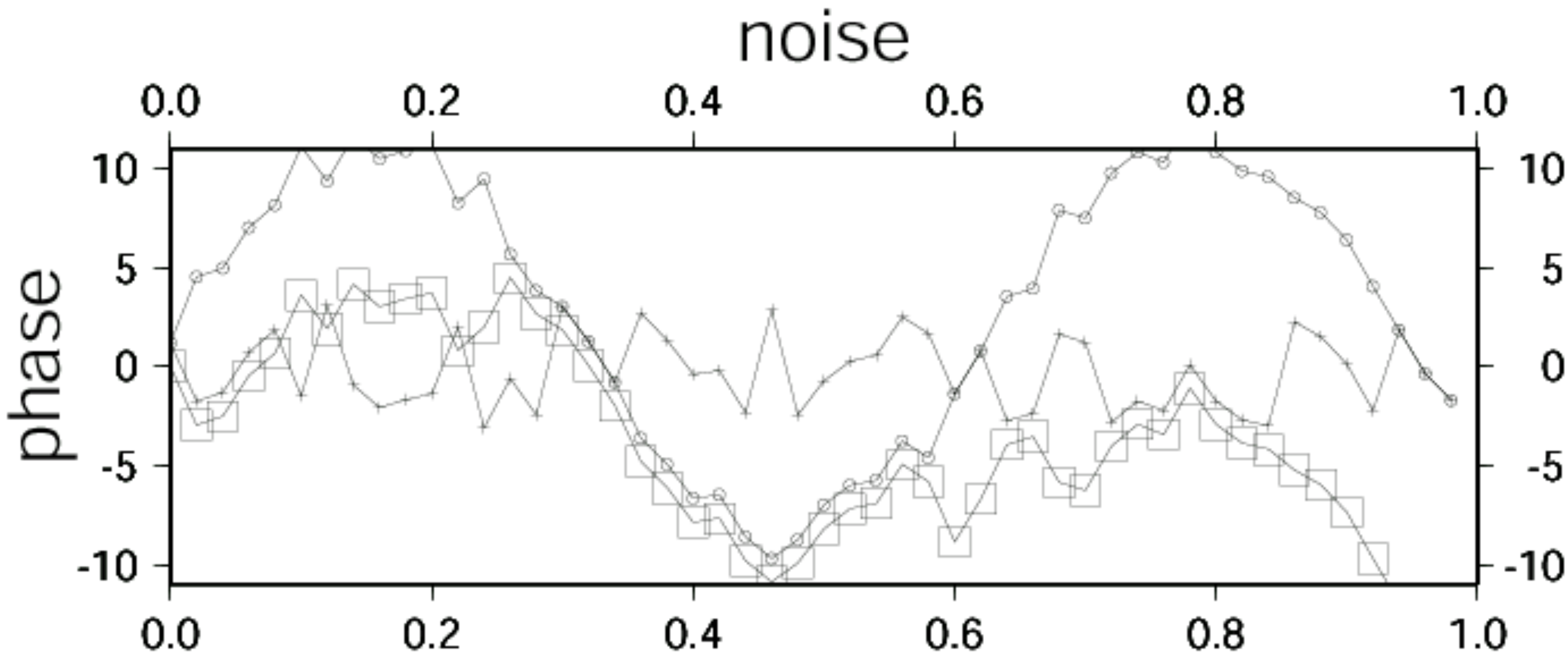


Problem Number 2: Noise and/or gaps in the data

.

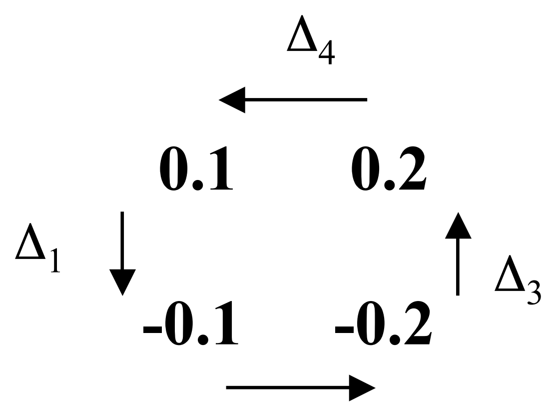
Changes on the surface surface (e.g., plowing, snow, erosion) may cause the two images to de-correlate, introducing noise.

We use **spatial coherence** to measure **temporal correlation**



Differences around a closed loop should sum to zero.

$$
\begin{aligned}
\Delta_1 &= -0.2 \\
\Delta_2 &= -0.1 \\
\Delta_3 &= \;\;0.4 \\
\Delta_4 &= \underline{-0.1} \\
\text{Sum: } & \boldsymbol{\mathit{0.0}}
\end{aligned}
$$



$\Delta_2 = -0.2 - (-0.1) = -0.1$

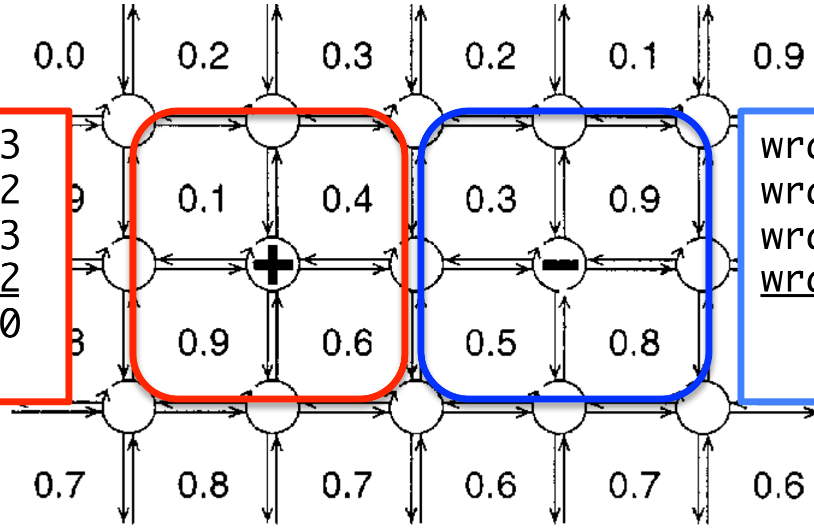
(in cycles – multiply by 2π to get radians)

We know that topographic surfaces are conservative

- Any points that violate this rule should be avoided.
- These points are known as residues.
- Any integration path that circles a residue will contain errors
  => need to make "branch cuts"
- A residue is a property of phase differences, not a single pixel.
- can be positive or negative

Wrapped differences should sum to zero around a loop. If sum is not zero, then identify the loop as a *residue.*

```
wrap(0.4 - 0.1) =  0.3
wrap(0.6 – 0.4) =  0.2
wrap(0.9 – 0.6) =  0.3
wrap(0.1 – 0.9) =  0.2
                = +1.0
```




```
wrap(0.9 – 0.3) = -0.4
wrap(0.8 – 0.9) = -0.1
wrap(0.5 – 0.8) = -0.3
wrap(0.3 – 0.5) = -0.2
                = -1.0
```

Fig. 1. Example network equivalent of the phase unwrapping problem. The numbers represent the 2-D array of phase samples (normalized to one cycle). Each 2×2 clockwise loop integral of wrapped phase gradients is a node in the network, and positive and negative residues result in supply and demand nodes. Neighboring nodes are connected by arcs, or possible flow paths. The amount of flow on an arc represents the difference (in cycles) between the unwrapped and the wrapped phase gradients associated with that arc. The net amount of flow out of a node must be equal to the node's surplus.

Chen, C. W., and H. A. Zebker (2000), Network approaches to two-dimensional phase unwrapping: intractability and two new algorithms, *J. Opt. Soc. Am. A, 17*, 401-414.

Chen and Zebker's (2000)

1) A branch-cut algorithm minimizes the length of discontinuity by an (L^0) norm.
2) Flynn yields a L^1 solution.
3) Least-squares yield an L^2 solution.

C&Z claim low norms are best.

For interferograms corrupted by noise, L^0 and L^1 algorithms yield similar solutions.
For layover, where discontinuities separate severe phase gradients, L^1 algorithms do not do well.

GMT5SAR uses SNAPHU

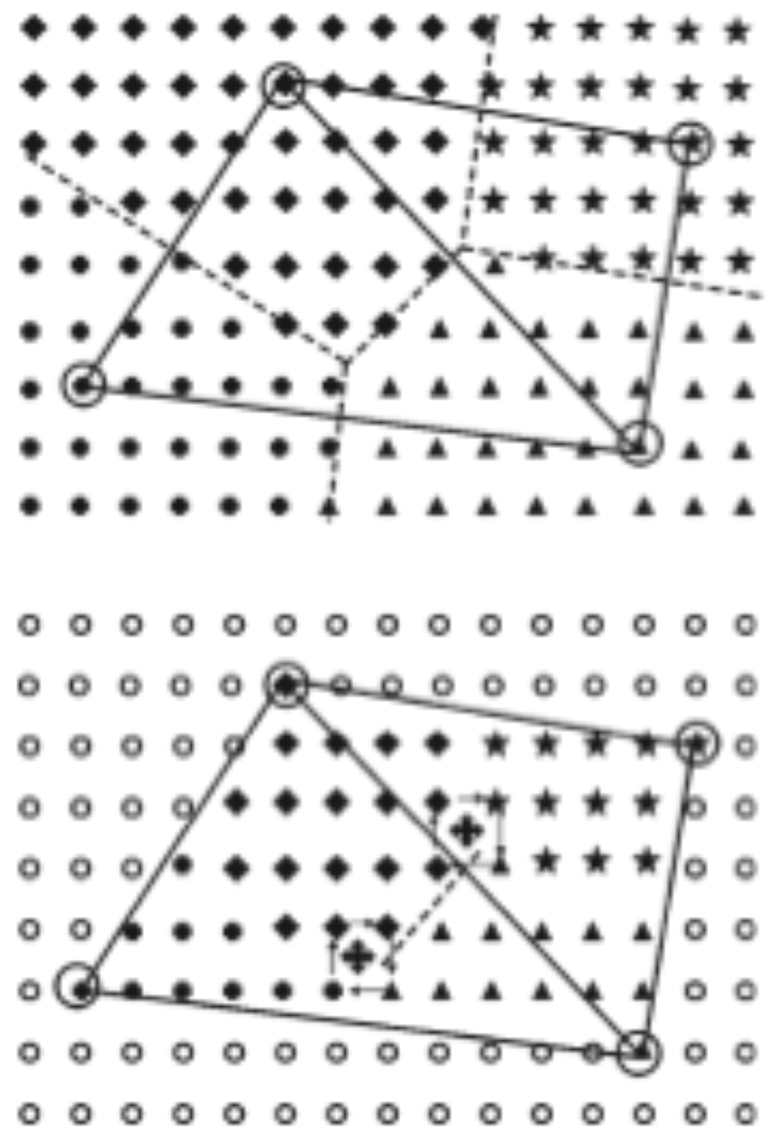
Phase unwrapping for the impatient

- **Eric Lindsey, Ph.D.**
- **GMTSAR workshop, August 2015**

- Sometimes you have a scene with large decorrelated areas
- In these cases, SNAPHU can take ~forever
- Setting the correlation threshold higher won't help (why?)
- We can use interpolation to speed the computation and improve the results

Nearest Neighbor Interpolation

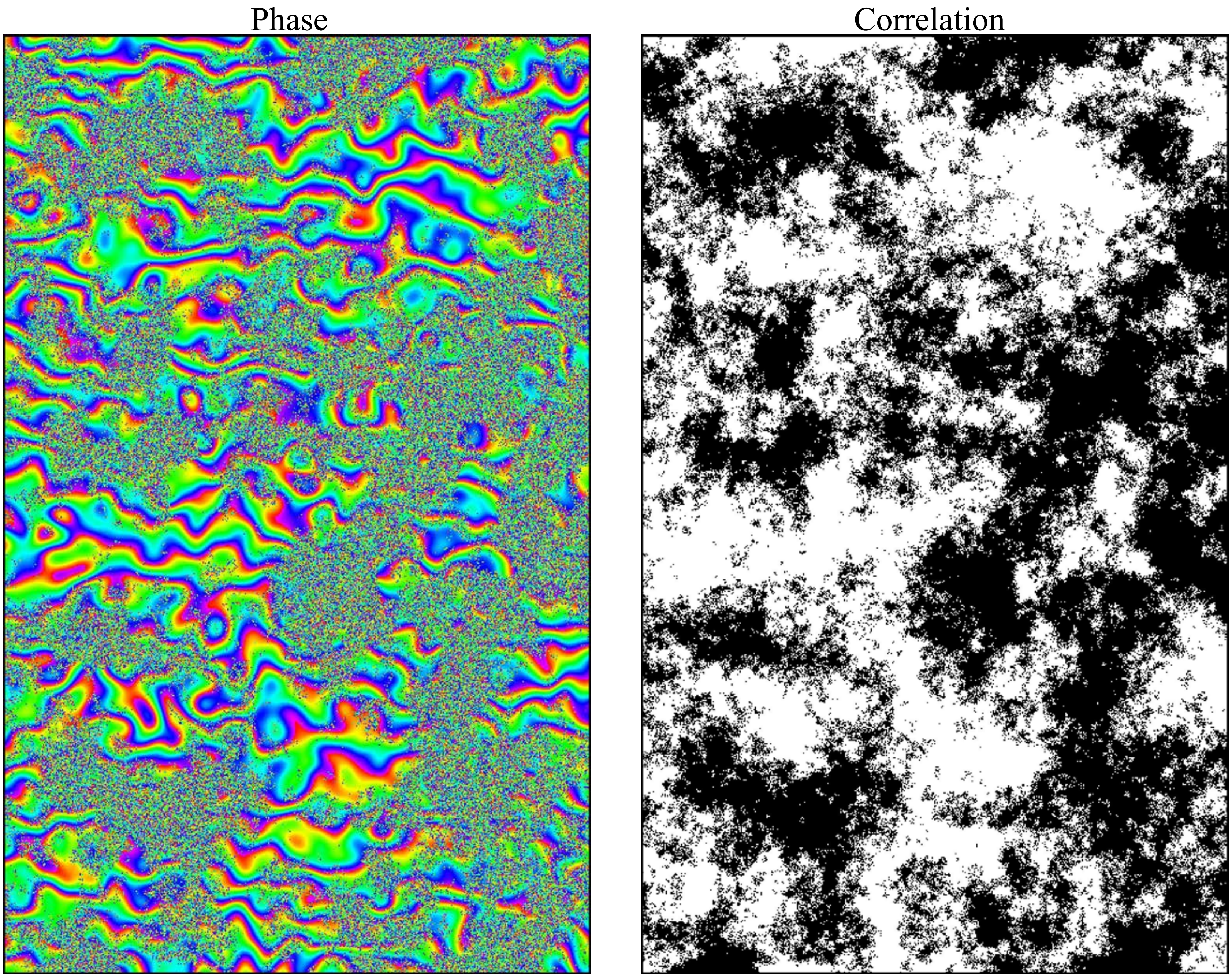
- We can think of the correlated pixels as a sparse dataset
- Nearest neighbor interpolation preserves the topology of any loops containing residues
- This means the unwrapped, masked result should be the same, whether or not we interpolate first
- Reference:
  P.S. Agram and H.A. Zebker, “Sparse two-dimensional phase unwrapping using regular grid methods,” *IEEE Geosci. Rem. Sens.*, 2009.



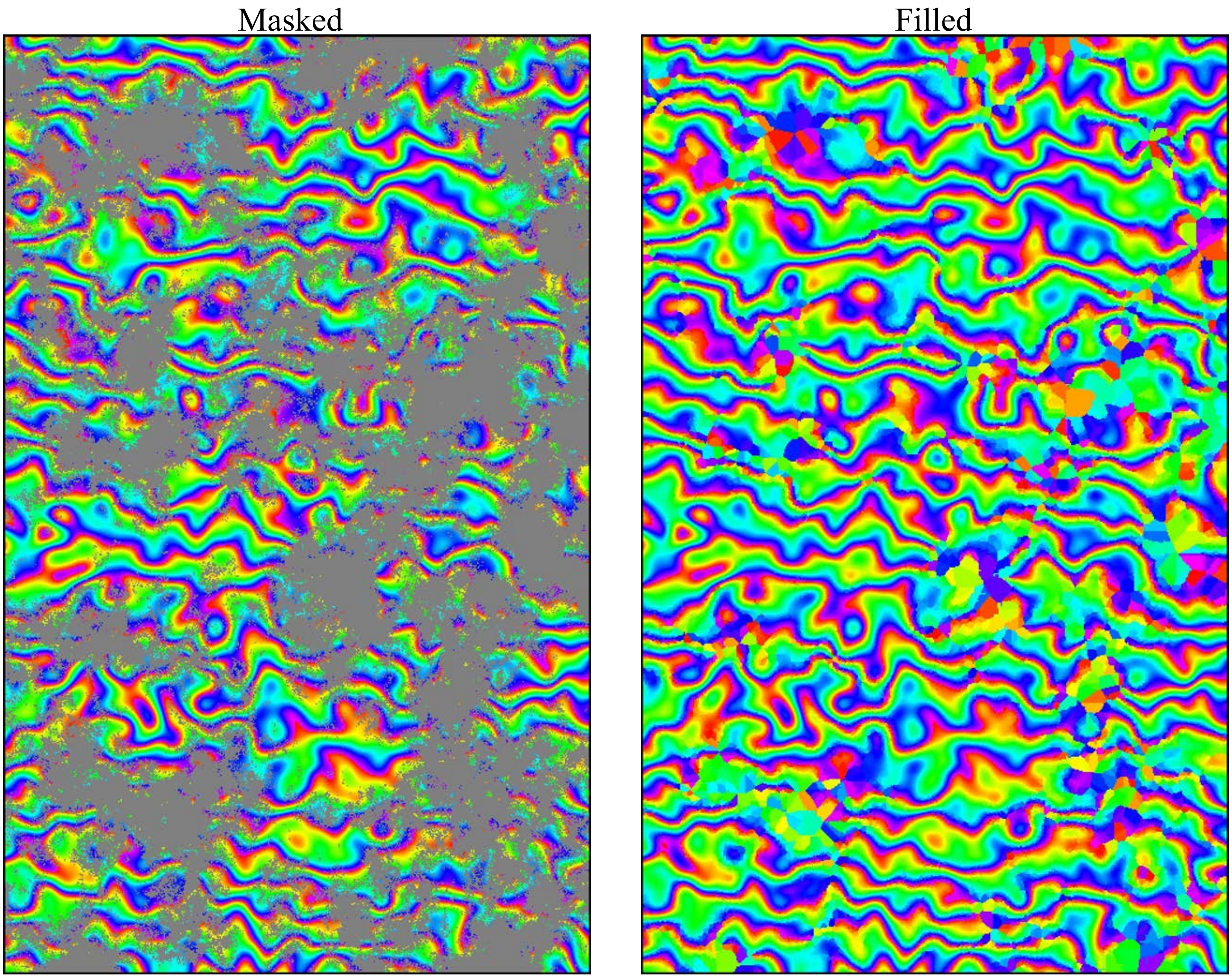
Implementation

```
# get x,y bounds
set minx = `grdinfo -C $in.grd |cut -f 2`
set maxx = `grdinfo -C $in.grd |cut -f 3`
set nx = `grdinfo -C $in.grd |cut -f 10`
set boundsx = "$minx $maxx"
set miny = `grdinfo -C $in.grd |cut -f 4`
set maxy = `grdinfo -C $in.grd |cut -f 5`
set ny = `grdinfo -C $in.grd |cut -f 11`
# for some reason we have to reverse these two
set boundsy = "$maxy $miny"

# first convert to ascii
grd2xyz $in.grd -S -V > $in.gmt

# run gdal, then convert back to grd
gdal_grid -of GTiff -txe $boundsx -tye $boundsy -outsize $nx $ny -
l $in -a nearest $in.gmt $out.tiff
gdal_translate -of GMT -ot Float32 $out.tiff $out.grd

# fix the grd header metadata
grdedit $out.grd -T #(note: must be pixel node registration for
snaphu)
grdedit $out.grd -R$minx/$maxx/$miny/$maxy
```

