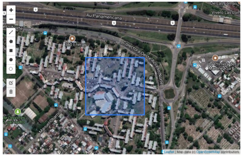
Automated 3D reconstruction from satellite images

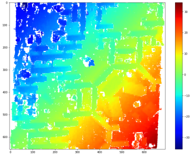
SIAM IS18 Minitutorial

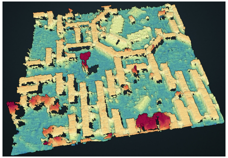
Gabriele Facciolo, Carlo de Franchis, and Enric Meinhardt-Llopis

Centre de Mathématiques et de Leurs Applications (CMLA)
École Normale Supérieure Paris - Saclay

June 8, 2018

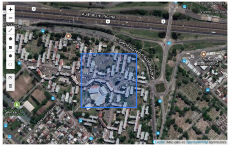


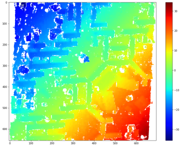
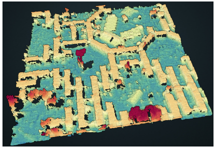




Overview

- This is a hands-on course on satellite image processing
- Objective: go from satellite images to a 3D model



- Plan and logistics
  1. Basics of satellite images
     - First on-line exercise with Jupyter at https://gfacciol.github.io/IS18/
  2. Stereovision with satellite images
     - Second on-line exercise with Jupyter Notebook
  3. Triangulation and digital elevation model generation
     - Third on-line exercise with Jupyter
  4. 3D from collections of images

Speakers


Gabriele Facciolo


Carlo de Franchis


Enric Meinhardt-Llopis

Thank:

- Mila Nikolova, Fiorella Sgallari, and all the organizers
- Jean-Michel Morel and all our colleagues at *ENS Paris-Saclay* and *Master2 MVA*
- Katherine Scott for the inspiring *"Python from Space"* notebooks

Acknowledge:

cnes DGA FINANCÉ PAR ANR ONR Revolutionary Research . . . Relevant Results MBDA MISSILE SYSTEMS KAYRROS

Towards a daily cover of the Earth

Increasing resolution

- 2008: RapidEye, 5 satellites (5m)
- 2008: GeoEye (0.4m)
- 2011: Pleiades, 2 satellites (0.7m)
- 2013: SkySat, 5 satellites (1m)
- 2015: Sentinel-2, 2 satellites (10m)
- 2017: Planet Labs Flock-1, 150 μ-satellites (3-5m)
- 2018: Satellogic, 5 μ-satellites (1m), $\sim$**300 planned**

Need for automatic analysis

- Automatic event detection, ignoring radiometric changes
- Independence of acquisition geometry
- Use historical data

Port-au-Prince in 2012
(after the 2011 quake)

and in 2013

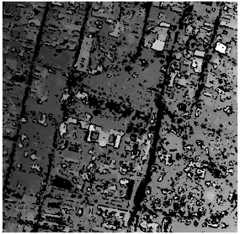
Towards a daily cover of the Earth

Increasing resolution

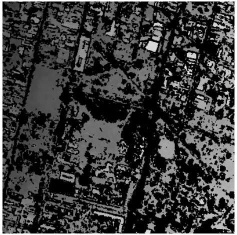
- 2008: RapidEye, 5 satellites (5m)
- 2008: GeoEye (0.4m)
- 2011: Pleiades, 2 satellites (0.7m)
- 2013: SkySat, 5 satellites (1m)
- 2015: Sentinel-2, 2 satellites (10m)
- 2017: Planet Labs Flock-1, 150 μ-satellites (3-5m)
- 2018: Satellogic, 5 μ-satellites (1m), $\sim$**300 planned**

Need for automatic analysis

- Automatic event detection, ignoring radiometric changes
- Independence of acquisition geometry
- Use historical data



Elevations in 2012
(after the 2011 quake)



Elevations in 2013

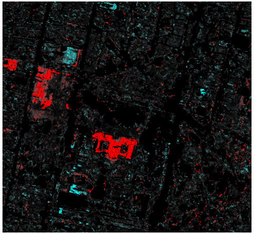
Towards a daily cover of the Earth

Increasing resolution

- 2008: RapidEye, 5 satellites (5m)
- 2008: GeoEye (0.4m)
- 2011: Pleiades, 2 satellites (0.7m)
- 2013: SkySat, 5 satellites (1m)
- 2015: Sentinel-2, 2 satellites (10m)
- 2017: Planet Labs Flock-1, 150 μ-satellites (3-5m)
- 2018: Satellogic, 5 μ-satellites (1m), $\sim$**300 planned**

Need for automatic analysis

- Automatic event detection, ignoring radiometric changes
- Independence of acquisition geometry
- Use historical data



Difference between the 2012 and 2013 elevations (Demolished, Built)

What a time to be alive!

Traditionally, newcomers to remote sensing should pay a high cost (in terms of knowledge) to work with satellite images.

- access to images
- file formats
- cartographic standards
- projection models
- specialized software
- etc...

A short Sentinel 2 time series

Today this is no longer true:

```
import tsd  # timeseries downloader
aoi = {'coordinates': [[[2.306, 48.831], [2.306, 48.869],
        [2.376, 48.869], [2.376, 48.831]]], 'type': 'Polygon'}
tsd.get_sentinel2(aoi, out_dir='paris') # download data from ESA
```

Why 3D digital models?

They are an essential tool for:

- large-scale measurements:
  - snow height on glaciers [Berthier et al. 2014]
  - forests evolution [Gumbricht 2012]
  - assessment after natural disasters [Yésou et al. 2015]
- change detection [Chaabouni-Chouayakh et al. 2010]
- cartography (orthorectification) [Leprince et al. 2007]
- more generally, image comparison

Bassies (Pyrénées), 2015-03-11

Why 3D digital models?

They are an essential tool for:

- large-scale measurements:
  - snow height on glaciers [Berthier et al. 2014]
  - forests evolution [Gumbricht 2012]
  - assessment after natural disasters [Yésou et al. 2015]
- change detection [Chaabouni-Chouayakh et al. 2010]
- cartography (orthorectification) [Leprince et al. 2007]
- more generally, image comparison

Bassies (Pyrénées), 2014-10-26

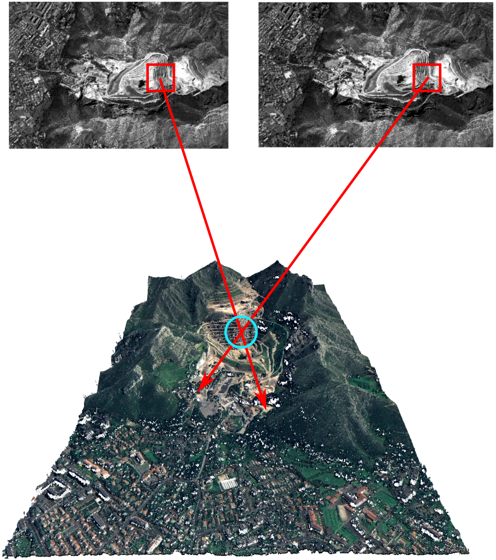
3D models and orthorectification

Images: AirbusDS Pleiades

Stereovision core idea: parallax is proportional to height

Images: AirbusDS Pleiades

3D reconstruction from image pairs

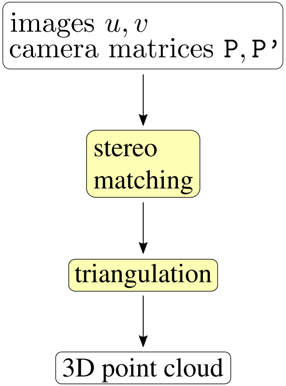


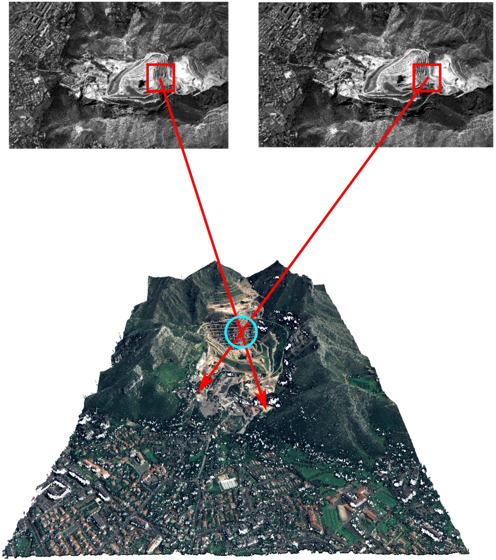
General principle:

- find corresponding pixels
- intersect the back-projected 3D lines

Need a camera model, and its parameters.

