The example of disaster management

INDONESIA / JAVA - Bantul Region - Earthquake May 27, 2006 - Damage Assessment - Overview

INDONESIA / JAVA - Damage Assessment of the Earthquake on May 27, 2006 - Jetis Region

IKONOS PRE-DISASTER IMAGE - May 9, 2006
IKONOS POST-DISASTER IMAGE - May 28, 2006
The example of disaster management

New constraints

- Short system time response
  - Much more than a short revisit cycle
  - Reactive satellite tasking
  - Quick image availability
    - Actually, information availability
- High spatial resolution + wide swath
  - Being able to see details ...
  - ... without actually knowing where to look for them!
- High spatial + spectral resolution
  - Geometric and radiometric information to finely discriminate objects and their characteristics
- Permanent (geostationary) and high resolution (low orbit) observation
How to find a solution

- Don’t worry, space agencies have clever people to solve this kind of problems
- Inter-satellite links for retasking and up/downlinks
- Agile platforms for one-pass image mosaics
- Virtual huge telescopes by aperture synthesis
- Deployable antennas
- Constellations of smaller, cheaper satellites
- Many clever devices that you thought couldn’t exist out of Star Trek
- And other top secret things I can’t talk about ...

Still any challenge for IP people?

- The solutions cited above are still expensive
  - An IP-based solution is software
  - Software is cheap. Best software is free!
- Users’ expectations go further and further (and quicker)
  - Everybody has a GPS receiver, Google Earth, Open Street Maps, etc.
  - Thus new imaginative uses for geospatial information appear every day
  - Updating software is quicker and cheaper than building a satellite
New value for old sensors

- RS scientists are good at finding new uses for existing sensors
  - some examples were given before
  - sometimes is just a proof of concept: DEM extraction with SPOT 5 P+XS
  - sometimes operational applications are built upon these ideas: Persistent Scatterers Interferometry
- In some sense, the International Charter Space and Major Disasters uses this approach
  - Most of the available systems are not designed for disaster management
- 2 paths leading to these new ideas
  - a real need triggers the idea:
    - how can I deliver this fancy information with this set of old crappy sensors?
  - take the system to its limits: squeeze the resolution, interpolate, deconvolve

Existing ancillary data

- Why extracting from images what you already (nearly) have elsewhere?
  - Maps, vector data bases, the Web, etc.
  - Remember Kalman
    - Update your guesses (ancillary) with new data (RS)
- However
  - How to assess the quality of ancillary data?
  - How to compare ancillary data and RS images?
    - Level of representation?
  - How to assess the quality of the updated geoinformation?
    - Measuring the performances of the algorithms: reliability, precision, accuracy, etc.
Measuring the performances of a system

▶ Assess
  ▶ the quality of the input information
  ▶ the performances of each individual processing block
  ▶ the quality improvement introduced by fusion

▶ So the quality of the output is known

Fusing new kinds of data

▶ Beware of fancy, useless fusion
  ▶ Have you heard of optical/SAR fusion? Me too.
  ▶ Have you seen real applications of it? Me neither!

▶ The way to put it is:
  ▶ Given a user need, which is the most straightforward way
    (data availability, price, delays) to get the information?
  ▶ If this means that fusion is needed, go for it.
  ▶ Usually, pixel-level fusion is not the best way to do it
  ▶ except for pan-sharpening, and maybe medium resolution
    multi-temporal series

▶ On the other hand:
  ▶ Fusing any kind of data may be possible. Just choose the
    appropriate representation level.
New computing approaches

- VHR means huge data volume
- Parallelism, streaming:
  - These are techniques allowing to overcome computation time and memory capability limitations
  - They need to split the data in order to process them
  - Algorithms which can not process data by chunks are dead!
- Scalability of algorithms is crucial for end-user applications