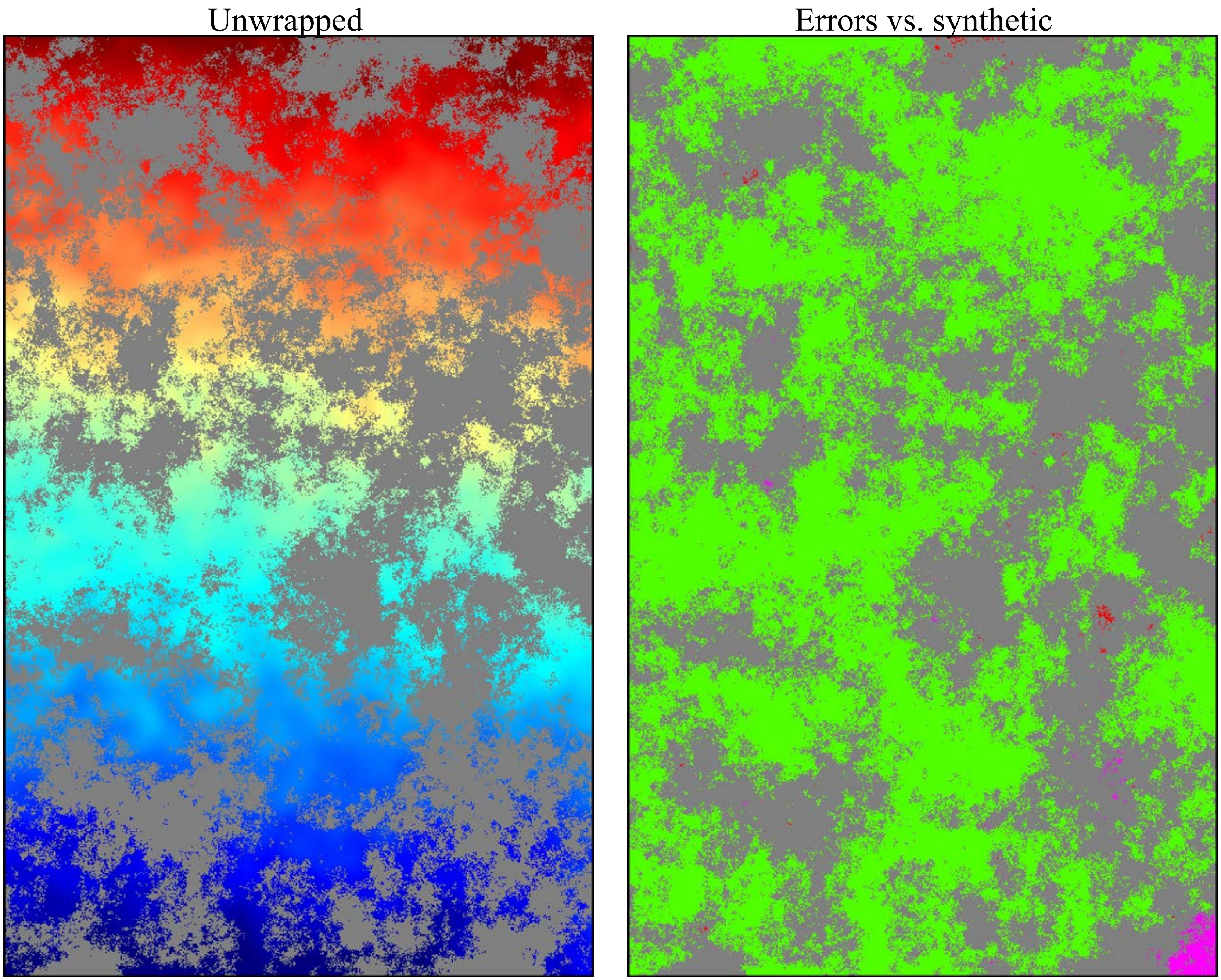
Synthetic example



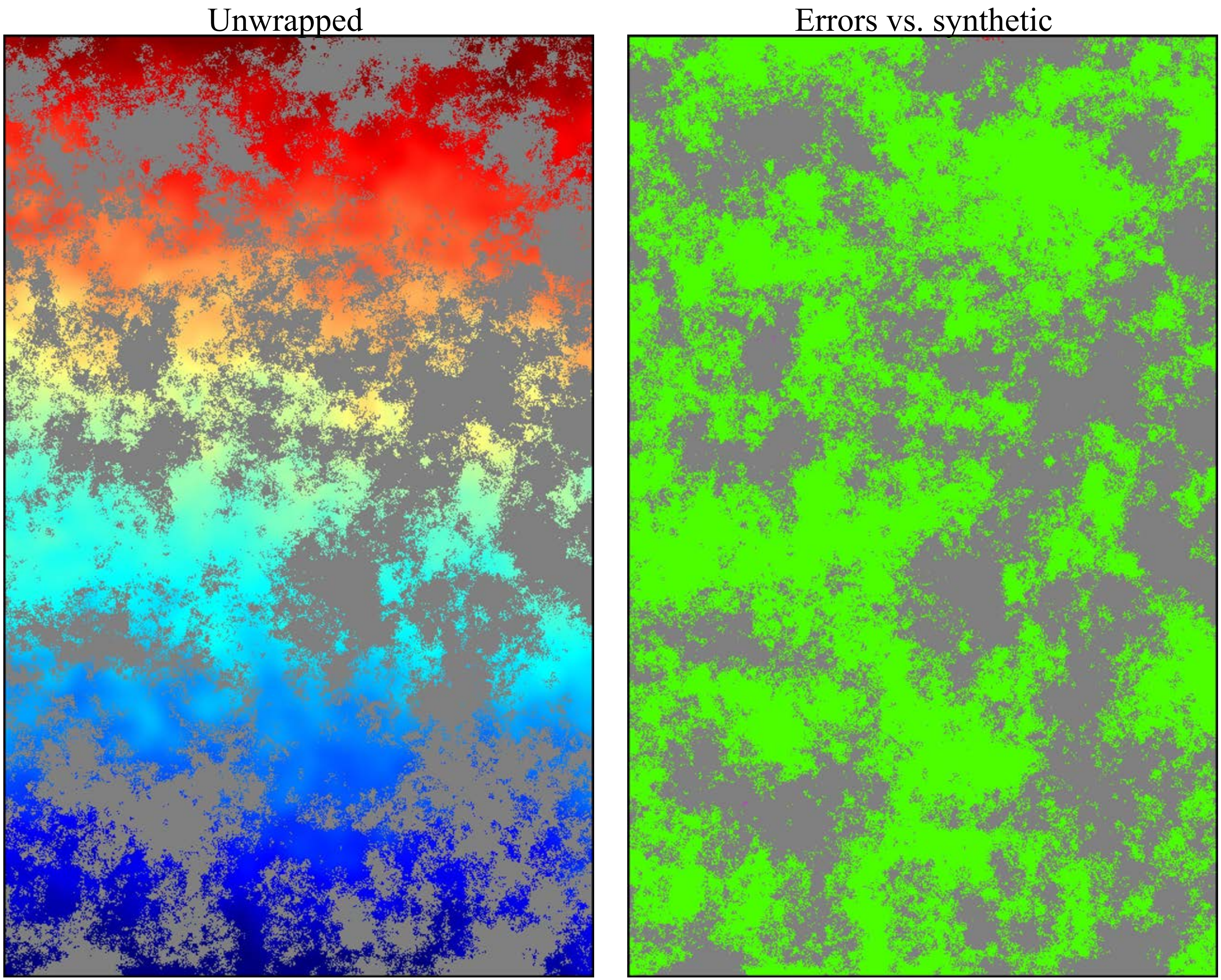
Mask, then interpolate



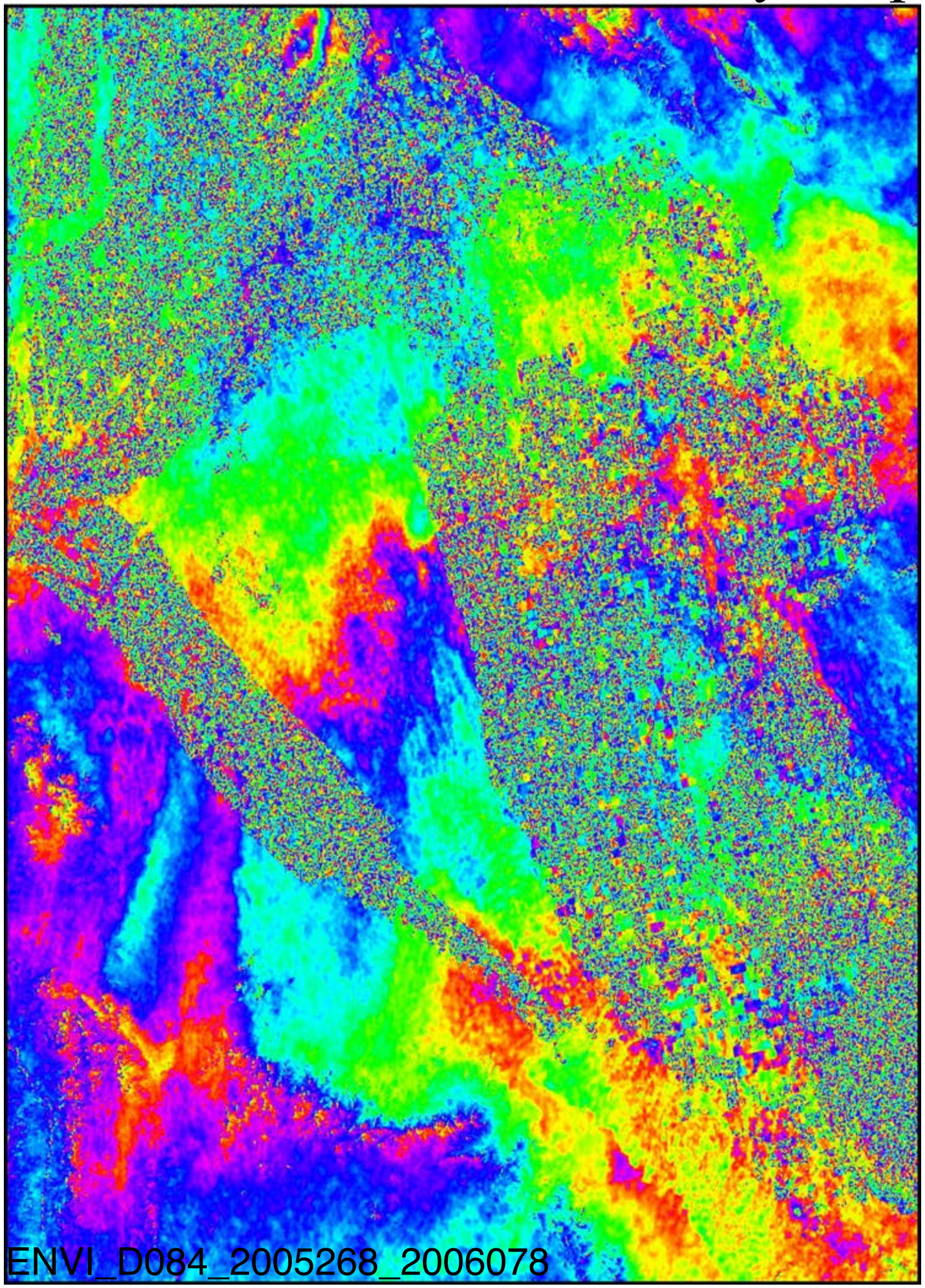
Basic unwrapping: 5.5 sec

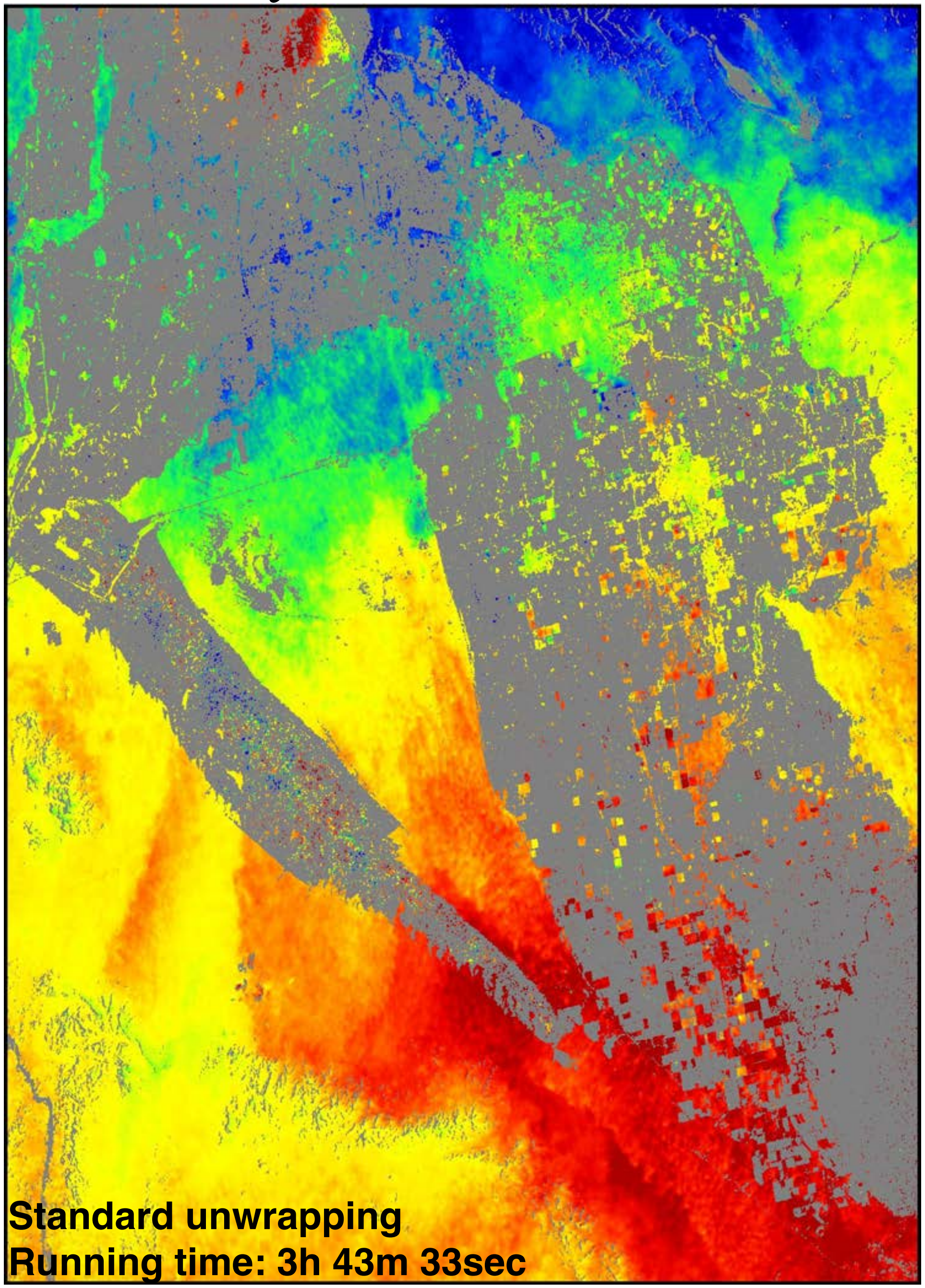


Interpolation-assisted: 0.9 sec (6x speedup!)