3D reconstruction from image pairs

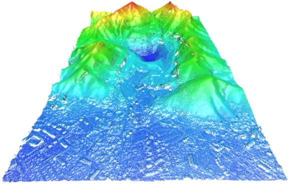




Satellite Stereo Pipeline (S2P)

- Modular 3D stereo pipeline for satellite images
- Developed at ENS Paris-Saclay and CNES
- Open source: https://github.com/MISS3D/s2p
- Currently used by CNES in production





An automatic and modular stereo pipeline for pushbroom images,
C. de Franchis, E. Meinhardt-Llopis, J. Michel, J.-M. Morel, G. Facciolo. ISPRS Annals, 2014.

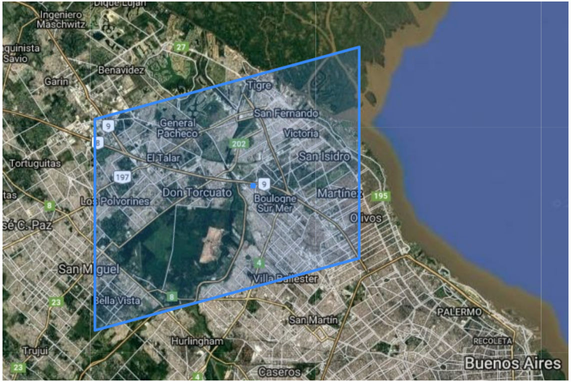
Images: AirbusDS Pleiades

Section 1. Coordinate Systems and Geometric Modeling

Satellite images == big data

Put a camera in space

- altitude: 400 to 700km
- acquire very large images 40000×40000 pixels

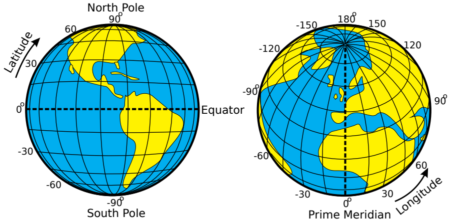


footprint of an entire image

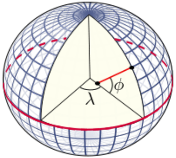
a small crop

Geographic and projected reference systems

Geographic: describe 3D points relative to a reference ellipsoid using **latitude**, **longitude**, and **altitude**

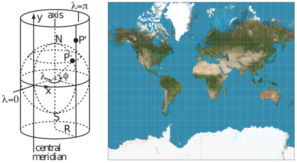


The reference ellipsoid:

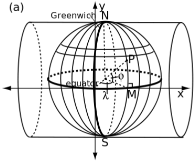


Projected: transform the elliptical earth into a flat surface

- **Mercator:** preserves shapes but not size



- **Universal Transverse Mercator:** uses **easting** and **northing**



The Linear Pushbroom Camera

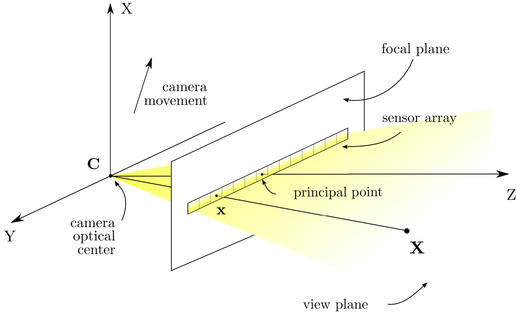
Similar to a pinhole camera, but:

- only **one line** of pixel sensors in the focal plane,
- the camera center **moves** at constant speed.

Internal parameters:

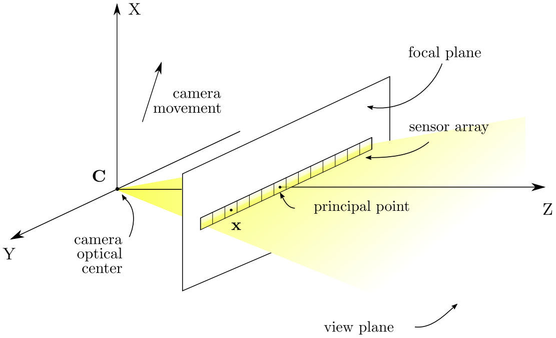
- focal length: f,
- position of the principal point: y_0,
- size of the pixel sensors: w,
- dwell time: δ_t.

[Gupta and Hartley 1997] R. Gupta and R. Hartley. Linear pushbroom cameras. *TPAMI*, 1997.



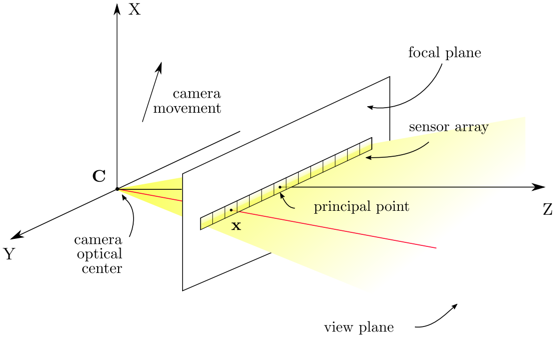
Geometric relationship between image and space

- Given a complete set of parameters, and an image point $\mathbf{x}$, what is the back-projected ray?



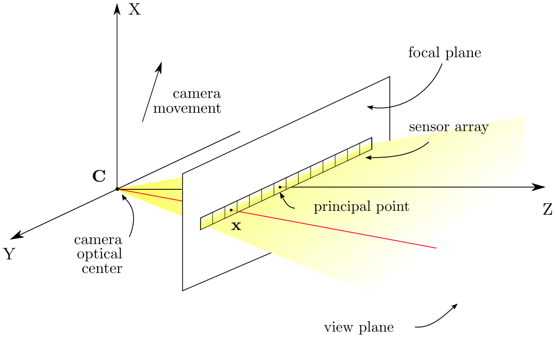
Geometric relationship between image and space

- Given a complete set of parameters, and an image point $\mathbf{x}$, what is the back-projected ray?



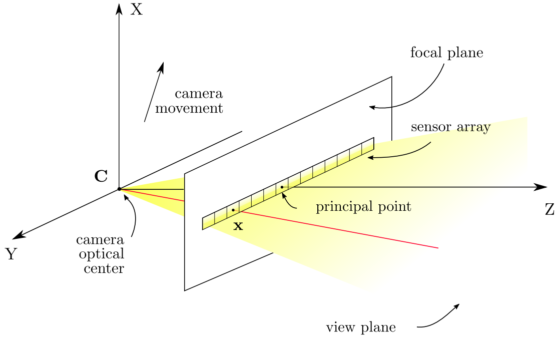
Geometric relationship between image and space

- Given a complete set of parameters, and an image point $\mathbf{x}$, what is the back-projected ray?
- Where does it intersect the Earth surface?