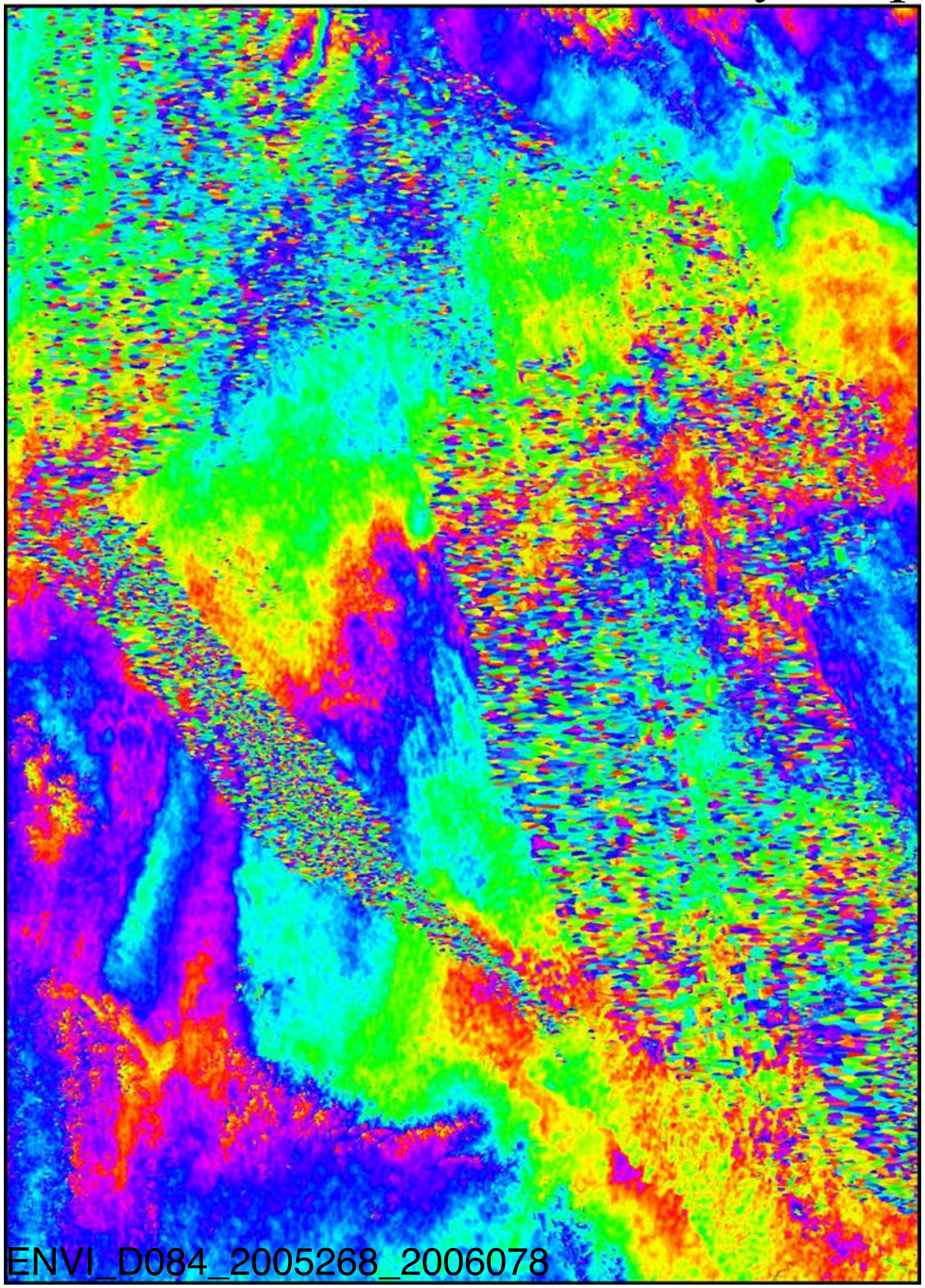


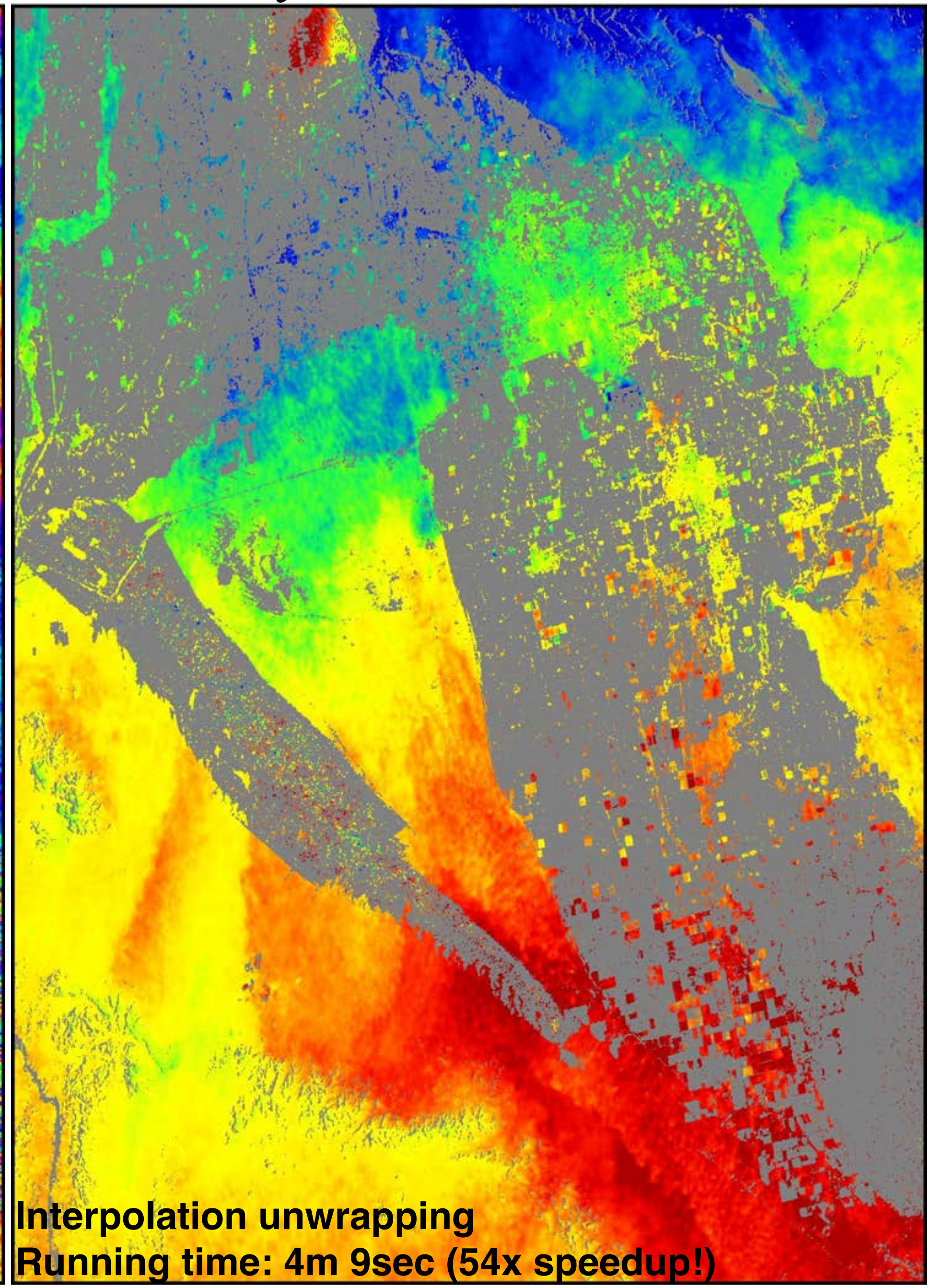
Case study: Imperial Valley, CA



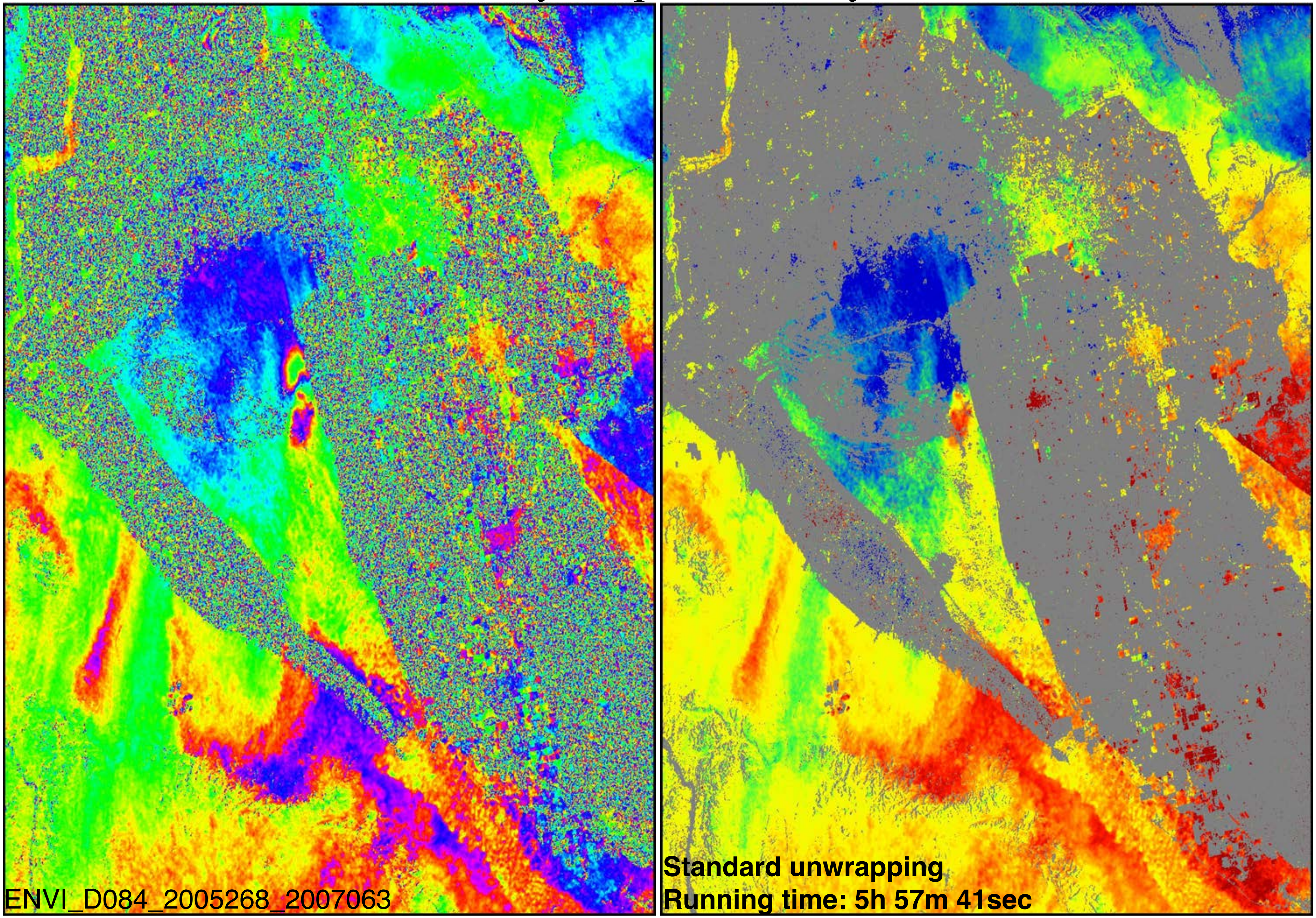


Case study: Imperial Valley, CA

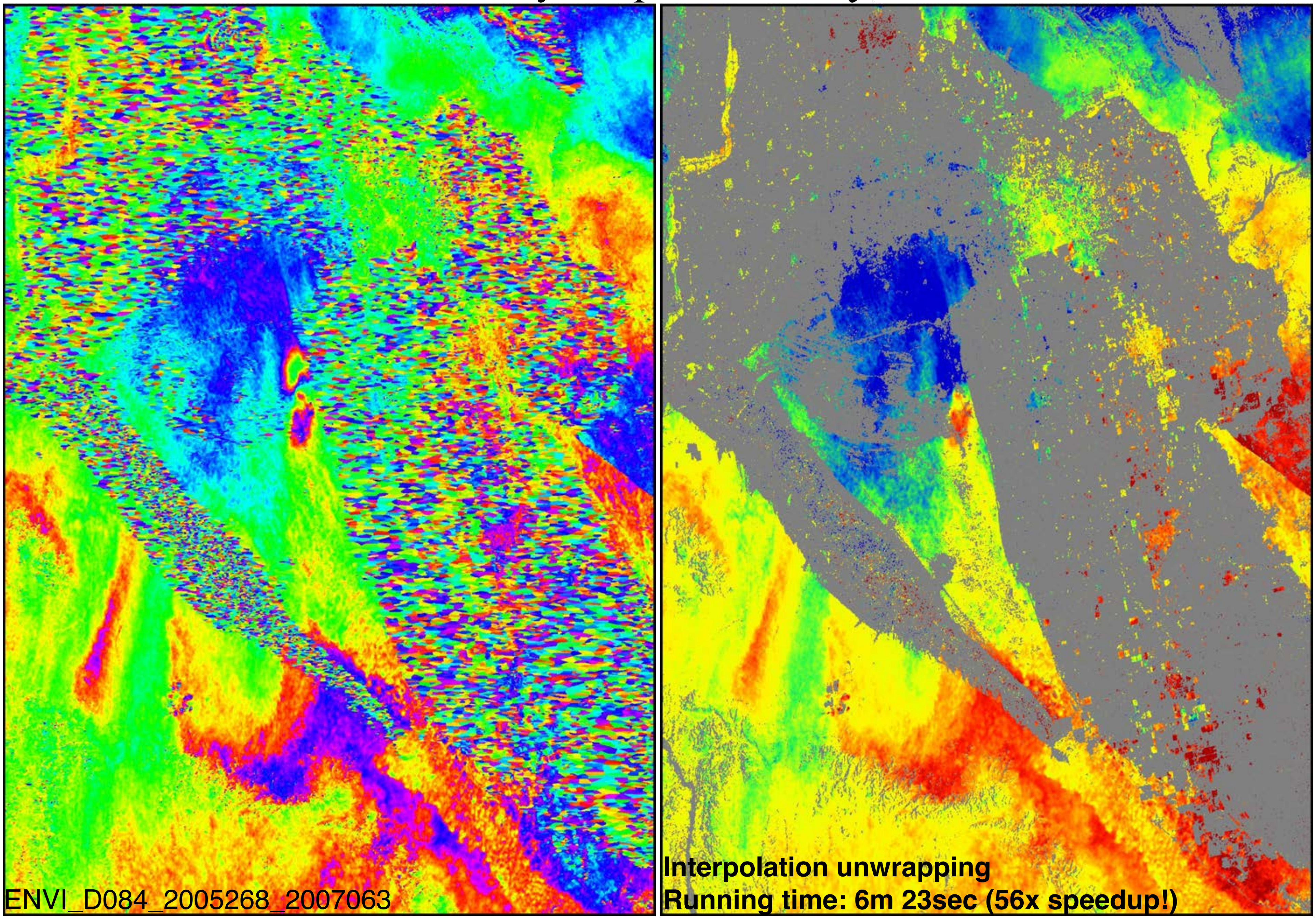




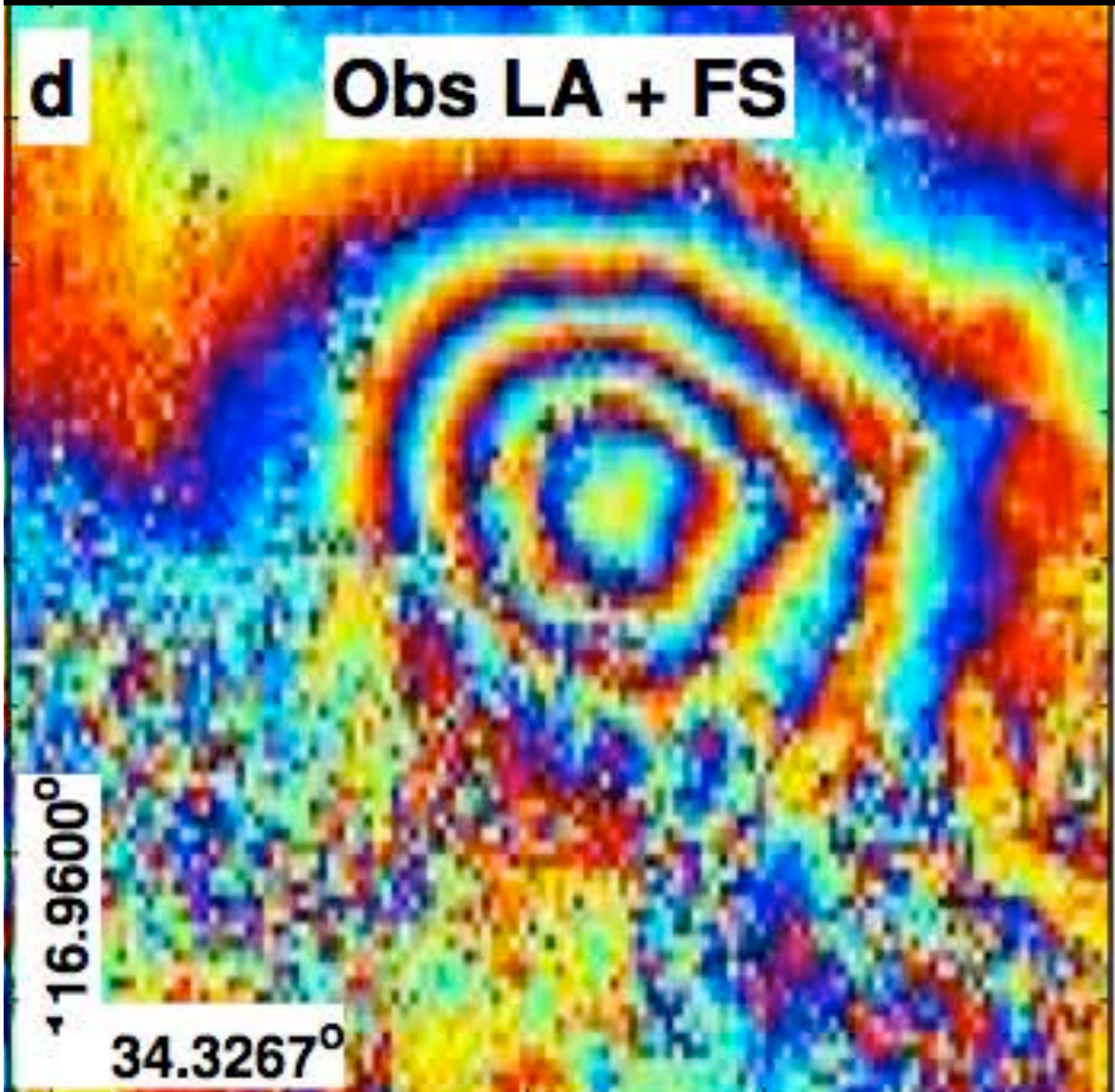
Case study: Imperial Valley, CA

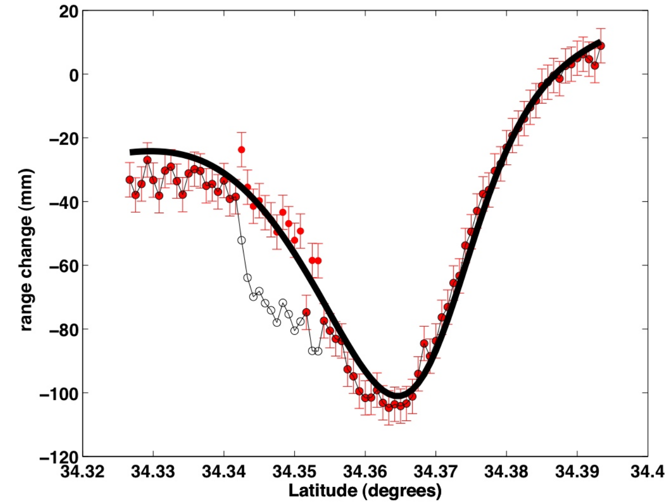


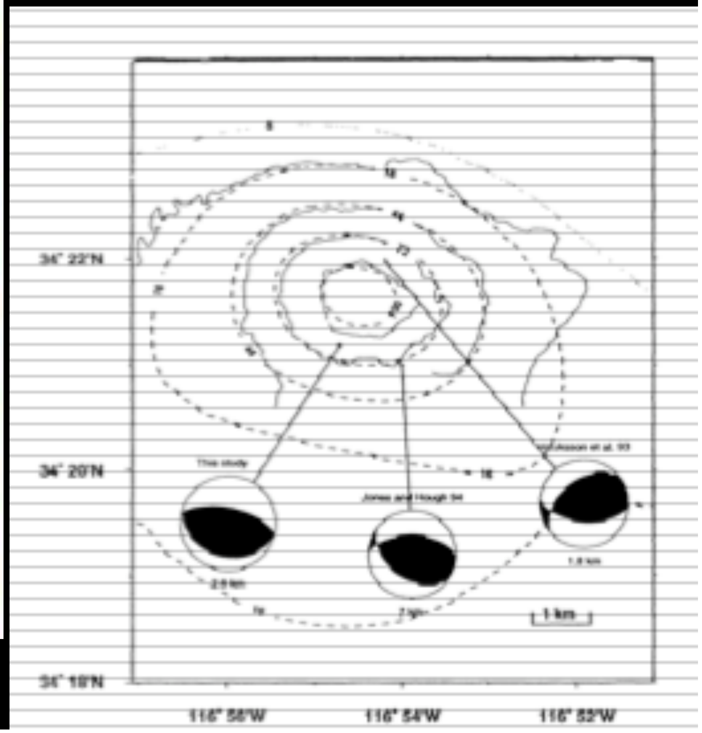
Case study: Imperial Valley, CA



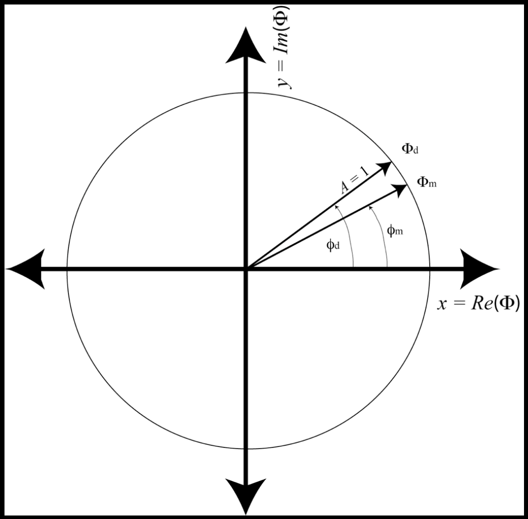
Interval 1992.3 to 1993.5 includes quake on 1992.9







GIPhT = General Inversion of Phase Technique



do not unwrap

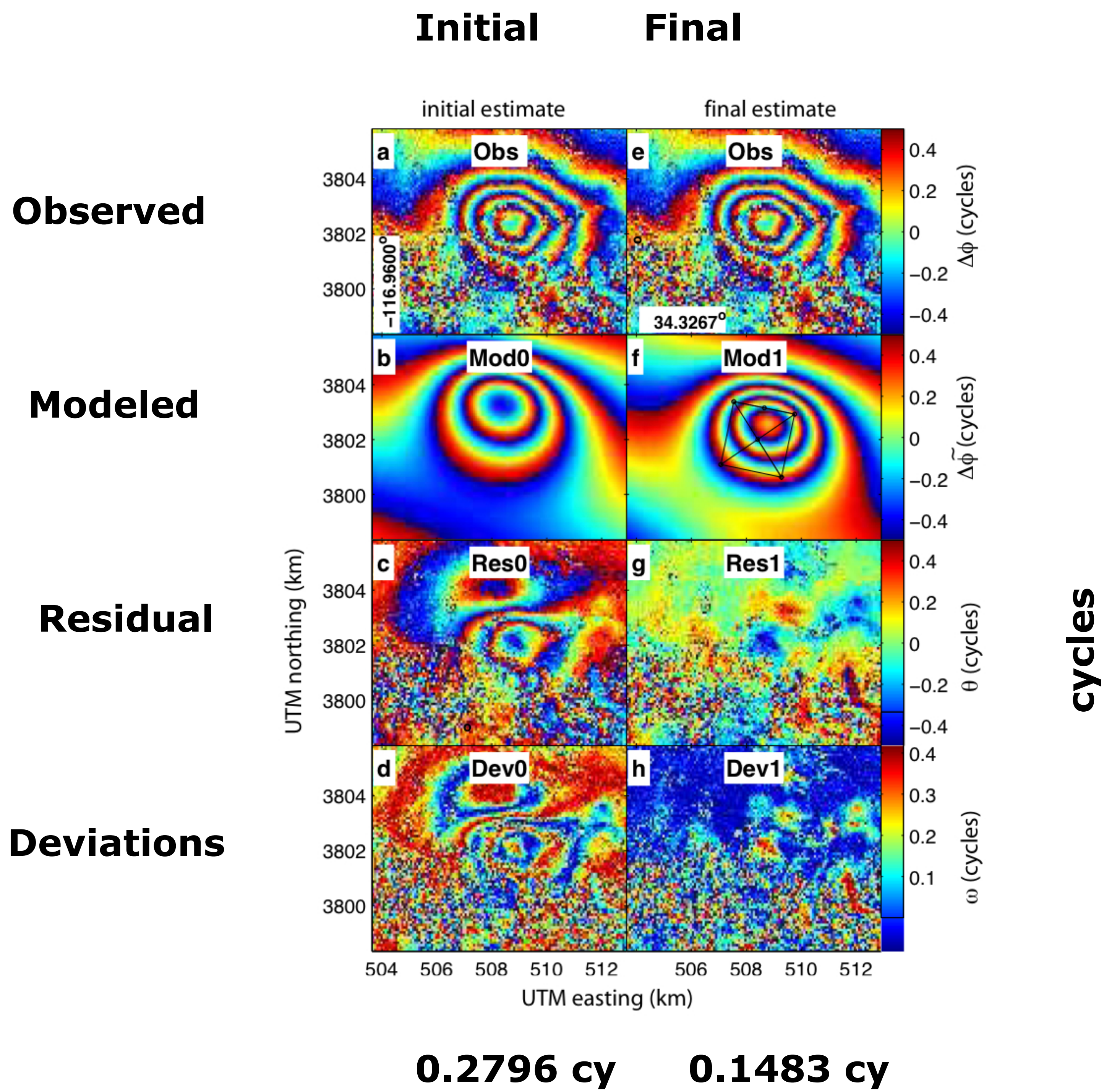
"A technique for modeling radar interferograms without phase unwrapping: Application to the M 5 Fawnskin, California earthquake of 4 December 1992." *Geophys. J. Int.* [2009]

- Select pixels
- Model phase without unwrapping
- Misfit is average angular cost
- Minimize with simulated annealing

Advantages:

- Avoid unwrapping errors
- Statistics: residuals are distributed as Von Mises
- Allows general time- and space- dependence
- Evaluate goodness of fit
- Evaluate uncertainty of parameter estimates
- Open source lesser GPL
- https://github.com/feigl/gipht
- Kurt Feigl (feigl@wisc.edu)

Example



Initial
Final
Observed
Modeled
Residual
Deviations
initial estimate
final estimate
a Obs
e Obs
b Mod0
f Mod1
c Res0
g Res1
d Dev0
h Dev1
-116.9600°
34.3267°
3804
3802
3800
UTM northing (km)
504 506 508 510 512
506 508 510 512
UTM easting (km)
Δϕ (cycles)
θ (cycles)
ω (cycles)
0.4 0.2 0 −0.2 −0.4
0.4 0.3 0.2 0.1
cycles
0.2796 cy
0.1483 cy

Phase gradient

$$\psi = \frac{\partial \phi}{\partial x}$$

$$\psi \doteq \frac{\Delta \phi}{\Delta x}$$

$$\psi \doteq \frac{\phi^{(k+1)} - \phi^{(k)}}{x^{(k+1)} - x^{(k)}}$$

do not unwrap

Sandwell, D. T., and E. J. Price (1998), Phase gradient approach to stacking interferograms, *Journal of Geophysical Research: Solid Earth, 103*, 30183-30204. http://dx.doi.org/10.1029/1998JB900008

Ali, S. T., and K. L. Feigl (2012), A new strategy for estimating geophysical parameters from InSAR data: application to the Krafla central volcano, Iceland, *Geochemistry, Geophysics, Geosystems, 13*.

http://dx.doi.org/10.1029/2012GC004112

- Quad-tree resampling
- estimate 3 parameters per patch
- Misfit is circular mean deviation

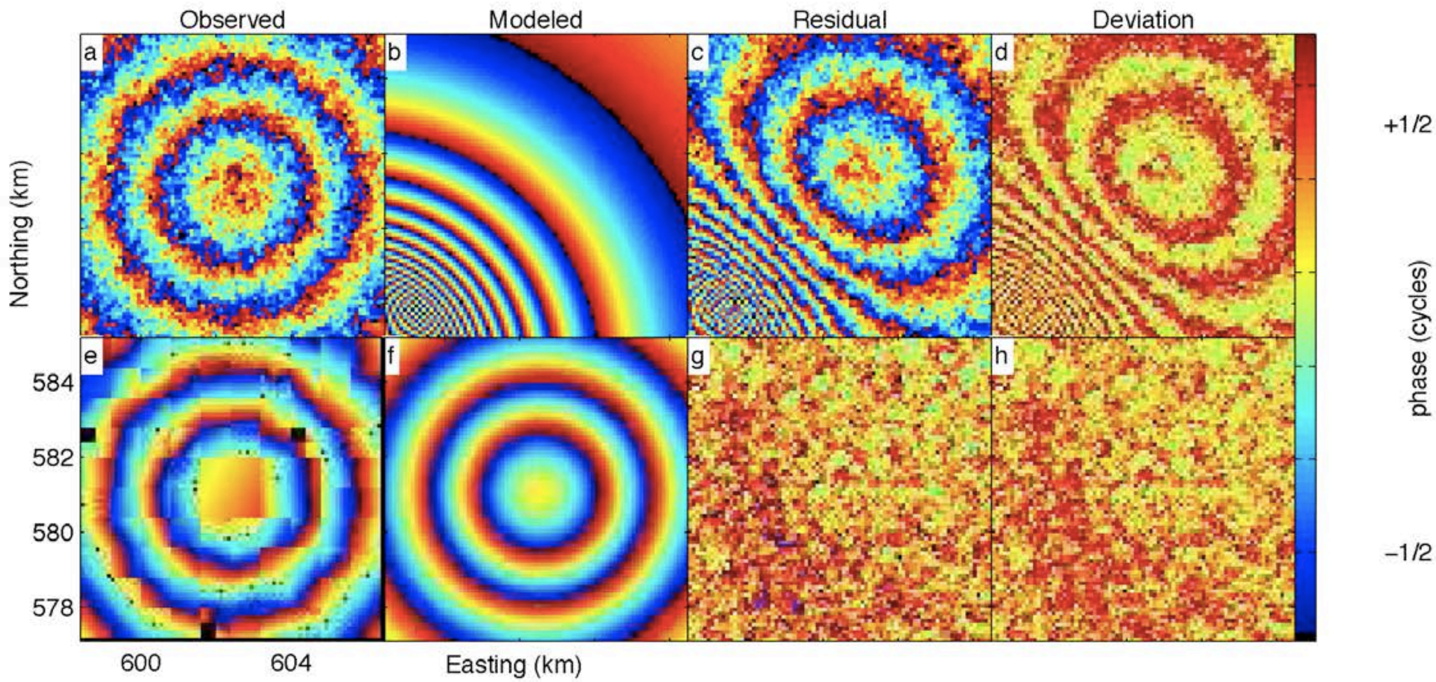
Advantages:

- Avoid unwrapping errors
- quad-tree breaks spatial correlation
- phase gradient is like a strain (e.g. Hooke’s Law
- known covariance
- Open source lesser GPL
- https://github.com/feigl/gipht
- Kurt Feigl (feigl@wisc.edu)