Geometric relationship between image and space

- Given a complete set of parameters, and an image point $\mathbf{x}$, what is the back-projected ray?
- Where does it intersect the Earth surface?
- Express this line in a coordinate system which rotates with the Earth.



Geometric relationship between image and space

$$\begin{bmatrix} c_{\tau+\lambda_0} & -s_{\tau+\lambda_0} & 0 \\ s_{\tau+\lambda_0} & c_{\tau+\lambda_0} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_{i-\frac{\pi}{2}} & -s_{i-\frac{\pi}{2}} \\ 0 & s_{i-\frac{\pi}{2}} & c_{i-\frac{\pi}{2}} \end{bmatrix} \begin{bmatrix} c_{-\alpha_t-\frac{\pi}{2}} & 0 & s_{-\alpha_t-\frac{\pi}{2}} \\ 0 & 1 & 0 \\ -s_{-\alpha_t-\frac{\pi}{2}} & 0 & c_{-\alpha_t-\frac{\pi}{2}} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\varphi & -s_\varphi \\ 0 & s_\varphi & c_\varphi \end{bmatrix} \begin{bmatrix} c_\psi & 0 & s_\psi \\ 0 & 1 & 0 \\ -s_\psi & 0 & c_\psi \end{bmatrix} \begin{bmatrix} c_\omega & -s_\omega & 0 \\ s_\omega & c_\omega & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{pmatrix} 0 \\ w(y-y_0) \\ f \end{pmatrix}$$

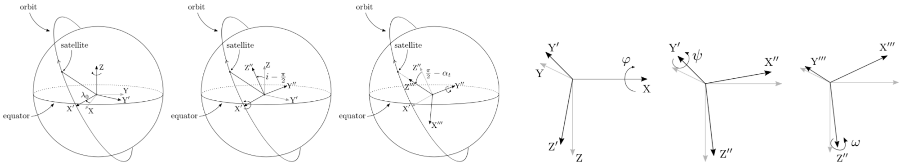
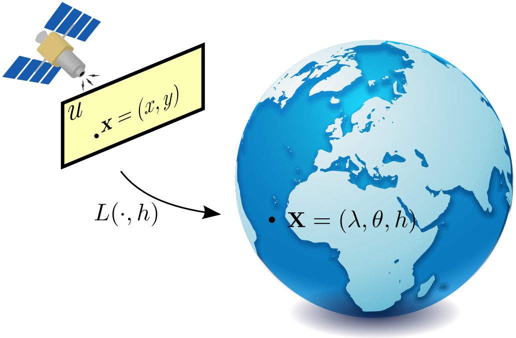


Image formation model

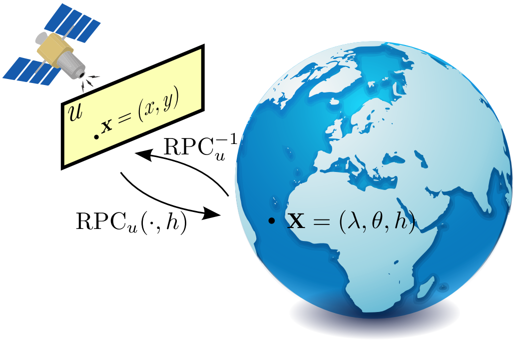
Localization function:

$$L : \mathbf{R}^2 \times \mathbf{R} \to [-\pi, \pi] \times [-\frac{\pi}{2}, \frac{\pi}{2}]$$
$$(\mathbf{x}, h) \mapsto (\lambda, \theta),$$



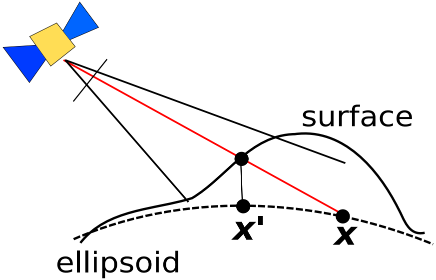
The Rational Polynomial Camera Model

- A **true camera model** is difficult to implement
- For end-users, image vendors provide a very close approximation of the *localization* function L, given as a **Rational Polynomial Functions** with degree 3.
- Its inverse, with respect to $\mathbf{x}$, is given as well.



Localization of a point on a non-elliptical earth

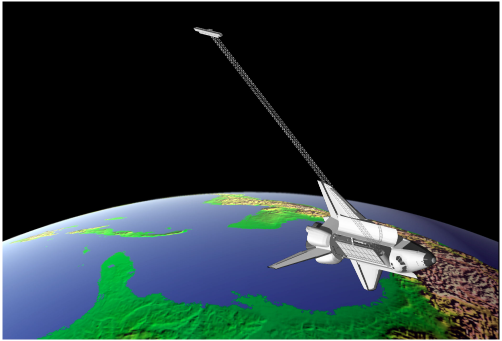
Find the coordinates of an image point on the globe

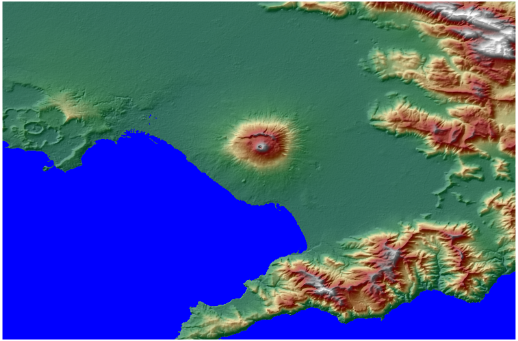


Shuttle Radar Topography Mission (SRTM)

SRTM provides a near-global high-resolution digital topographic database of Earth

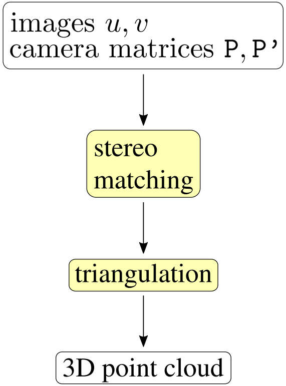
- Acquired in the year 2000
- Resolution 30 m

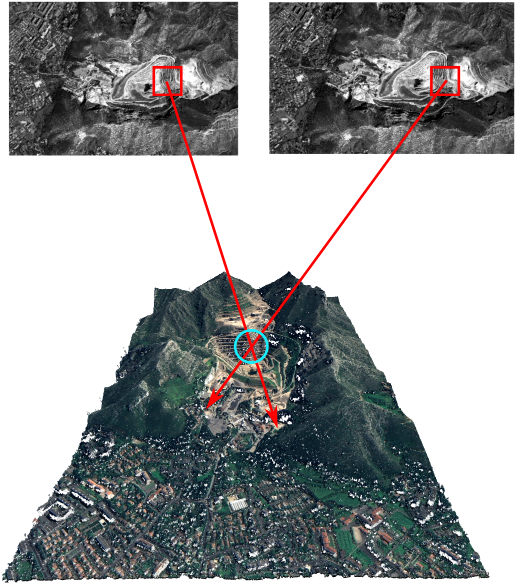




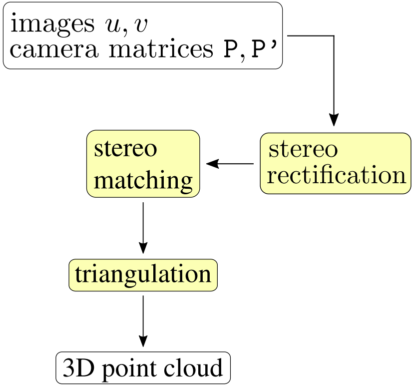
Section 2. Epipolar Rectification and Stereo Matching