wrapped phase

initial

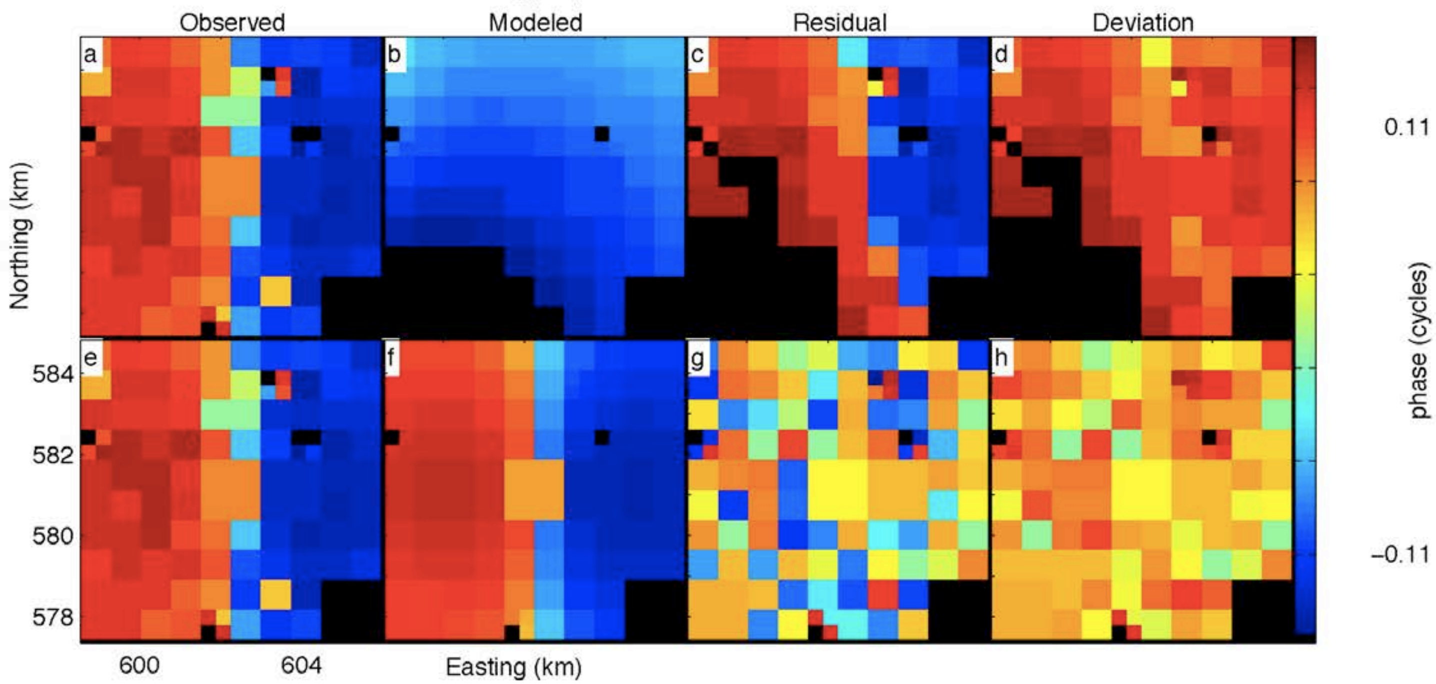
final



eastward gradient

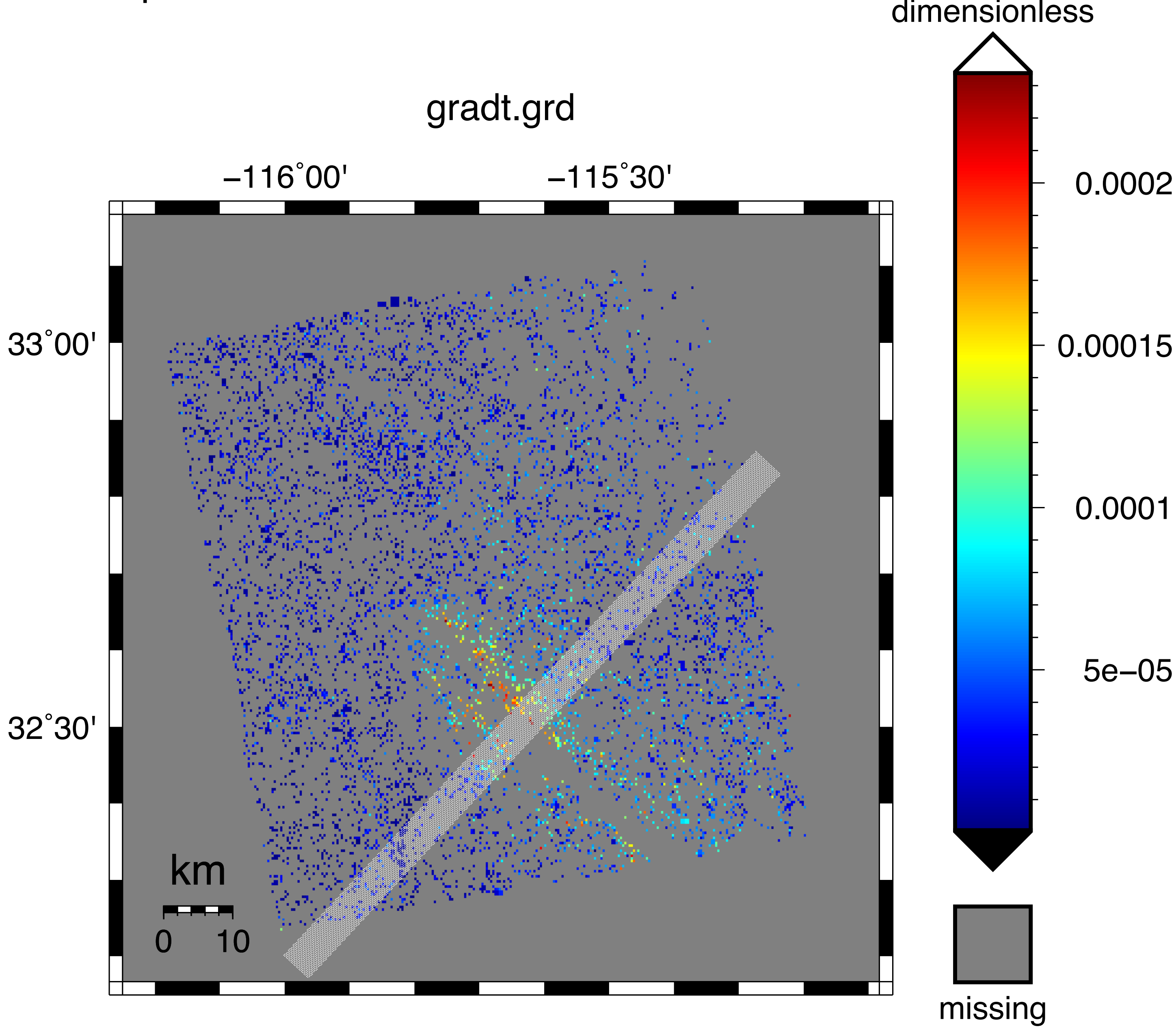
initial

final

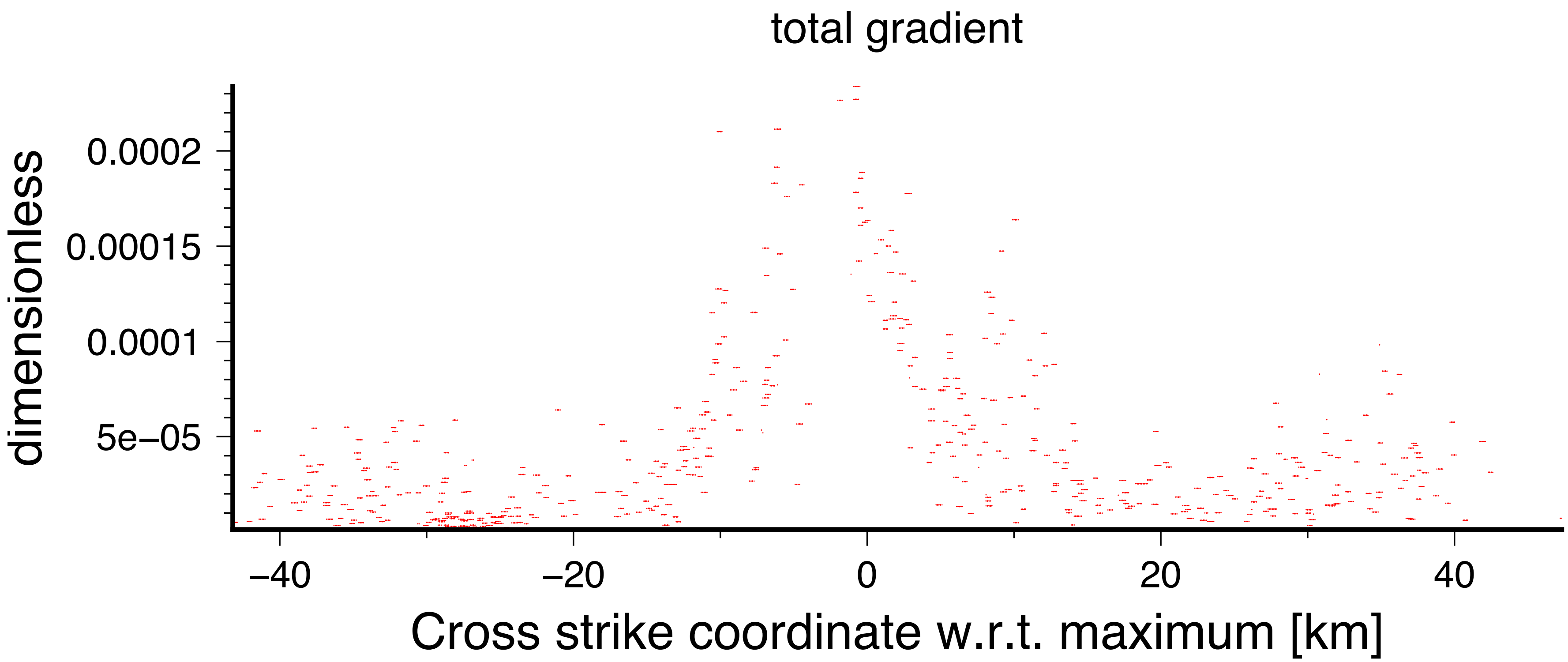


S. Tabrez Ali and Kurt Feigl (2012, G-cubed), http://dx.doi.org/10.1029/2012GC004112

Baja Earthquake

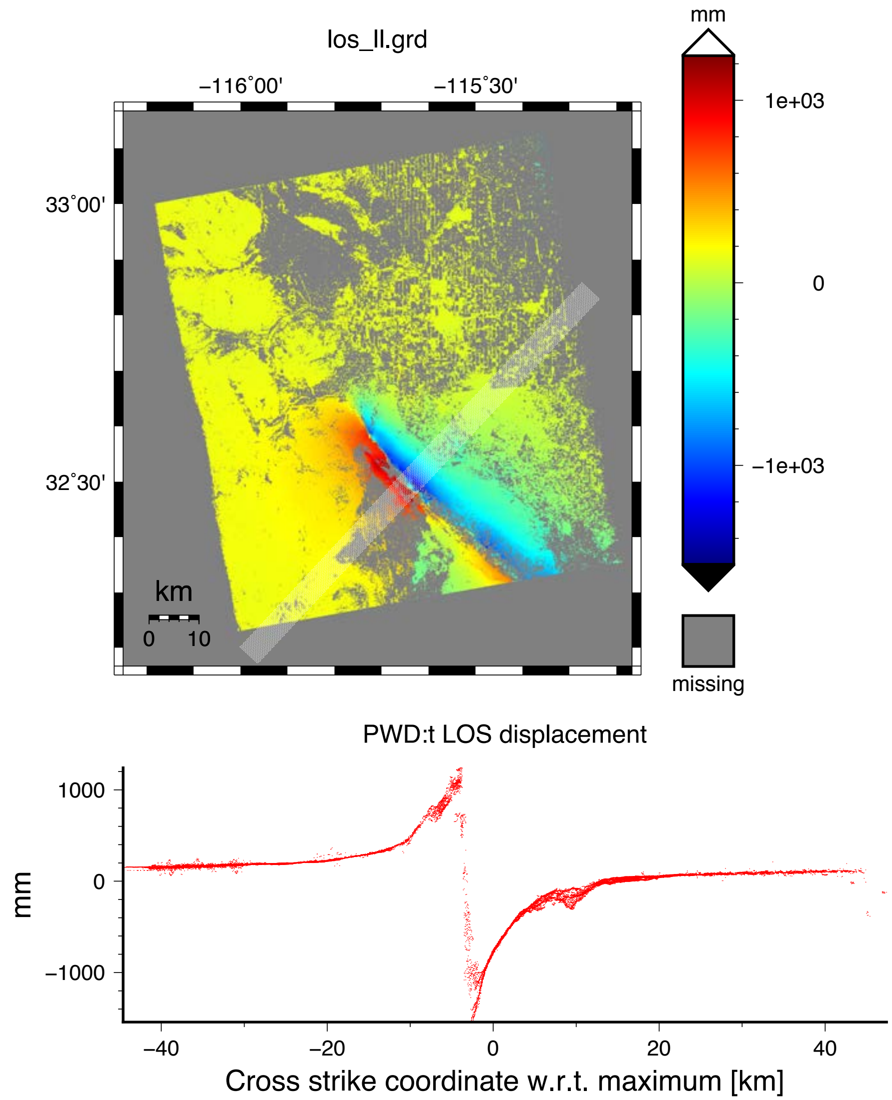


Baja Earthquake

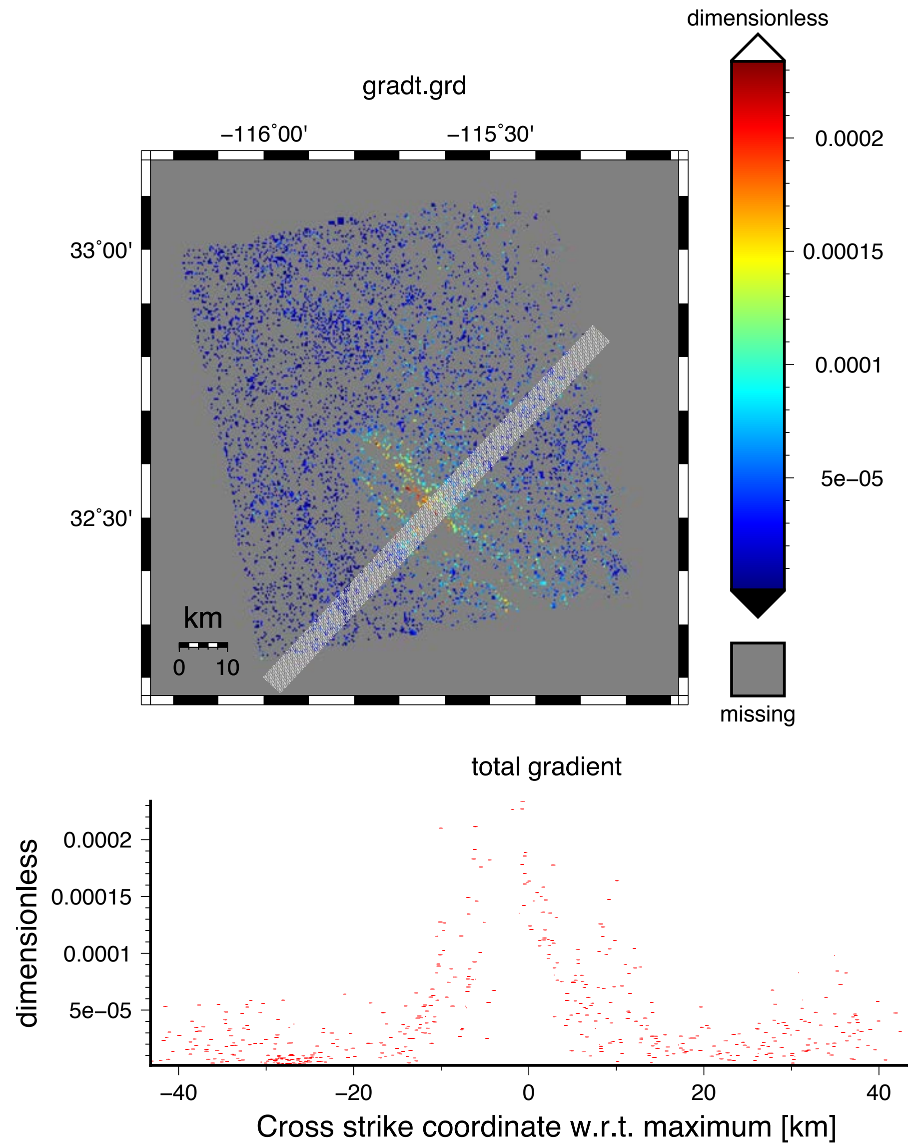


Baja Earthquake

LOS displacement (mm)



range gradient (dimensionless)



Northern Volcanic Zone, Iceland

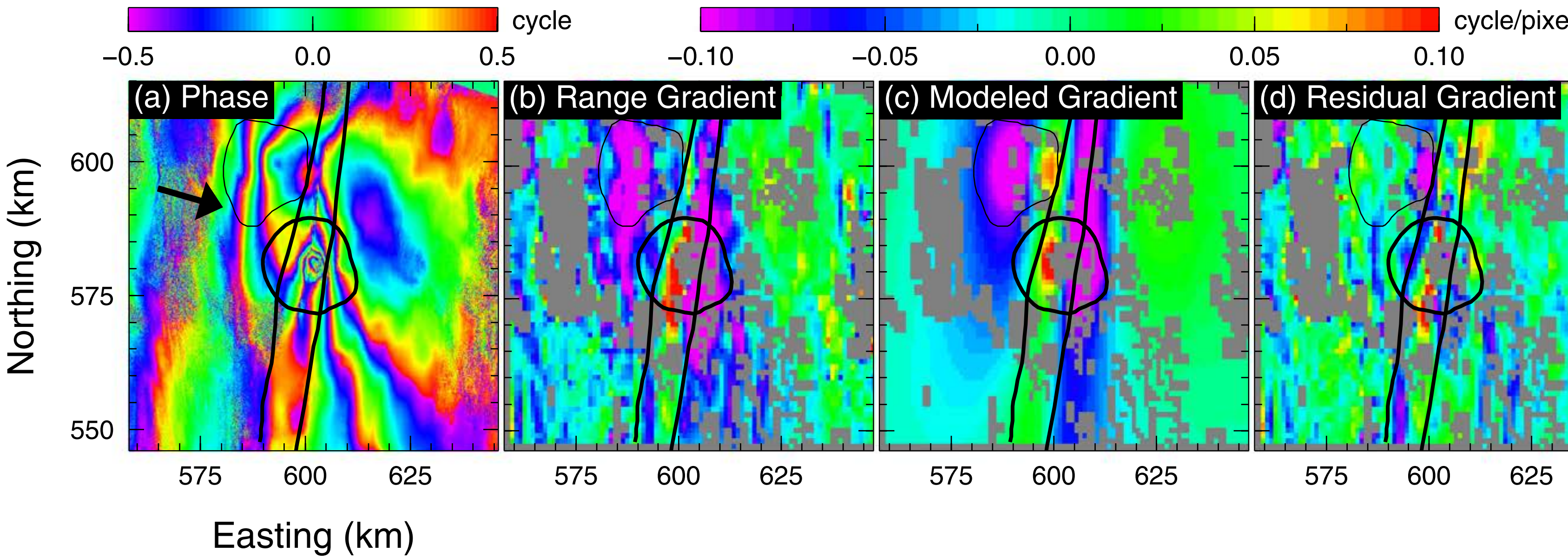
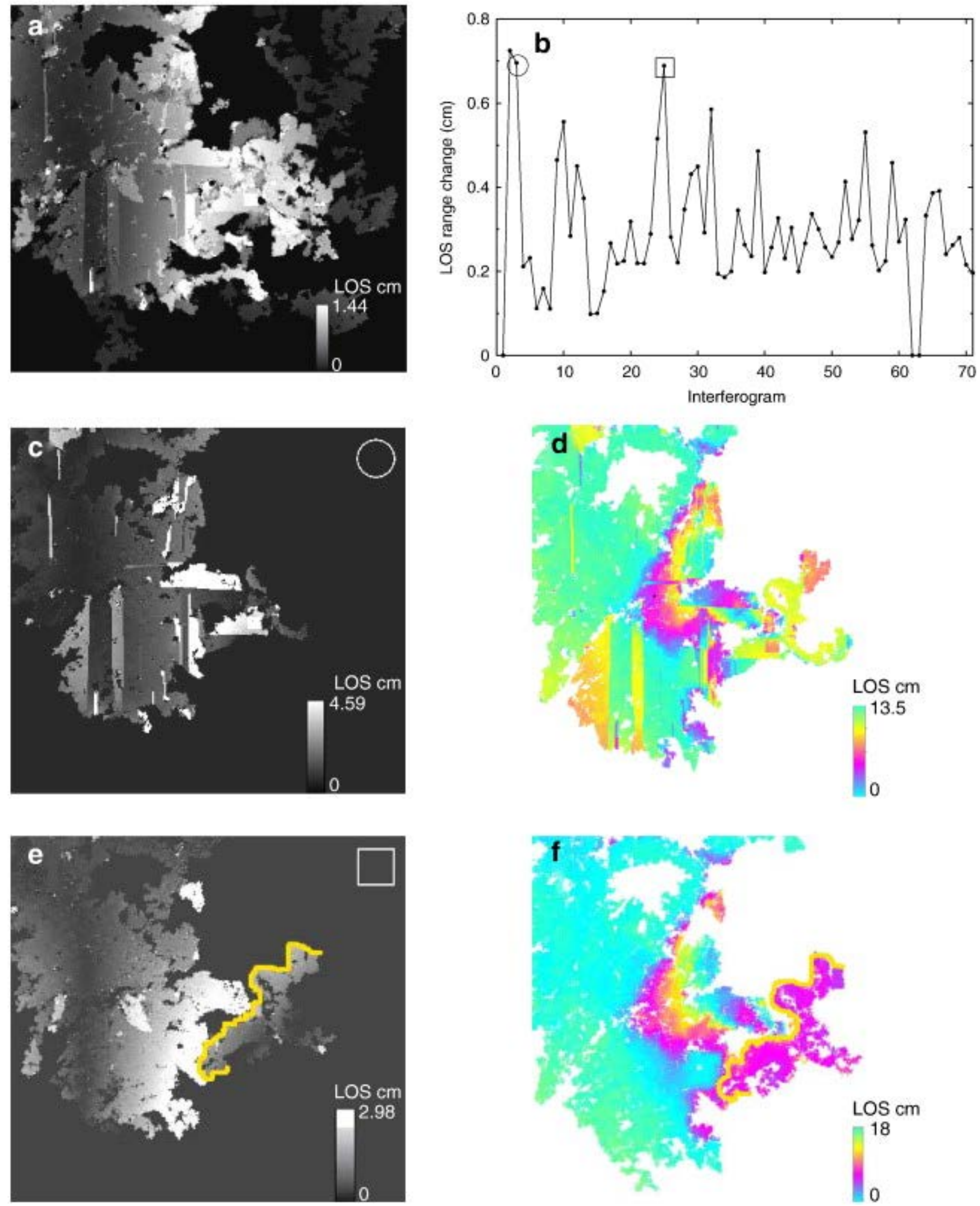


Figure 3. (a) Interferogram showing observed values of wrapped phase change $\Delta\phi$ for pair 1, spanning the time interval 1993.48–1998.63. One coloured fringe corresponds to one cycle of phase change, or 28 mm of range change. Black arrow shows the projection of the look vector (from sensor to target) onto the horizontal surface. (b) Observed values of the range gradient ψ in 28-mm cycles per 100-m pixel such that 0.05 cycles pixel^{-1} corresponds to 1.4×10^{-4} or 140 microstrain after resampling by the quad-tree algorithm. (c) Modelled values of the range change gradient ψ' calculated from the final estimate of the parameters in the model for this individual pair. (d) Residual range gradient formed by subtracting the modelled values from the observed values. Gray areas have been excluded by the quadtree resampling procedure. All coordinates are in easting and northing in kilometers in the ISN93 Lambert projection (Rennen 2002).

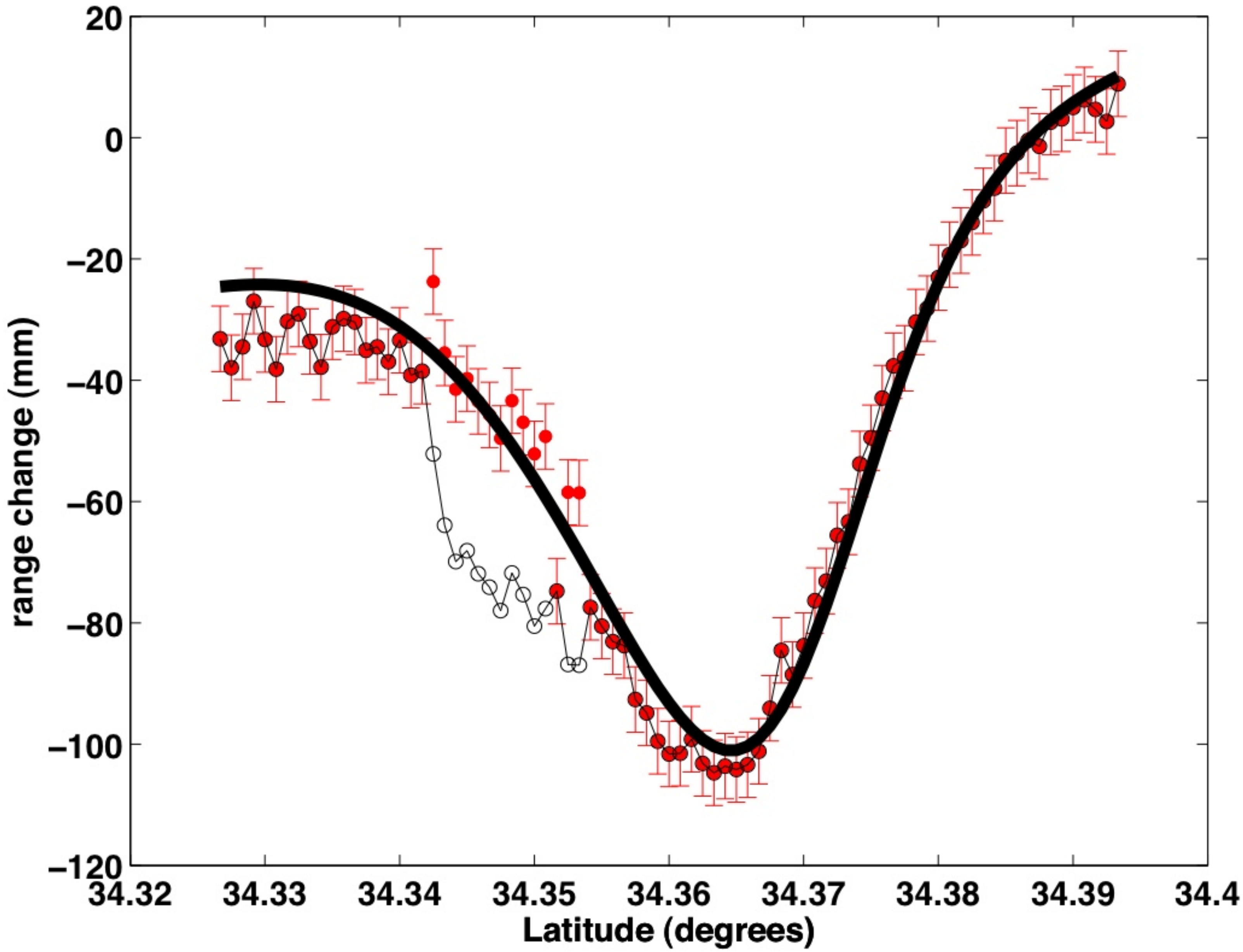
Ali, S. T., K. L. Feigl, B. B. Carr, T. Masterlark, and F. Sigmundsson (2014), Geodetic measurements and numerical models of rifting in Northern Iceland for 1993–2008, Geophys. J. Int., 196, 1267-1280. http://dx.doi.org/10.1093/gji/ggt462

How can we tell if unwrapping is working? Look at it



Lopez-Quiroz et al. [2009]

unwrap(observed) = model + wrap(residual)



Phase unwrapping

Assume that we have two signals taken at different times:

$g_1 = a_1 \exp(i4\pi R_1/\lambda)$

$g_2 = a_2 \exp(i4\pi R_2/\lambda)$

a1,a2 = complex reflectivity

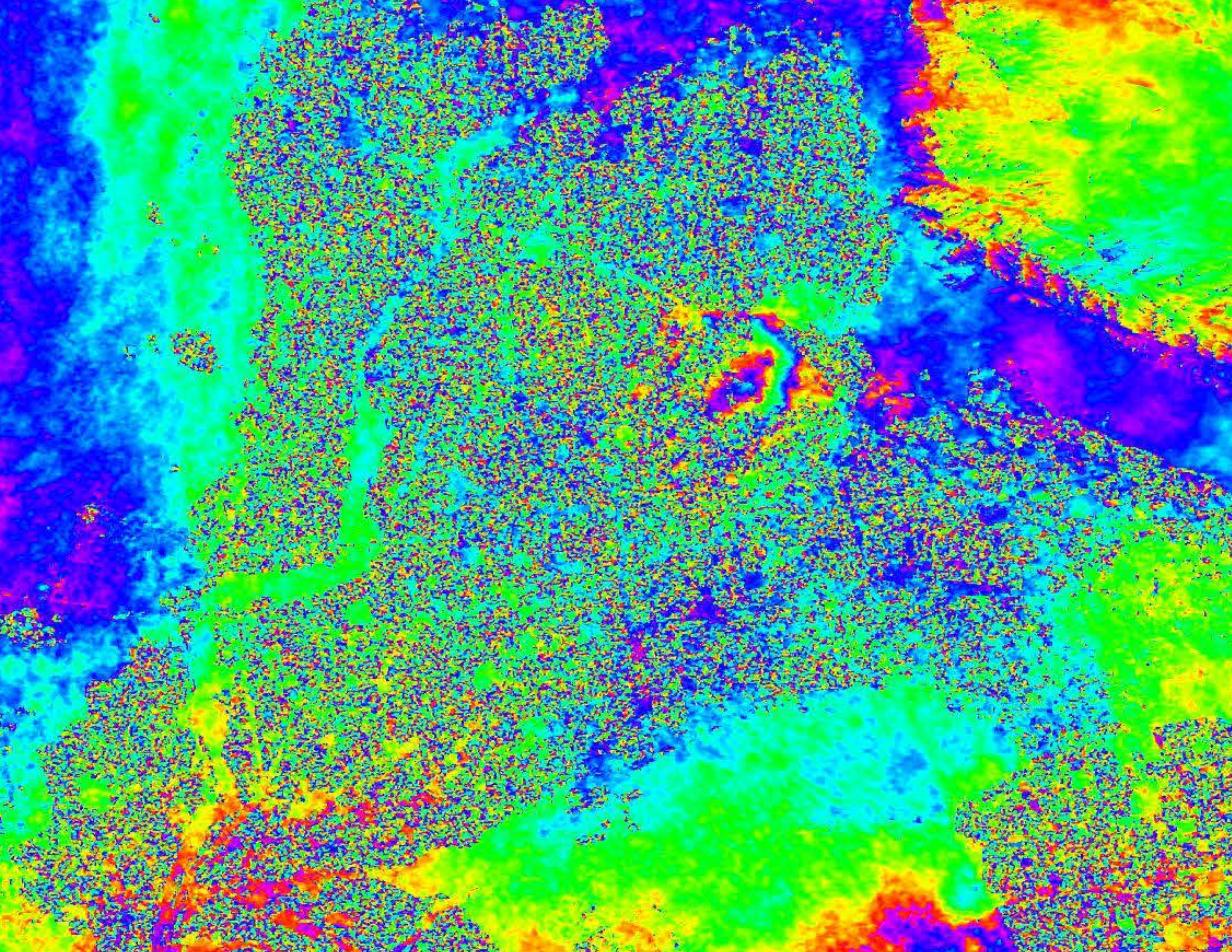
R_1, R_2 is range from antenna to surface

λ = wavelength

At a given point, assume $a_1 = a_2 = a$

$(g1)(g2^*) = (a^2)\exp[i4\pi(R_1 - R_2)] = s(t)$

The phase of this function is proportional to the effective difference in range, which in turn depends on satellite geometry, topography, soil moisture, or maybe even *deformation*.