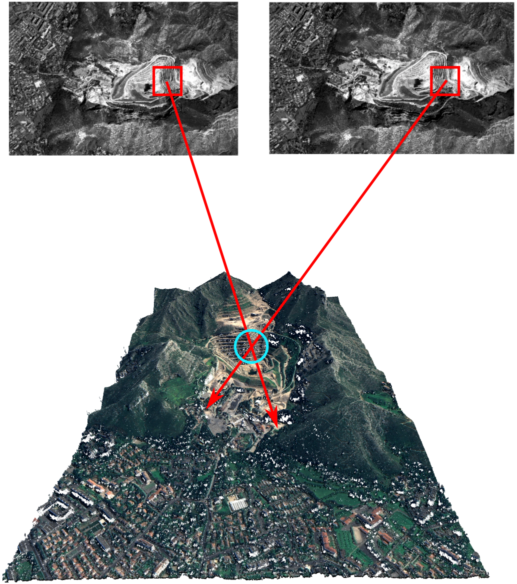
Baseline 3D reconstruction algorithm





Baseline 3D reconstruction algorithm





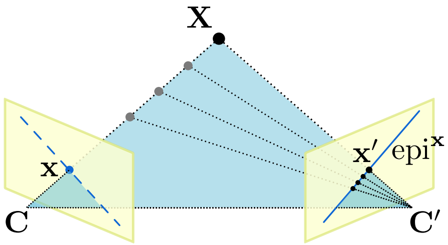
Epipolar rectification: what is it?

Process of **resampling** the images in such a way that depth variations cause **apparent motion** in the **horizontal** direction only.

Pinhole cameras

- $\mathbf{C}, \mathbf{C}'$ and $\mathbf{x}$ define a plane, called the **epipolar plane**.
- Its intersection with the second image is the **epipolar line** of $\mathbf{x}$, denoted by $\mathrm{epi}^{\mathbf{x}}$.
- All the $\mathbf{x}' \in \mathrm{epi}^{\mathbf{x}}$ share the same epipolar plane, hence the **same** epipolar line in the first image.

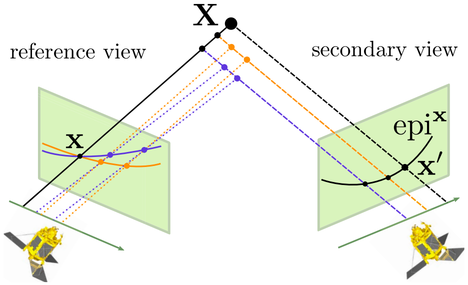
Conclusion: there is a one-to-one correspondence between epipolar lines.



Pushbroom cameras

- Satellite cameras are not **pinhole**, but **pushbroom**.
- As the camera center moves, the epipolar plane becomes a **doubly ruled surface**, namely a **hyperbolic paraboloid**.
- Epipolar lines become **curves**, still denoted by $\mathrm{epi}^{\mathbf{x}}$.
- All the $\mathbf{x}' \in \mathrm{epi}^{\mathbf{x}}$ have a **different** epipolar surface, hence a **different** epipolar line in the first image.

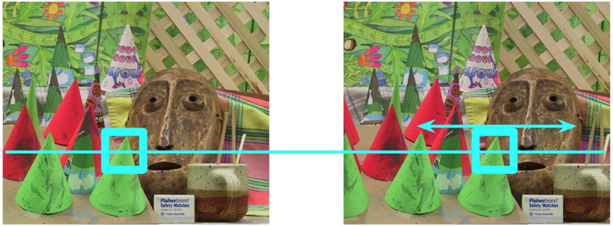
Conclusion: there is **no** one-to-one correspondence between epipolar curves.



Epipolar rectification: why?

Why epipolar rectification:

- speed: reduces the exploration from 2D to 1D
- robustness: reduces the risks for false matches
- compatibility: allows to use standard stereo-matching algorithms



It is just an intermediate step. Then it could be done **locally**. What if you try to **approximate** locally the pushbroom camera model with a pinhole camera model?

RPC approximation

Let $P : \mathbb{R}^3 \to \mathbb{R}^2$ be the RPC projection function. The first order Taylor approximation of P around point X_0 is

$$\begin{aligned} P(X) &= P(X_0) + \nabla P(X_0)(X - X_0) \\ &= \nabla P(X_0)X + T \end{aligned} \tag{1}$$

with $T = P(X_0) - \nabla P(X_0)X_0$ and $\nabla P(X_0)$ the jacobian matrix.

This can be rewritten using homogeneous coordinates as

$$P(X) = \underbrace{\begin{bmatrix} \nabla P(X_0) & T \\ 0 & 1 \end{bmatrix}}_{\text{matrix of size (3, 4)}} \begin{bmatrix} X \\ 1 \end{bmatrix} \tag{2}$$

This is the projection function of an **affine camera**! [Hartley and Zisserman 2004]

Epipolar rectification: how?

In general:

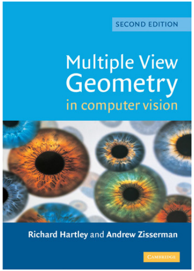
1. Find keypoint matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ with SIFT [Lowe 2004, Rey Otero 2014]
2. Estimate the fundamental matrix $\mathtt{F}$ [Hartley and Zisserman 2004] with RANSAC.

$$\mathbf{x}_i'^{\top} \mathbf{F} \mathbf{x}_i = 0$$

Epipolar rectification: how?

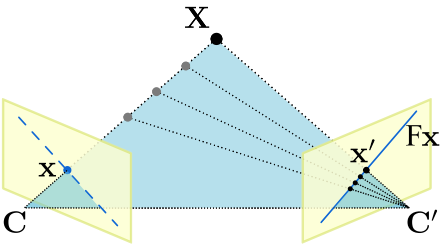
What is the fundamental matrix?

The fundamental matrix is the algebraic representation of epipolar geometry. [Hartley and Zisserman 2004]



The fundamental matrix $\mathtt{F}$ of a pair of cameras is a 3×3 matrix of rank 2 such that any pair of corresponding points $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ verify the equation

$$\mathbf{x}_i'^{\top} \mathtt{F} \mathbf{x}_i = 0.$$



The fundamental matrix song:

https://youtu.be/DgGV3l82NTk

Epipolar rectification: how?

In general:

1. Find keypoint matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ with SIFT [Lowe 2004, Rey Otero 2014]
2. Estimate the fundamental matrix $\mathtt{F}$ [Hartley and Zisserman 2004] with RANSAC.

$$\mathbf{x}'^{\top}_i \mathtt{F} \mathbf{x}_i = 0$$

Epipolar rectification: how?

In general:

1. Find keypoint matches $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ with SIFT [Lowe 2004, Rey Otero 2014]
2. Estimate the fundamental matrix $\mathtt{F}$ [Hartley and Zisserman 2004] with RANSAC.

$$\mathbf{x}_i'^{\top} \mathbf{F} \mathbf{x}_i = 0$$

3. Estimate resampling homographies $\mathtt{H}$ and $\mathtt{H}'$ [Loop Zhang 1999]

$$\mathtt{F} = \mathtt{H}'^{\top} \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \mathtt{H}$$

Epipolar rectification: how?

If you know the two camera matrices $\mathtt{A}$ and $\mathtt{B}$:

1. Compute the fundamental matrix $\mathtt{F}$ [Hartley and Zisserman 2004]

$$\mathtt{F}_{ji} = (-1)^{i+j} \det \begin{bmatrix} \sim \mathbf{a}^i \\ \sim \mathbf{b}^j \end{bmatrix} \tag{3}$$

   where $\sim \mathbf{a}^i$ denotes the matrix obtained from $\mathtt{A}$ by omitting the row $\mathbf{a}^i$. This formula expresses directly each entry of $\mathtt{F}$ in terms of determinants computed from the entries of $\mathtt{A}$ and $\mathtt{B}$.

2. Estimate resampling homographies $\mathtt{H}$ and $\mathtt{H}'$ [Loop Zhang 1999]

$$\mathtt{F} = \mathtt{H'}^\top \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \mathtt{H}$$

Epipolar rectification: how?

If the two cameras are **affine**, then:

- The fundamental matrix has a special form:

$$\mathtt{F} = \begin{bmatrix} 0 & 0 & a \\ 0 & 0 & b \\ c & d & e \end{bmatrix} \tag{4}$$

  This expresses the fact that the epipolar lines are bundles of **parallel** lines.
- The rectification can be achieved with just a **similarity** (composition of rotation, zoom and translation).

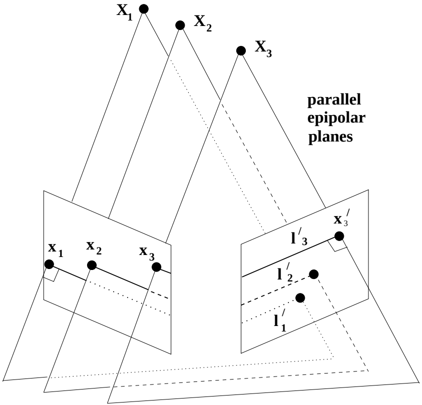


Figure courtesy of Hartley and Zisserman, 2004

Epipolar rectification: how?

Two similarities that transform the epipolar lines in a set of matching horizontal lines can be computed directly from $\mathtt{F}$.

$$\mathtt{S}_1 = \left[\begin{array}{cc|c} \multicolumn{2}{c|}{\multirow{2}{*}{$z\mathtt{R}_1$}} & 0 \\ & & t \\ \hline 0 & 0 & 1 \end{array}\right] \qquad \mathtt{S}_2 = \left[\begin{array}{cc|c} \multicolumn{2}{c|}{\multirow{2}{*}{$\frac{1}{z}\mathtt{R}_2$}} & 0 \\ & & -t \\ \hline 0 & 0 & 1 \end{array}\right] \tag{5}$$

where $z = \sqrt{\frac{r}{s}}$, $t = \frac{e}{2\sqrt{rs}}$ with $r = \sqrt{a^2 + b^2}$, $s = \sqrt{c^2 + d^2}$ and the two rotations $\mathtt{R}_1$ and $\mathtt{R}_2$ are given by

$$\mathtt{R}_1 = \frac{1}{\sqrt{a^2 + b^2}} \begin{bmatrix} b & -a \\ a & b \end{bmatrix} \qquad \mathtt{R}_2 = \frac{1}{\sqrt{c^2 + d^2}} \begin{bmatrix} -d & c \\ -c & -d \end{bmatrix} \tag{6}$$

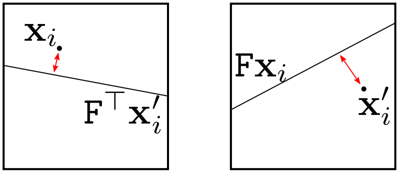
Epipolar rectification: conclusion

We have a **blind** way to rectify pushbroom images using a 1^{st} order Taylor approximation of their RPC camera model. How accurate is the approximation?

The are several ways to measure it:

1. Estimate the projection approximation error of a single camera:
   - estimate $\max||P(X) - AX||$ for X varying in a neighborhood of X_0
   - evaluate the 2^{nd} order term of the Taylor approximation
2. Estimate the rectification approximation error of two cameras:
   - 2.1 compute the fundamental matrix `F` of their affine approximations
   - 2.2 measure how well `F` fits the exact projections of some 3D points

Epipolar rectification: results

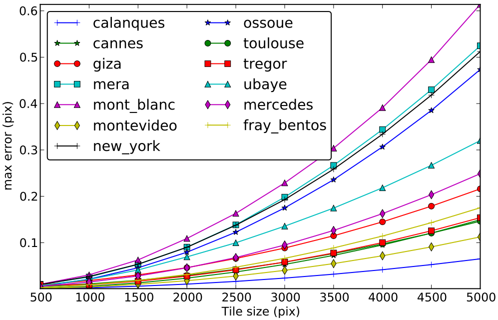


To evaluate the method, measure the **epipolar error**:

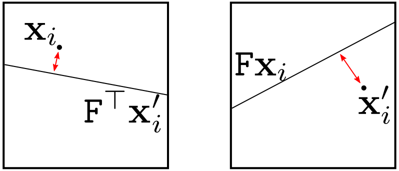
$$\max_{i \in \{1,\dots,n\}} \max\{d(\mathbf{x}'_i, F\mathbf{x}_i), d(\mathbf{x}_i, F^\top \mathbf{x}'_i)\},$$

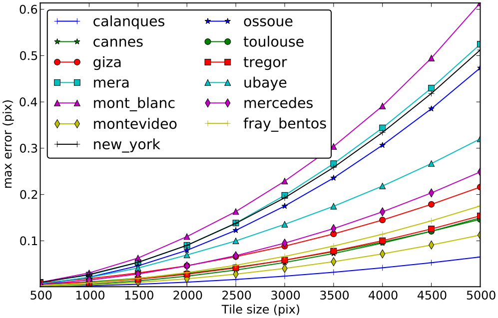
where $d(\mathbf{x}', F^\top \mathbf{x})$ is the **vertical disparity**:

$$d(\mathbf{x}', F\mathbf{x}) = \frac{|\mathbf{x}'^\top F\mathbf{x}|}{\sqrt{(F_1^\top \mathbf{x})^2 + (F_2^\top \mathbf{x})^2}}$$



Epipolar rectification: results





Conclusion:

- After epipolar rectification, the maximal error w.r.t true camera model (RPC) is only 0.05 pixel!
- Working with small areas of interest (e.g. 500×500 meters) permits to do the usual epipolar rectification with enough accuracy for stereo matching.

Epipolar rectification: in practice

input images

Epipolar rectification: in practice

rectified from the RPC's affine approximation

Epipolar rectification: in practice

rectified from the RPC's affine approximation

It's still moving vertically, isn't it?

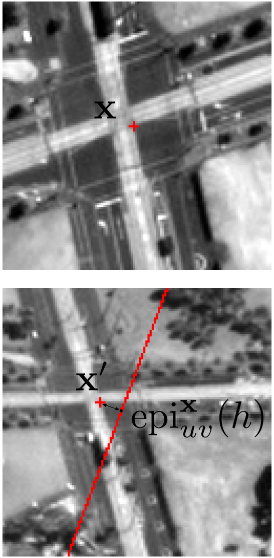
The relative pointing error

Due to **attitude measurement** inaccuracies, the RPC functions may contain an **error of a few pixels**.

Given two corresponding points $\mathbf{x} \leftrightarrow \mathbf{x}'$, the epipolar curve

$$\text{epi}_{uv}^{\mathbf{x}} : h \mapsto P_v(L_u(\mathbf{x}, h), h)$$

may not pass through $\mathbf{x}'$.