How can we tell if it is working?

(1) Look at it

(2) Re-wrap: need GMT macro

(3) Permute master and slave:

if difference

unwrap(M-S) – unwrap(S-M)

does not equal zero, then suspect unwrapping errors

Exercise:

compare_unwrappings.csh

https://uwmadison.box.com/s/fzlyset9ixw5kxc1ksa2e393jmohu5j8

```
# These four files should go into your path somewhere
histogram.csh   #
given a .grd file, calculate a histogram and statistics

plot_grd_ll.csh #
given a .grd file in latitude, longitude (ll) coordinates,
plot a map and a profile

plot_grd_ra.csh #
given a .grd file in range, azimuth (ra) coordinates,
plot an image and a profile

make_profile.csh #
make a profile (called by plot_grd_ll.csh)

# This one is specialized
grdmath.macros  #
macros for GMT5 grdmath for wrapping phase
(place in your home directory called ~/.gmt
```

GMT5ShortCourse2015/ALOS_Baja_EQ
config.alos.txt

```
# VARIATION NUMBER 1: change defo_max
# Allow phase discontinuity in unwrapped phase. This is needed for interferograms
having sharp phase jumps.
# defo_max = 0 - used for smooth unwrapped phase such as interseismic deformation
# defo_max = 65 - will allow a phase jump of 65 cycles or 1.82 m of deformation at
C-band
# defo_max = 20 works best for Baja Earthquake
# defo_max =  5 works, but not so well for Baja Earthquake

# VARIATION NUMBER 2: permute master (M) and repeat (S)
# switch the master and slave when doing intf.
# put "1" if assume master as repeat and slave as reference
# put "0" if assume master as reference and slave as repeat [Default]
# phase = repeat phase - reference phase

# Goal: build the following table
#             20          05
#       ------------------------
#   MS  |           |           |
#       ------------------------
#   SM  |           |           |
#       ------------------------
```

compare_unwrappings.csh

```
# assign short names
set MS_20 = ./INTF_WITH_ALOS_DATA_defomax20/2009351_2010124/unwrap.grd
set MS_05 = ./INTF_WITH_ALOS_DATA_defomax5/2009351_2010124/unwrap.grd
set SM_20 = ./INTF_WITH_ALOS_DATA_flipped/2010124_2009351/unwrap.grd
set SM_05 = ./INTF_WITH_ALOS_DATA_flipped_defomax5/2010124_2009351/unwrap.grd

# compare
gmt grdmath $MS_20 $SM_20 ADD = MS_20vsSM_20.grd;
gmt grdmath $MS_20 $MS_05 SUB = MS_20vsMS_05.grd;
gmt grdmath $MS_05 $SM_05 ADD = MS_05vsSM_05.grd;
```

```
grdmath.macros

WRAP1 = STO@A 2 DIV PI DIV RINT RCL@A 2 DIV PI DIV EXCH SUB 2 MUL PI MUL
: usage A WRAP1 to return wrapped value of A in radians

WRAP2 = STO@B DIV STO@R RINT RCL@R EXCH SUB RCL@B MUL
: usage A B WRAP2 to return A wrapped on B

# re-wrap using WRAP1 macro
gmt grdmath unwrap.grd WRAP1 = rewrap.grd         ;
gmt grdedit -D//"radians"/1///"rewrapped phase"/”using WRAP1"  rewrap.grd

# re-wrap using WRAP2 macro
gmt grdmath unwrap.grd 2 PI MUL WRAP2 = rewrap_2pi.grd         ;
gmt grdedit -D//"radians"/1///"rewrapped phase”/”using WRAP2  rewrap_2pi.grd
```

Linear operator A $\quad A: E \rightarrow F$

where E and F are vector spaces over a field k
A is called a linear operator from E to F if

$$A(x+y) = A(x) + A(y)$$

$$A(\lambda x) = \lambda A(x)$$

for all x, y in R and λ in k. The simplest examples are:

the zero linear operator, which takes all vectors into 0
identity linear operator I , which leaves all vectors unchanged.