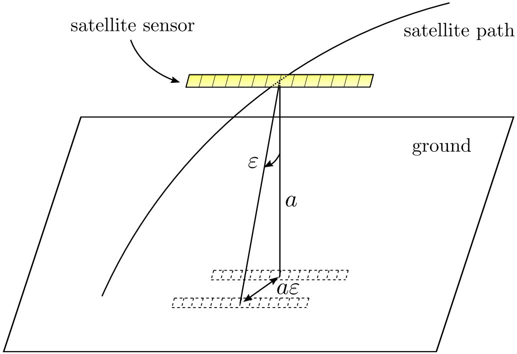


The relative pointing error: why?

The camera parameters are measured on board. What's their accuracy?

- internals: carefully calibrated (in-flight commissioning) ✓
- orbit parameters: cm accuracy with DORIS (GPS) intruments ✓
- attitude coefficients: a few tens of µrad ✗

$$a\varepsilon \approx 700\,\text{km} \times 50\,\mu\text{rad} = 35\,\text{m}$$



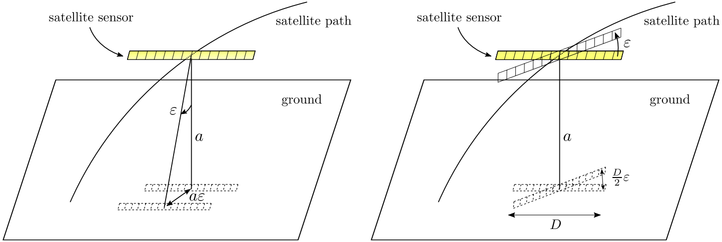
[de Lussy et al. 2012] Pléiades HR in flight geometrical calibration: location and mapping of the focal plane

Effect of attitude errors

The effect of a yaw error is negligible with respect to the effect of an error on roll or pitch.

$$a\varepsilon \gg \frac{D}{2}\varepsilon$$

- a: flying altitude
- D: swath width



Thus the effect of attitude errors is mostly a constant **image shift**.

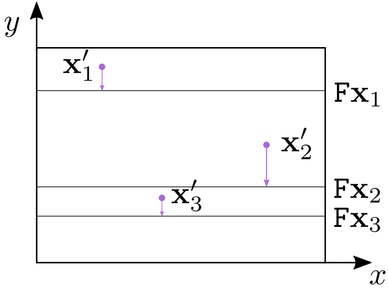
On small areas of interest

- epipolar curves are approximated by a bundle of **parallel lines**
- the effect of pointing error is approximated by a **constant offset**

Hence, given a set of **keypoint matches** (obtained with SIFT [Lowe 2004]), the error can be corrected with a **vertical translation** of the rectified images:

$$t^\star = \arg\min_t \frac{1}{N} \sum_{i=1}^{N} |y_i' - y_i + t|$$

where y_i and y_i' are the vertical coordinates of the keypoints in the rectified images.



Effect of pointing error before correction

[Lowe 2004]David G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004

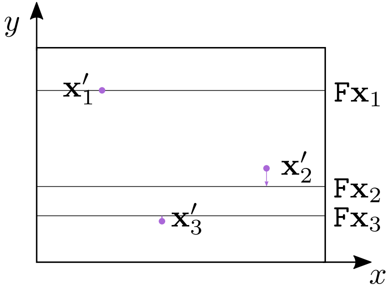
On small areas of interest

- epipolar curves are approximated by a bundle of **parallel lines**
- the effect of pointing error is approximated by a **constant offset**

Hence, given a set of **keypoint matches** (obtained with SIFT [Lowe 2004]), the error can be corrected with a **vertical translation** of the rectified images:

$$t^\star = \arg\min_t \frac{1}{N} \sum_{i=1}^{N} |y_i' - y_i + t|$$

where y_i and y_i' are the vertical coordinates of the keypoints in the rectified images.



Effect of pointing error after correction

[Lowe 2004]David G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, IJCV 2004

Local correction of the relative pointing error

before

after

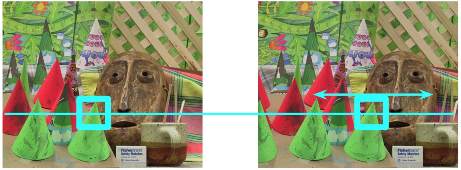
Robust stereo matching

Stereo matching computes correspondences between a pair of images (easier if they are **rectified**). We use SGM [Hirschmüller'05] to approximately minimize

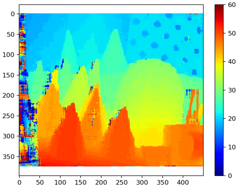
$$E(\mathsf{D}) = \sum_{\mathbf{p}\in\mathcal{V}} C_{\mathbf{p}}(\mathsf{D}_{\mathbf{p}}) + \sum_{(\mathbf{p},\mathbf{q})\in\mathcal{E}} V(\mathsf{D}_{\mathbf{p}}, \mathsf{D}_{\mathbf{q}})$$

Critical ingredients for remote sensing applications:

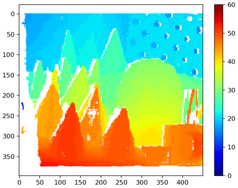
- **matching cost**: robustness to illumination changes (e.g. Census Transform [Zabih & Woodfill '94])
- **disparity post-processing**: removal of spurious matches (left-right, speckle)



Left and Right images



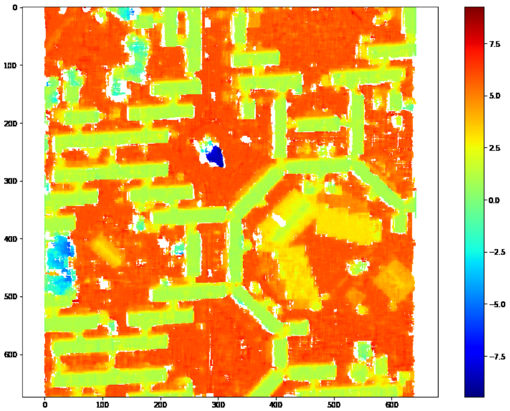
SGM+Census Disparity



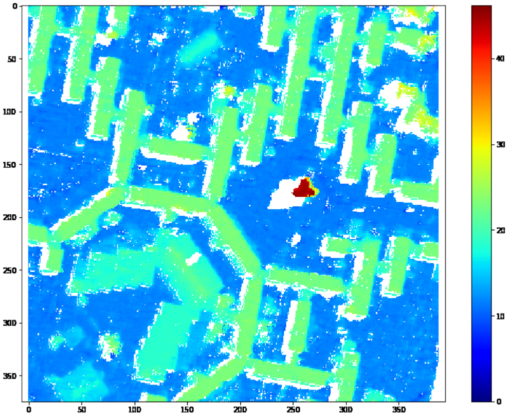
Filtered Disparity

Section 3. Triangulation and Digital Elevation Models

Disparity Charts vs. Elevation Maps : they are not the same!



disparity chart $d(i, j)$
$i =$ pixels
$j =$ pixels
$d =$ pixels

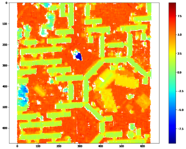
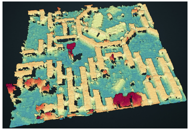
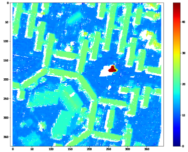


elevation map $z(x, y)$
$x =$ meters
$y =$ meters
$z =$ meters

Scheme of the whole pipeline (for one image pair)

Four steps to convert a pair of images to a DEM:

1.rectification 2.matching 3.triangulation 4.projection

 $\Rightarrow$  $\Rightarrow$  $\Rightarrow$  $\Rightarrow$ 

$I(i,j)$	$I(i,j)$	$d(i,j)$	$S \subseteq \mathbf{R}^3$	$z(x,y)$
original	rectified	disparities	3D points	DEM

Match triangulation

Input: a point correspondence between two satellite images
Output: a 3D point
Algorithm: (triangulation)

1. Let L_A, L_B, P_A, P_B be the localization and projection functions of each image
2. Let p, q be a corresponding pair of points between A and B
3. Solve the system of equations $\begin{cases} p = P_A(x, y, h) \\ q = P_B(x, y, h) \end{cases}$ or $P_B(L_A(p, h), h) = q$.

Match triangulation

Input: a point correspondence between two satellite images
Output: a 3D point
Algorithm: (triangulation)

1. Let L_A, L_B, P_A, P_B be the localization and projection functions of each image
2. Let p, q be a corresponding pair of points between A and B
3. Solve the system of equations $\begin{cases} p = P_A(x, y, h) \\ q = P_B(x, y, h) \end{cases}$ or $P_B(L_A(p, h), h) = q$.

Tricks:
1. In either case, the system is over-determined. You have to find a minimum-error solution.
2. Since the calibration functions are very smooth, you can linearize them and the system becomes linear (with a single unknown h, to be found by least squares).

Solving the triangulation equation

Given the a matching pair $\mathbf{p} \sim \mathbf{q}$, find h such that

$$P_B(L_A(\mathbf{p}, h), h) = \mathbf{q}$$

Observations:

1. Two equations and one unknown: overdetermined!
2. "Define" the solution as $h = \arg\min \|P_B(L_A(\mathbf{p}, h), h) - \mathbf{q}\|^2$
3. The functions P_B and Q_A are rational functions $\mathbf{R}^3 \to \mathbf{R}^2$ of degree 3 (80 coefficients each)
4. Very well-posed problem: Newton's method converges in 2 iterations.

Solving the triangulation equation

Given the a matching pair $\mathbf{p} \sim \mathbf{q}$, find h such that

$$P_B(L_A(\mathbf{p}, h), h) = \mathbf{q}$$

Observations:

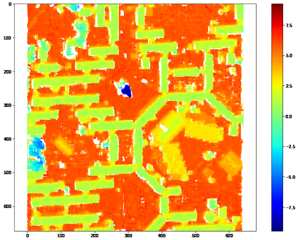
1. Two equations and one unknown: overdetermined!
2. "Define" the solution as $h = \arg\min \|P_B(L_A(\mathbf{p}, h), h) - \mathbf{q}\|^2$
3. The functions P_B and Q_A are rational functions $\mathbf{R}^3 \to \mathbf{R}^2$ of degree 3 (80 coefficients each)
4. Very well-posed problem: Newton's method converges in 2 iterations.
5. Still better: solve the linearized system $\mathrm{d}P_B \cdot \mathrm{d}L_A \begin{pmatrix} p_1 \\ p_2 \\ h \end{pmatrix} = \begin{pmatrix} q_1 \\ q_2 \end{pmatrix}$
   which has the form $\begin{pmatrix} a_1 \\ a_2 \end{pmatrix} h = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$, whose Moore-Penrose solution is $h = \dfrac{b \cdot a}{\|a\|^2}$

Dense stereo

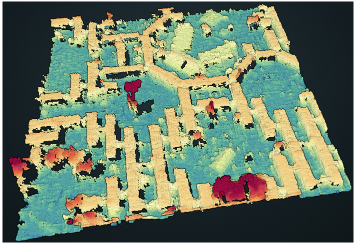
Input: two satellite images
Output: a 3D point cloud
Algorithm:

1. compute a dense field of correspondences
2. triangulate each correspondence

Input pair



Pixel correspondences



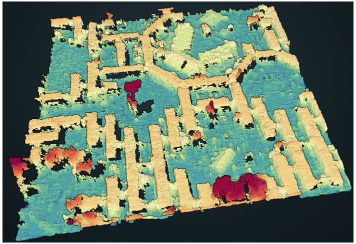
3D point cloud

Creation of a raster DEM

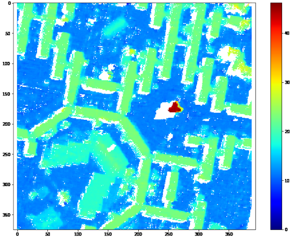
DEM = "Digital Elevation Model" an image whose pixel values represent heights

Input: a 3D point cloud
Input: a geographic grid
Output: a raster image (DEM)
Algorithm:

1. *average* all the 3D points that fall in each cell of the grid



3D point cloud



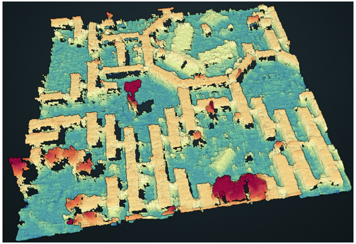
DEM

Creation of a raster DEM

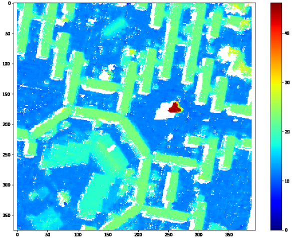
DEM = "Digital Elevation Model" an image whose pixel values represent heights

Input: a 3D point cloud
Input: a geographic grid
Output: a raster image (DEM)
Algorithm:

1. *average* all the 3D points that fall in each cell of the grid



3D point cloud



DEM

fancy DEM rendering

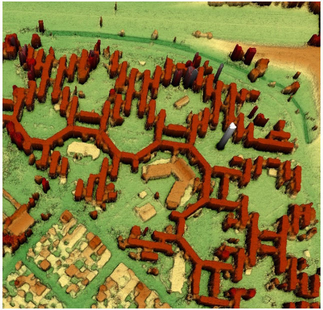
3D modeling form collections of multi-date images

3D modeling form collections of multi-date images

Challenge: Exploit the *growing* collection of satellite images

- from different satellites
- at different dates
- in different conditions

Input: multiple views

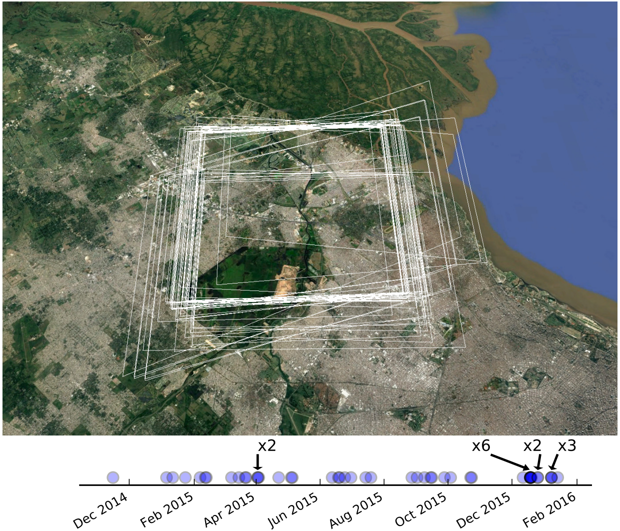


Output: 3D reconstruction

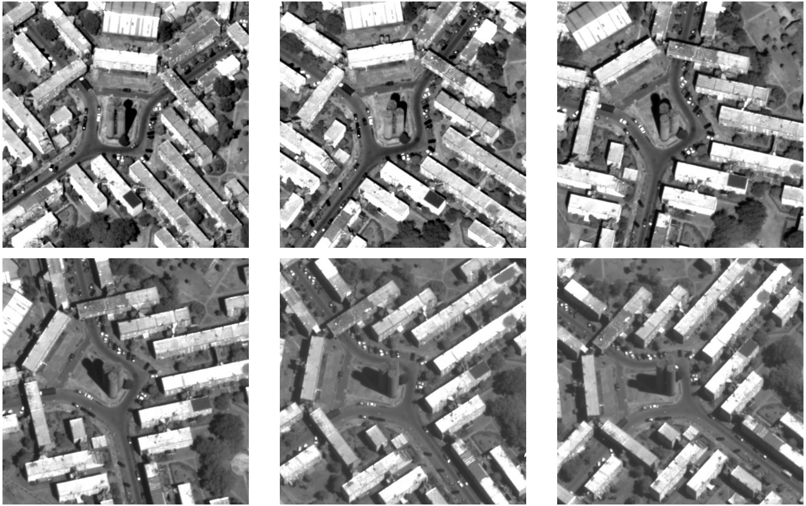
Images: WorldView3 from the MVS benchmark dataset of [Bosch et al 2016]