The wrap function is not a linear operator because

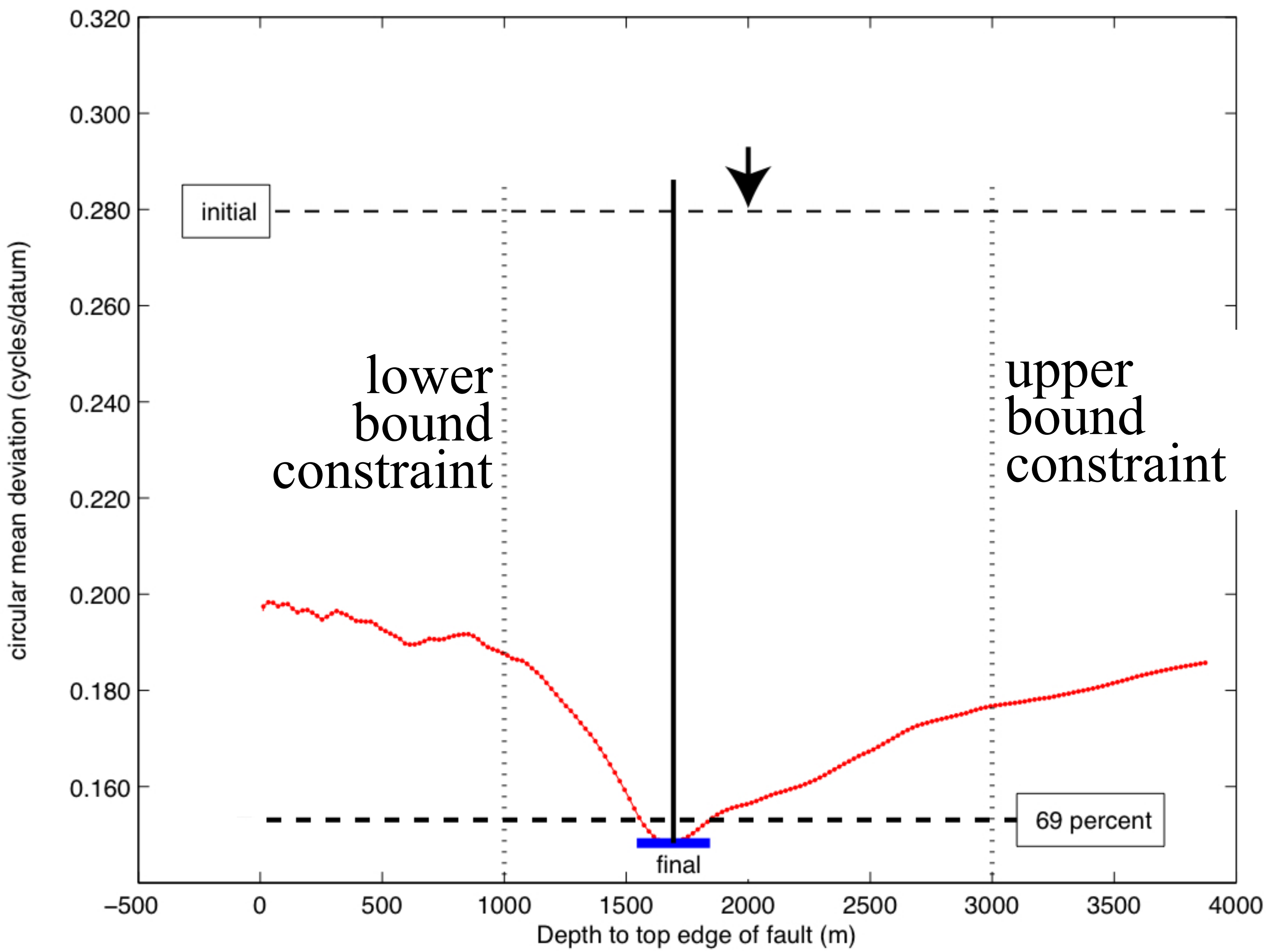
$$wrap(x+y) = wrap(x) + wrap(y)$$

only if

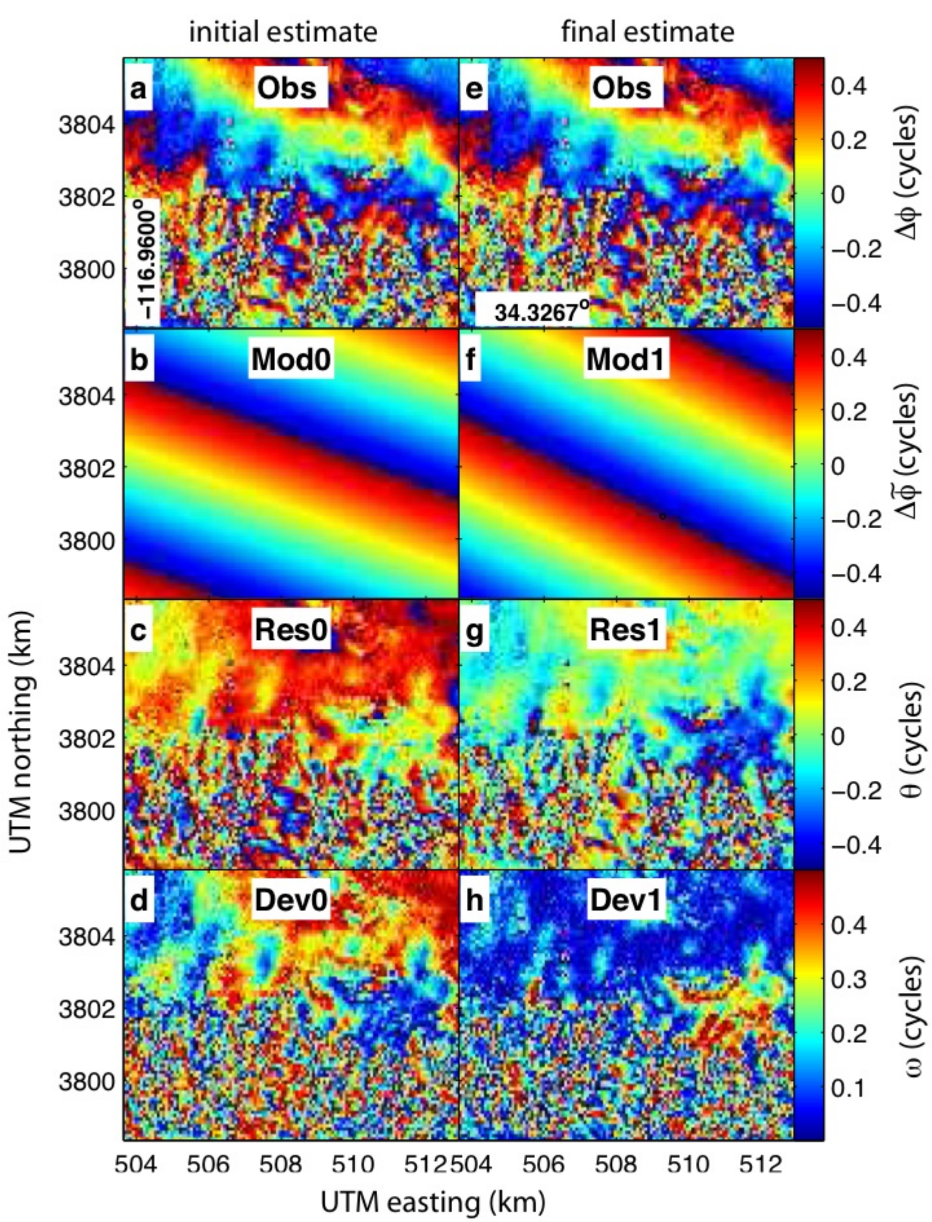
$$y = 2n\pi$$

$$wrap(\lambda x) \neq \lambda wrap(x)$$

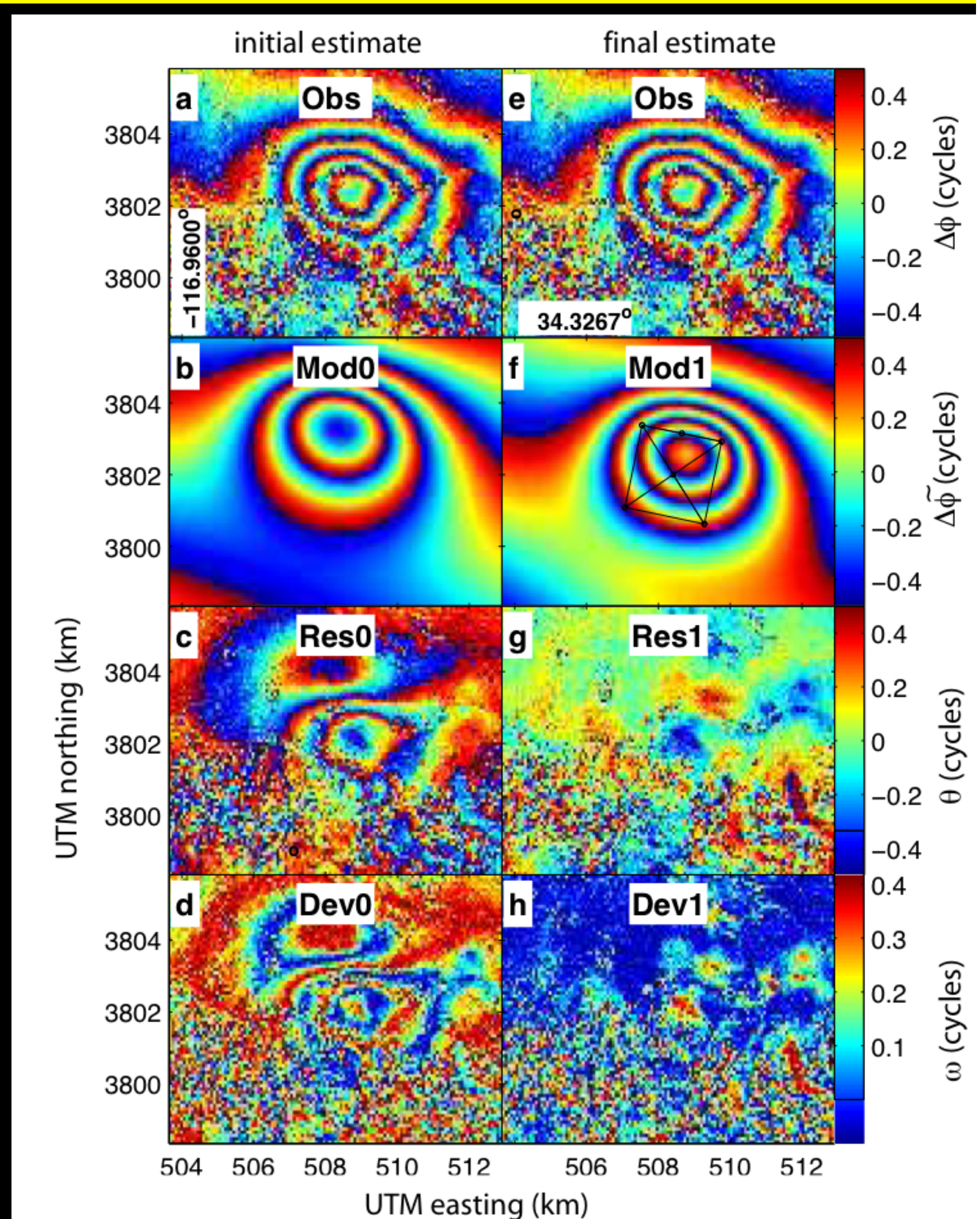
Fawnskin: 1 parameter: fault depth

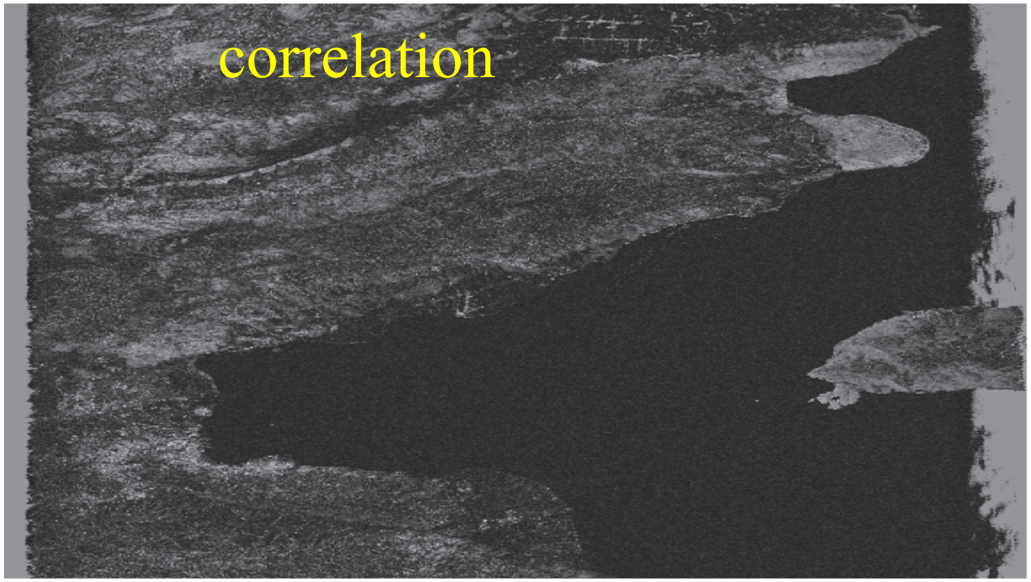


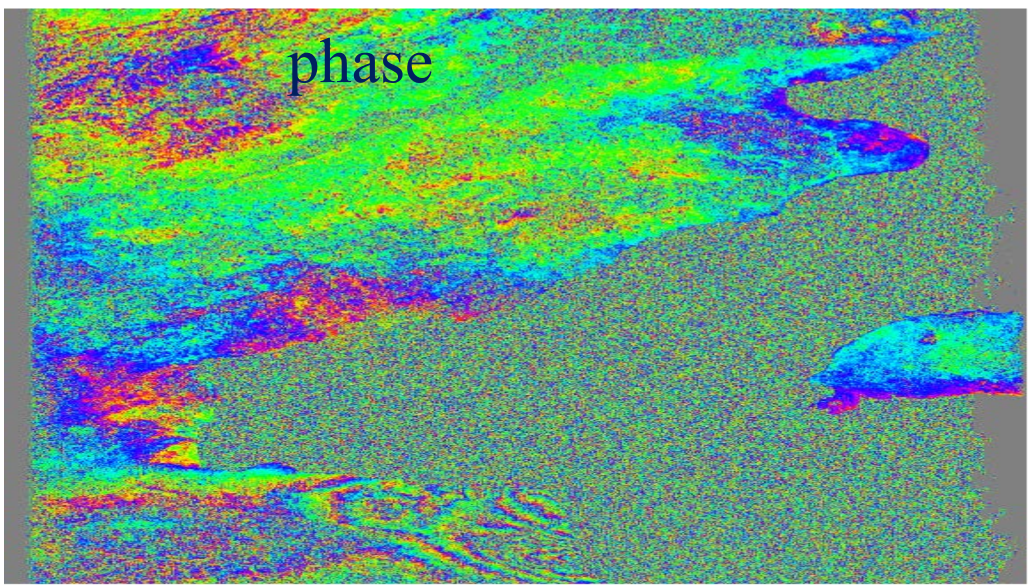
Pair 1 shows LS only



Pair 2 shows LS + FS







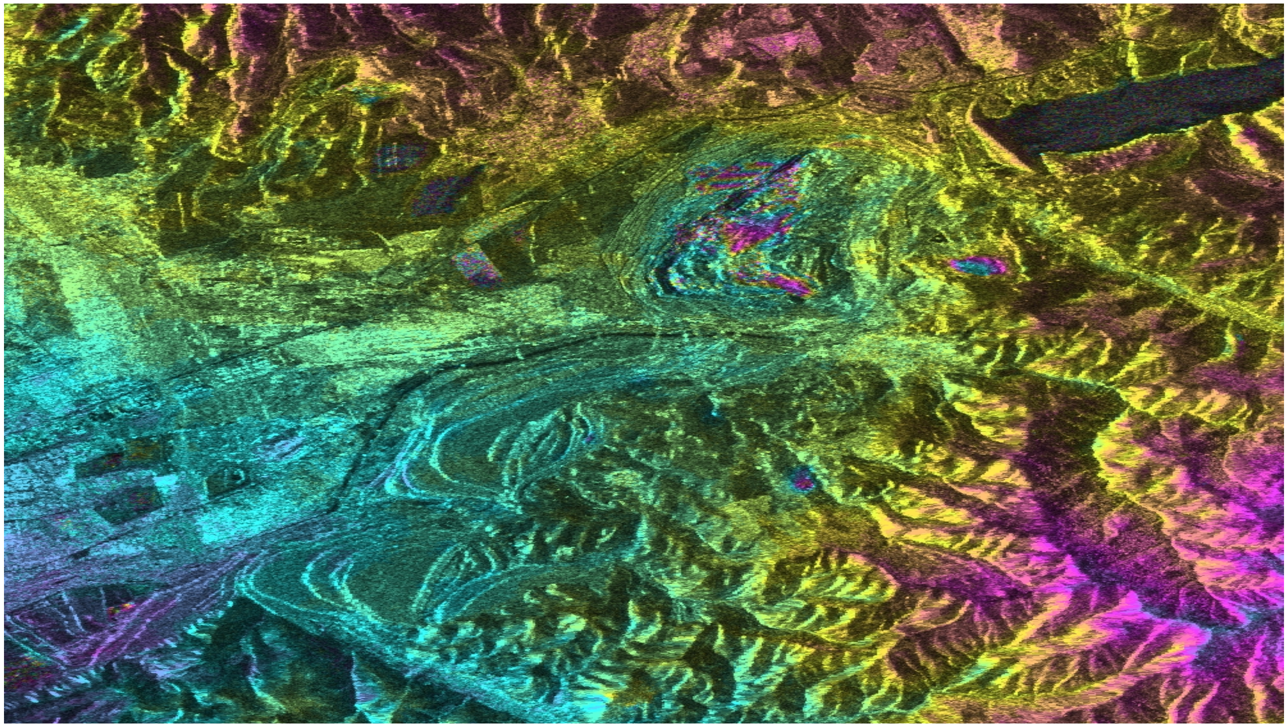
range

azimuth

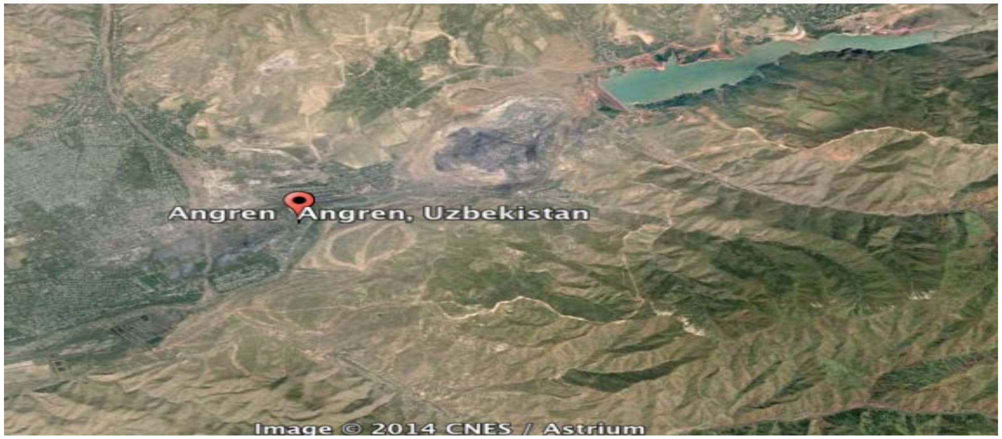
Haiti
ALOS L-band
(23 cm)
ascend
T447, F249

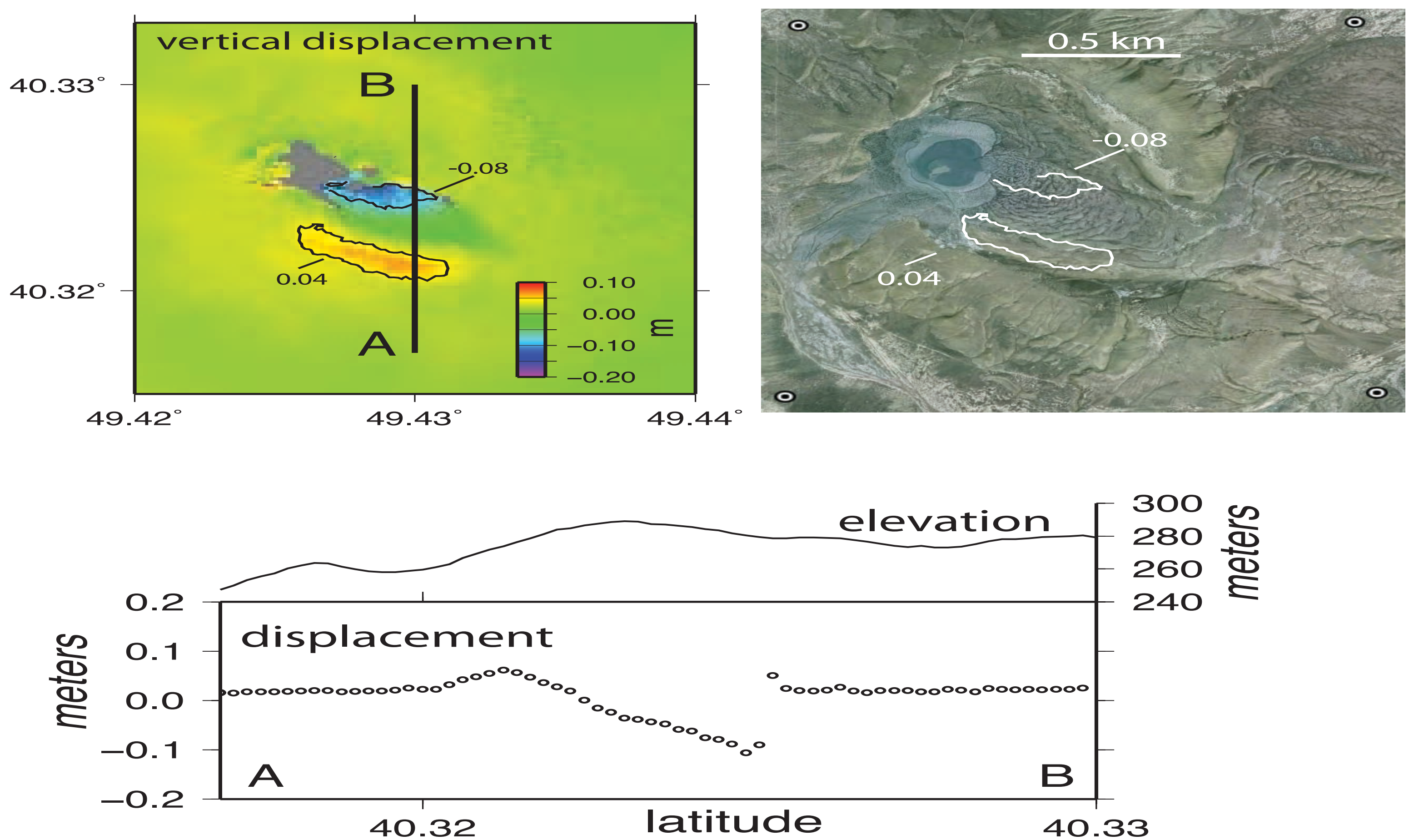
3/9/09-1/25/10

Bperp = 780
(gmtsar)

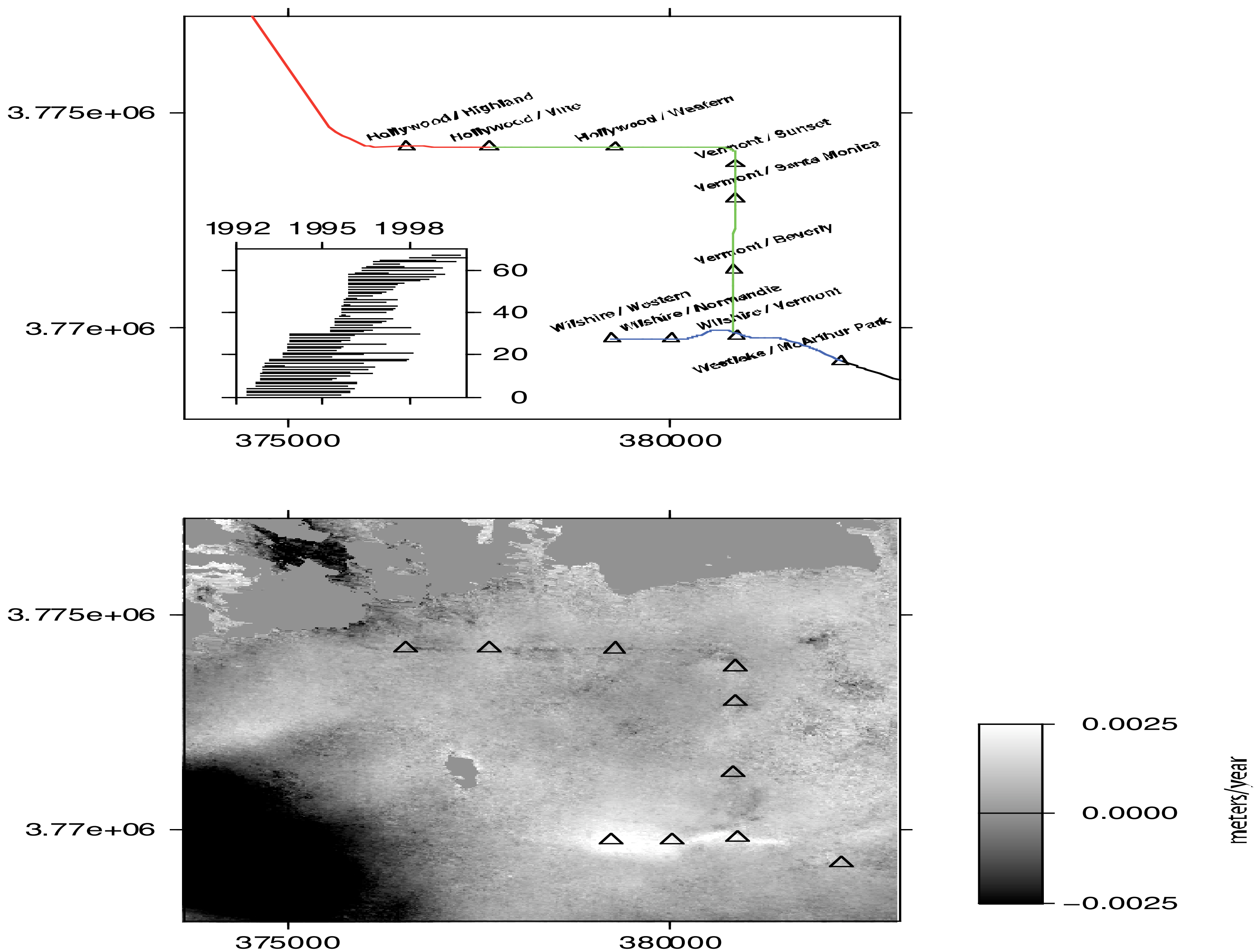


Coal mine in Uzbekistan





Mud volcano in Azerbaijan



Tunnels of LA

Overview of unwrapping

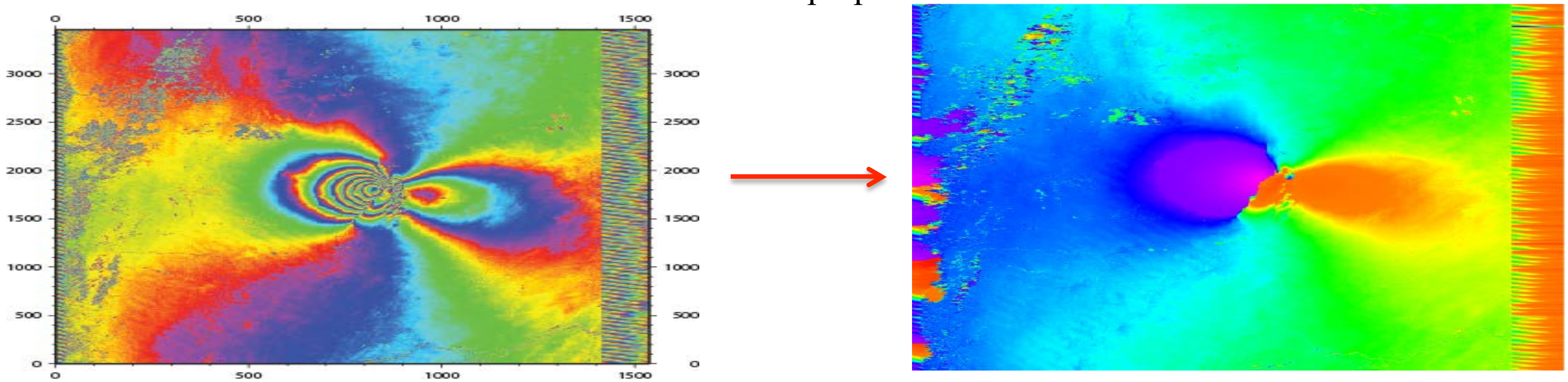
Given an interferogram(s)

- usually need to convert to useful units
- convert phase to m, cm, mm - we know radar wavelength
- we know geometry

Requires unwrapping

- often filter beforehand
- unwrapping not trivial

Saudi 7/1/08-8/19/08 ALOS Bperp 20 m



For 2D data the same problems exist.

How to unwrap?
Want to find a function that when wrapped, is "close" to the observed data (whatever "close" is).

Two basic approaches:

- Global methods that attempt to unwrap all pixels simultaneously.

- Local methods that solve along a path.

Note that the correlation data allows some estimation of how good the phase data is (maybe also amplitude/phase stability over time?)

Global

We want to find the function whose local derivatives "match" the observed derivatives given some measure :

$$e^p = [(\Phi_{i+1,k} - \Phi_{i,k}) - \Delta^x_{i,k}]^p + [(\Phi_{ik+1} - \Phi_{ik}) - \Delta^y_{ik})]^p$$

P = exponent

Φ = unknown function

Δ = derivatives of the observed phase (can calculate from the complex phase).

For p = 2, this is equivalent to the discrete version of Poisson's Equation

Two basic ways to solve:

Transform : FFT, DCT (discrete cosine transform-be careful with b.c.'s)

Matrix (will allow weighting but now nonlinear and requires iteration)

Can vary the exponent (i.e. don't have to use 2)

•An elegant and easy solution, but….
doesn't work very well with noise.

•Tends to underestimate true phase when noise exists
(it's a least-square fit).

•No easy way to add weighting short of iterating.

Matrix methods solve:

$$\mathbf{Ax} = \mathbf{b}$$

With weighting: $\mathbf{WAx} = \mathbf{Wb}$

W = matrix of weights
A = operator
B = set of observed phase values
x = unknown function

These work better than the transform methods but like all global solutions, do not provide a good fit anywhere.

Transform-based methods

Fast, but do not allow weighting

FFT requires periodic conditions and extension of data.

Apply 2D Fourier transform:

$$\rho_{i,k} = (\Delta^x_{i,k} - \Delta^x_{i-1,k}) + (\Delta^y_{i,k} - \Delta^y_{i,k-1})$$

$$\Phi_{m,n} = \frac{P_{m,n}}{2\cos(\pi m / M) + 2\cos(\pi n / N) - 4}$$

Φ = Fourier transform of ϕ

P = Fourier transform of ρ

1. Calculate the $\rho_{i,k}$ from the data
2. Calculate the 2D FFT of $\rho_{i,k}$
3. Calculate $\Phi_{m,n}$ from the transformed $\rho_{i,k}$
4. Do inverse FFT

Local (path following)

Similar to the 1D approach

1.) Calculate the differences of the wrapped phase.

2.) Wrap the differences.

3.) Set the value of the first value.

3.) Integrate along all values.

Do this along a line throughout 2D area (in a zigzag back and forth along the rows, for example)

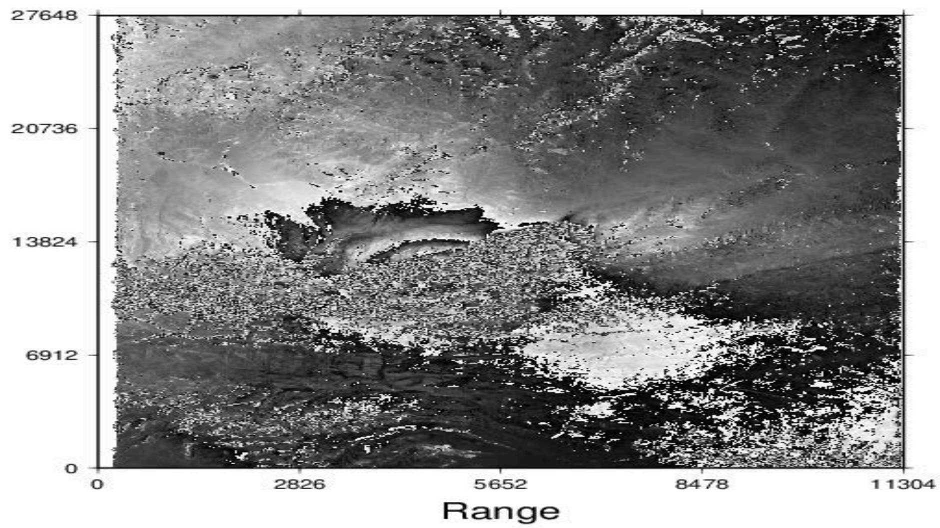
Works great if there is no noise.