IARPA MVS challenge dataset

47 Worldview3 images of Buenos Aires taken over 14 months

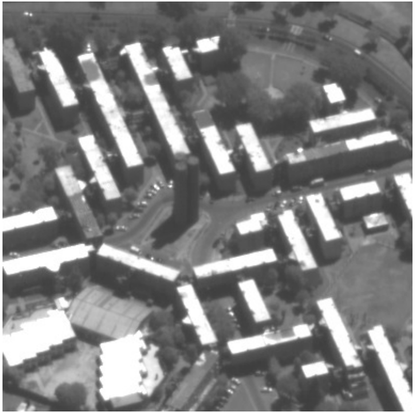


Baseline choice



The first and last images are very different, but consecutive images have a low b/h, thus are rather easy to match.

Slanted views



Notice that the images are very slanted.

Images: WorldView3 from the MVS benchmark dataset of [Bosch et al 2016]

Multi-date issue: vegetation changes

Multi-date pair, taken from a similar point of view.
Notice that the trees are different.

Images: WorldView3 from the MVS benchmark dataset of [Bosch et al 2016]

Multi-date issue: radiometric changes

Multi-date pair, taken from a similar point of view. One image is taken in winter (dark image, long shadows) and the other is taken in summer (brighter image, shorter shadows).

Images: WorldView3 from the MVS benchmark dataset of [Bosch et al 2016]

Multi-view stereo strategies

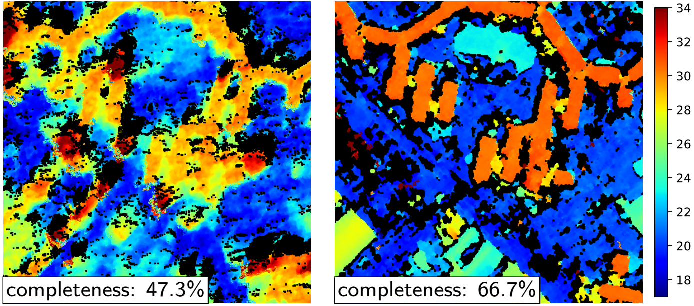
1. Traditional **bundle adjustment + multi-view.**
   Images can have very different appearances: different radiometry, changes, new structures.
   - needs tie points (not stable in multi-date)
   - solve large optimization problem with all the images

2. **Fusion of 3D models from stereo pairs.**
   Compute models independently and combine them as 3D point clouds/meshes.
   - uses geometry (more stable than tie points) to align
   - can incorporate new data without overhead
   - fusion uses statistical validation

Rationale: images may change but geometry does not

Our approach: choosing the good stereo pairs

The quality of 3D models from multi-date pairs varies wildly!
Our solution aggregates models computed from **well-chosen pairs.**

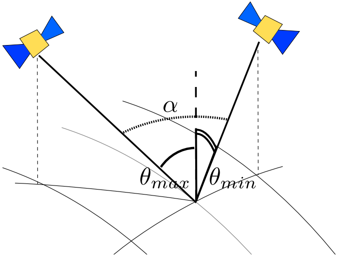


same-date pair (θ_{max}: 45°) multi-date pair (θ_{max}: 15°)

Automatic 3D Reconstruction from Multi-Date Satellite Images,
G. Facciolo, C. de Franchis, and E. Meinhardt, Earth Vision CVPRW, 2017
(Winning solution of the 2016 IARPA challenge)

Algorithm overview

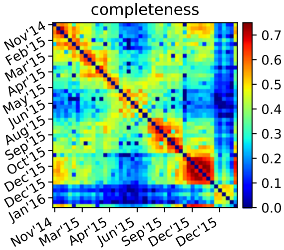
1. **Select only the "best" pairs**
   - Maximum incidence angle $\theta_{max} < 40°$
   - Angle between the views $\alpha \in [5, 45]°$
   - Temporal proximity



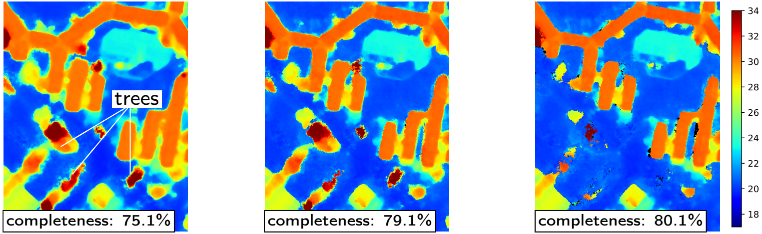
2. **Stereo matching of selected pairs**
   - Use S2P *Satellite Stereo Pipeline*
   - Triangulate and project
3. **Alignment and fusion**
   - **Align** surface models by correlation (corrects bias)
   - **Aggregate** by taking the heigh of the **lower k-medians cluster** at each pixel (removes seasonal vegetation)

Justifications

Pair selection strategy is obtained by studying all the 2162 stereo pairs.

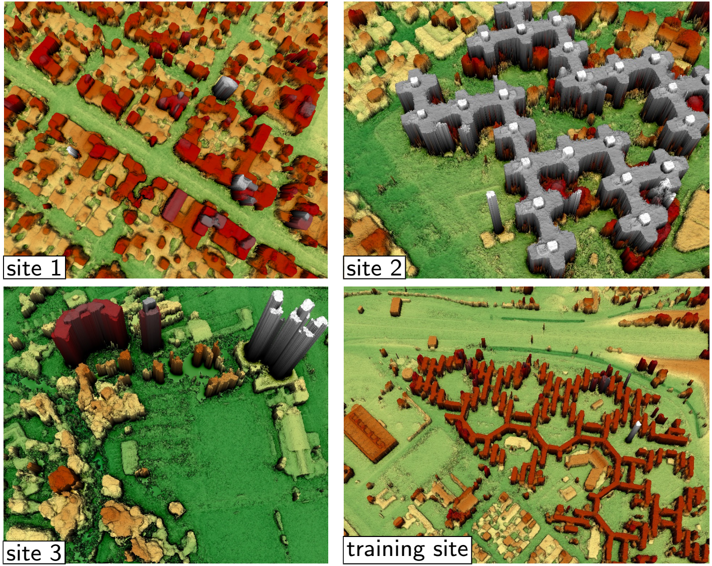


Fusion criterion



median of 700 pairs — median of 50 pairs — **k-medians of 50 pairs**

Fusion results



Conclusion

Goal of this tutorial: Get a free entrance to the community of satellite imaging.

Techniques learned:

- Use Taylor theorem to approximate all your functions to order 1
- Basic linear algebra
- Convert between different geodetic coordinate systems
- *Think globally, process your images locally*

Basis of this tutorial: Our MsC course on satellite imaging at the ENS Paris-Saclay