With noise:

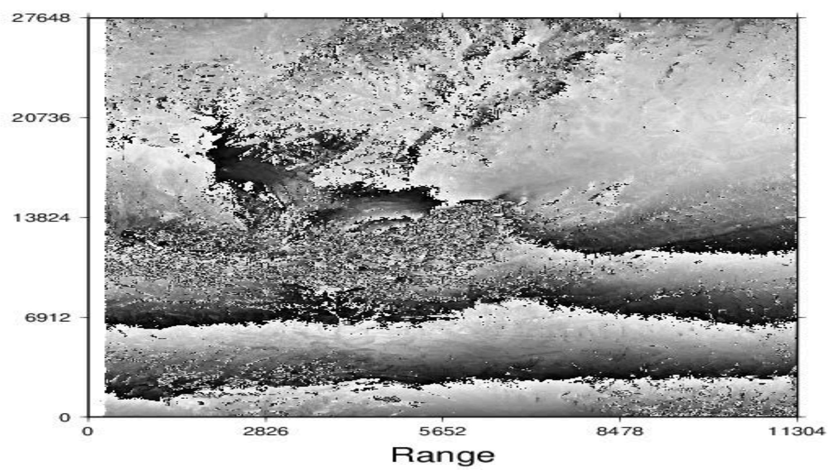
1) An error near the start of the path propagates along the whole path.

2) Answer may vary with path.

3) Need to identify bad pixels. How?



27648
20736
13824
6912
0
0
2826
5652
8478
11304
Range



27648
20736
13824
6912
0
0
2826
5652
8478
11304
Range

We know that topographic surfaces are conservative

- Any points that violate this rule should be avoided.
- These points are known as residues.
- Any integration path that circles a residue will contain errors

  => need to make “branch cuts”
- A residue is a property of phase differences, not a single pixel.
- can be positive or negative

1.) Identify all residues in data (marked as the upper left pixel)
2) Draw lines (branch cuts) between them to eliminate possibility of drawing a circle around a residue.
3) Unwrap the rest of the data

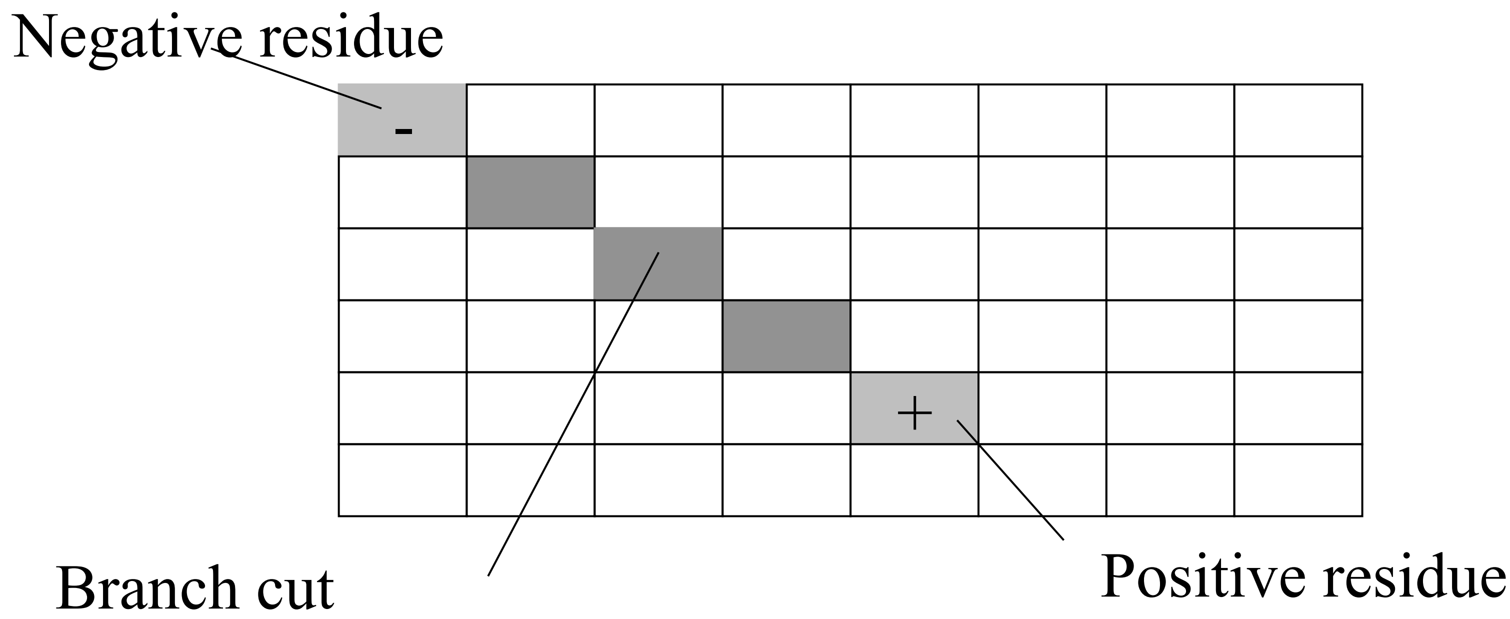
*Note: residues can be positive or negative.
A positive residue linked to a negative residue cancel each other out.

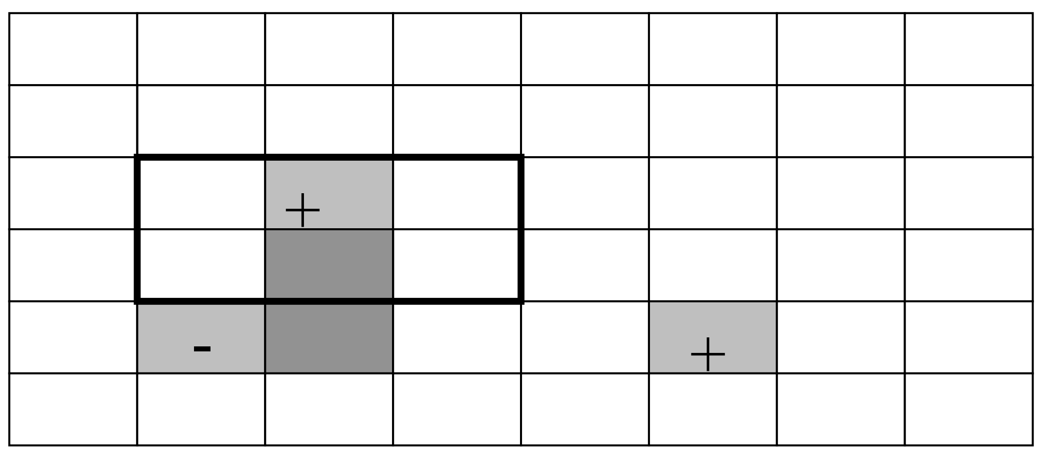
This is the basis of the Goldstein approach, (used in Roi_pac).

Often, poor data (with low correlation is masked out beforehand)

Goldstein* algorithm
[different from filtering]

1.) Calculate correlation for phase data.
2.) Mask out all areas with correlation less than a certain threshold value.
3.) Go through all pixels and identify residue locations (upper left of 4 pixels).
4) Start with first residue, look for nearest residue. Draw a “line” of marked pixels between the two.
- if residues cancel, go to next residue and start new “tree”
- otherwise, look for next nearest and draw line
- can also “cancel” by connecting to edge.
- connected lines are called a tree.
5) path-integrate along remaining pixels.





1) start

2) search in box

3) find second residue

4) connect with branch cut

•Fast
•Need to start integration (seed point) in area of good data.
•Residues often lie in areas of layover.
•Regions can get isolated from each other with dramatically different phase (by multiples of 2π).
•Some implementations allows manual connecting of regions.
•Some implementations allow "pre-processing" to connect closely-spaced opposite residues (dipoles) first.

•Minimizes distance between residues; does not minimize number of cycles needed to "unwrap".

•A similar algorithm uses quality rather than residues to define.

Flynn's minimum discontinuity

1) Identify lines of discontinuity (fringe lines)
    1) Difference between adjacent pixels > π
    2) Magnitude of discontinuity defined by number of multiples of 2π needed to fix.
2) Add multiples of 2π to eliminate lines of discontinuities that form loops.
3) Checks to see if operation creates more discontinuities than removes.
4) Continues in an iterative fashion.
5) At end, no more discontinuities can be removed without adding more.
6) Complicated algorithm (i.e. I looked at it and got a headache)

Comments:

- Slow. Masking helps.
- <u>Can be appended after other unwrapping algorithms</u>.

Filtering

objective: improve signal-to-noise

static
- usually lowpass
- convolve with set of filter coefficients (boxcar, Gaussian, etc)

adaptive
-*Goldstein and Werner* [1998] spectral filter.
- effective but “can significantly change the structure of the interferogram”[*Baran et al*, 2003]

adaptive filter

- filter parameter vary depending of properties of each patch

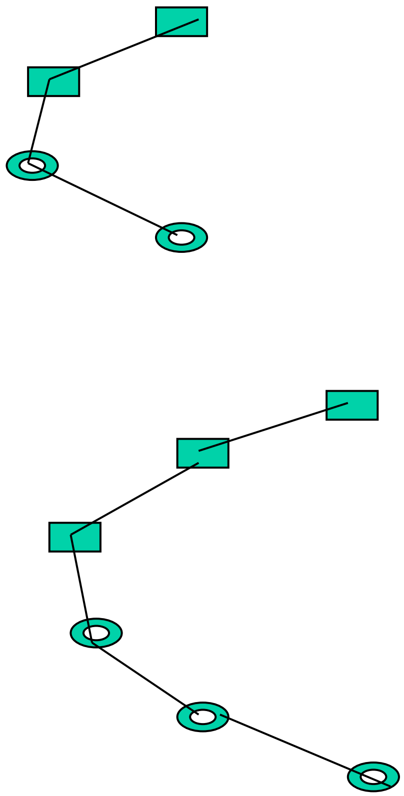
$$H(u,v) = (S\{|Z(u,v)|\}^{a})(Z(u,v))$$

Z(u,v)   Fourier spectrum of small 2D patch of complex interferogram
(perhaps 32 by 32 pixels)

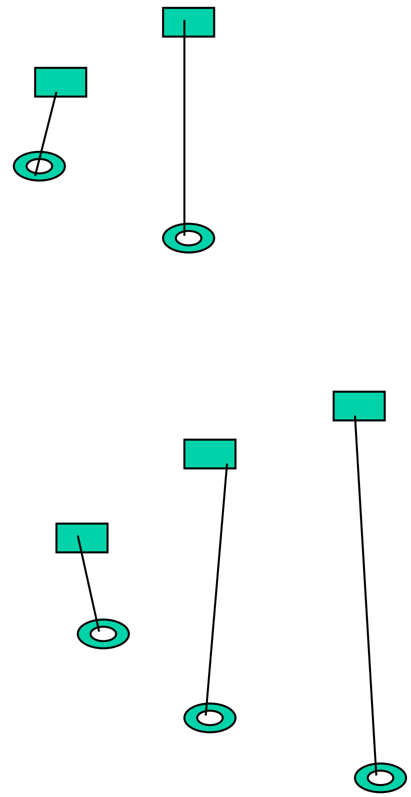
S   smoothing (e.g. 3 by 3 pixels)
H   output
α exponent (a = 0, no filter, a > 0 filters)

-take Fourier of small patch
-raise spectrum to power
- inverse FFT
-results depends on noise and phase!
- *Baran et al* [2003] use $\alpha = 1 - \gamma$ where γ is the coherence

Residues caused by topographic layover with illumination from one side



Good – L0



Bad – L1

Negative residue

Positive residue

The ultimate L0 algorithm would minimize the total cut length.

C&Z show that the L0 problem is equivalent to an NP-complete problem and therefore hard for efficient algorithms to solve completely.

•Therefore, the L0 branch-cut algorithm is a good place to start.
•Major problem is that cuts close on themselves.
•Total tree length is upper bound on total discontinuity length.

Modifications

1) Minimum span (MST)
    1) Define cuts so that a tree cannot connect to itself.
    2) Connect all trees.
    3) Use knowledge of residues to guide integration.
    4) Complete unwrapping
2) Minimum cost (MCF)
    1) Uses flow to reduce cycles

Removing topography (and deformation)

If an accurate topographic model is available, then many of these problems can be alleviated during calculation of the interferogram.

•Reduce need for unwrapping.
•Deformation model can also be included.
•Can be done iteratively.
•Also true for large deformations (maybe done in an iterative fashion)

The future:
persistent scatterers
3D unwrapping

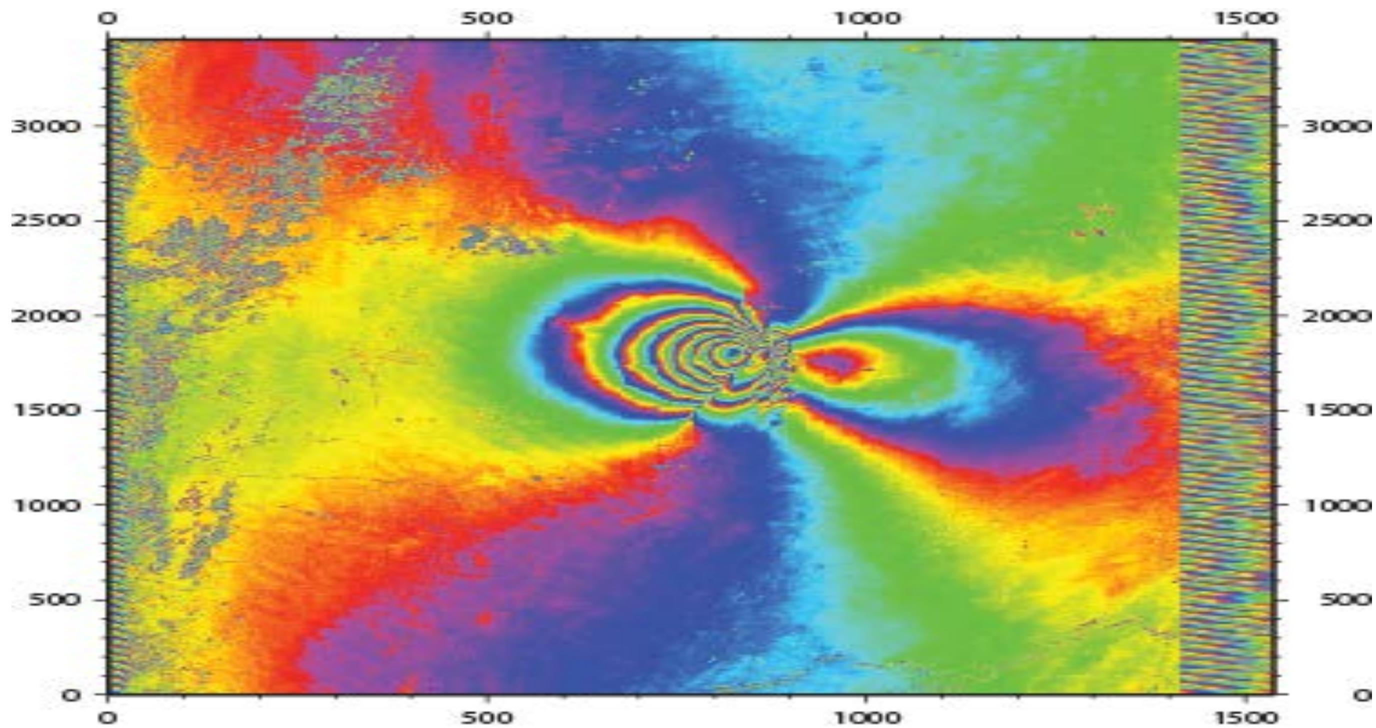
•The PS technique leads to widely spaced pixels. Phase relationships between these pixels may be challenging to define.

•If we have a time series of interferograms, phase unwrapping becomes a 3d problem.

Geocoding: One method

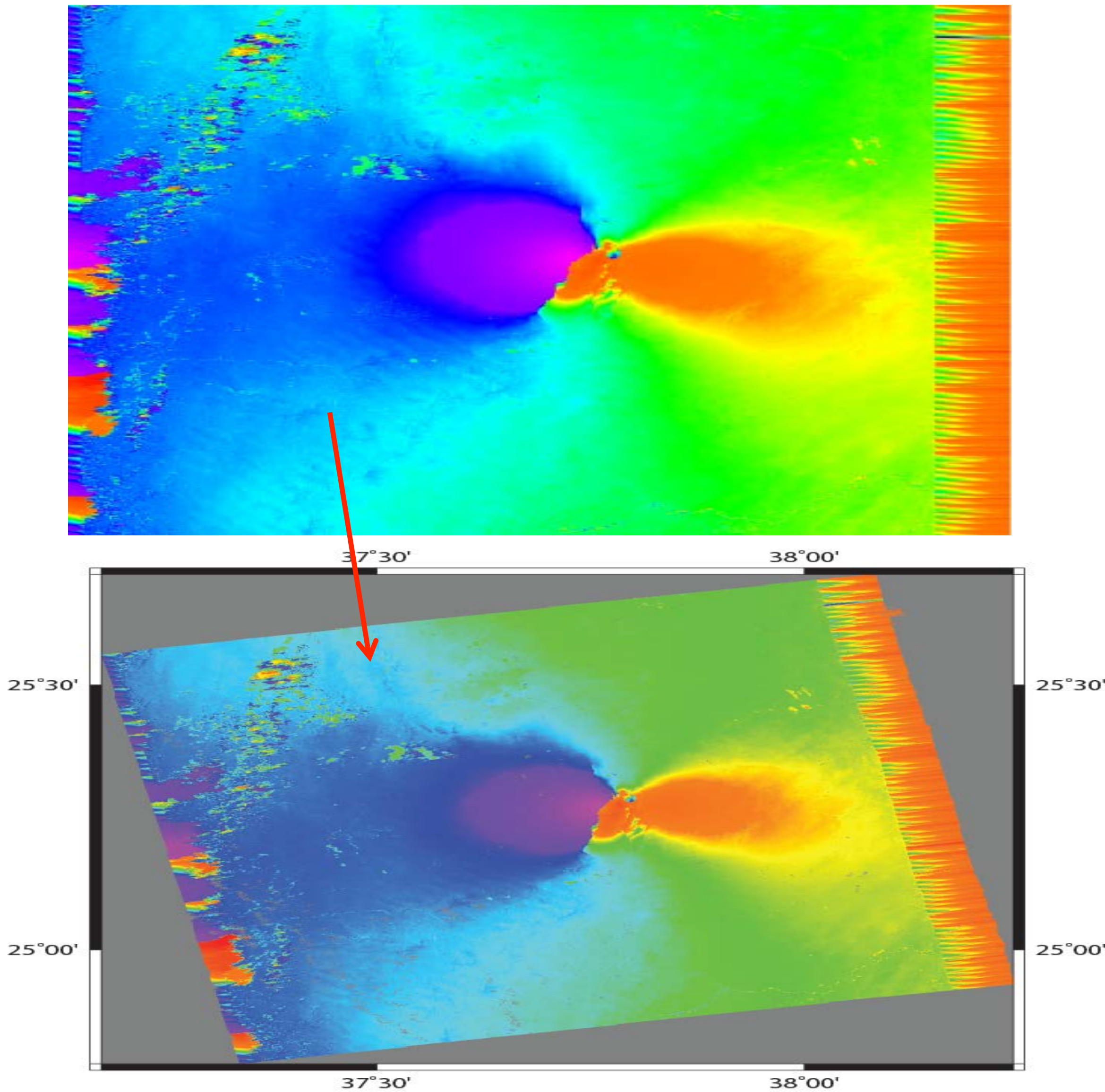
•"Fly" satellite along path with known orbit
•Find point (azimuth) of nearest approach to each DEM pixel with known latitude and longitude.
•Distance to point yields range
•This provides a mapping of range, azimith to latitude, longitude.

sometimes filtering is not necessary



Saudi
7/1/08-8/19/08
ALOS
Bperp 20 m
(low coherence masked out)

Geocoding



want to eliminate "fringes"