Analysis of Oblique Image Datasets with OrientAL

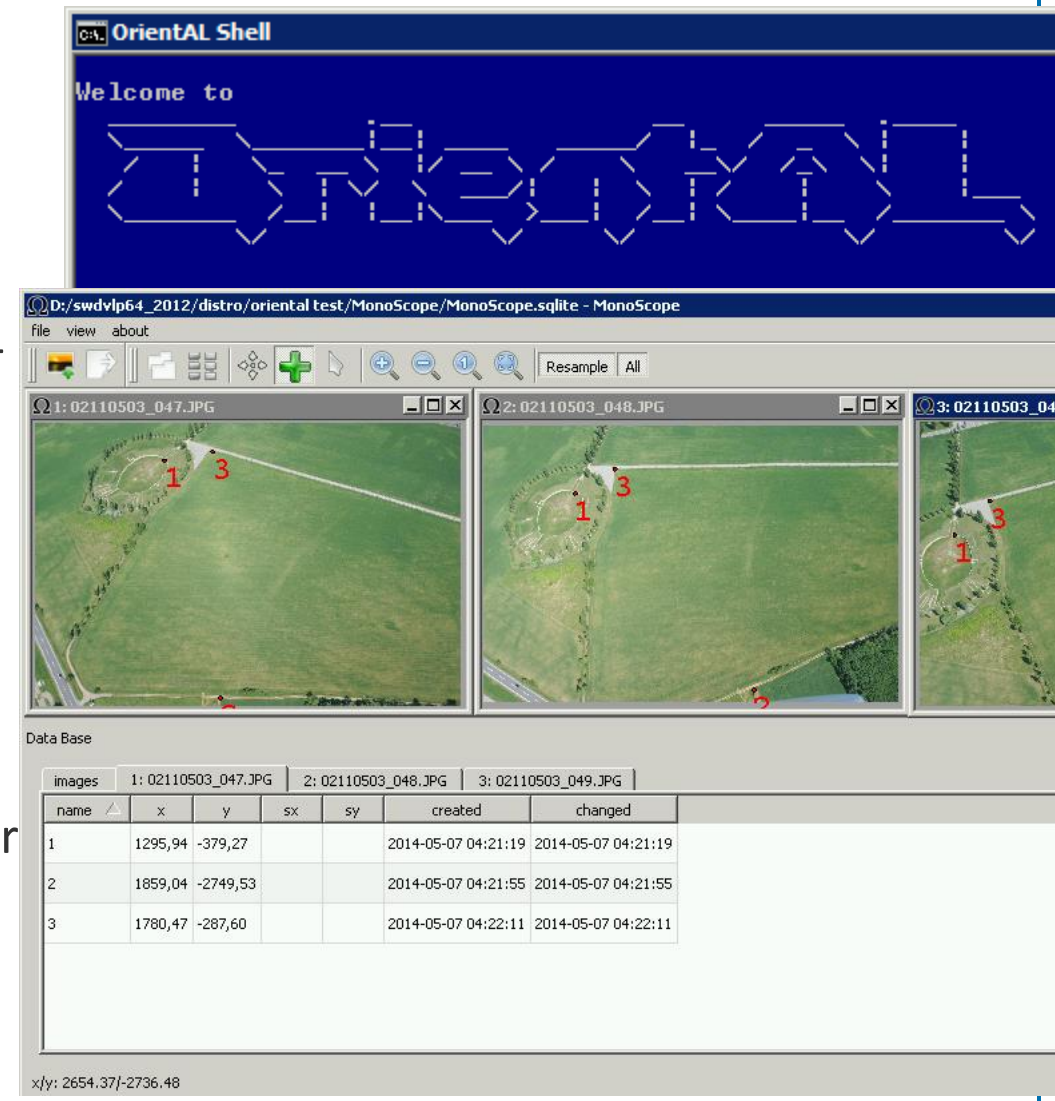
Willi Karel & Norbert Pfeifer
Wilfried.Karel@geo.tuwien.ac.at

Research Group Photogrammetry

Department of Geodesy and Geoinformation
Vienna University of Technology
http://photo.geo.tuwien.ac.at/

OrientAL

- Photogrammetric software package
- Python package: high level interface
- Multithreading, adjustment in C++ modules
- Using various 3rd party libraries (boost, OpenCV, GDAL, SQLite, Ceres, SuiteSparse, QT, …)
- Scripts for SfM, Geo-Referencing, Orthophoto generation
- Image measurement viewer/editor
- Focus on orientation, calibration



ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Feature points:
  - Detection & description:
    - oblique aerials -> use Affine SIFT
    - Extract <= 40k features w strongest response
  - Matching:
    - Symmetrically nearest
    - Threshold on ratio of first & second nearest neighbors.
    - Extract 10k best matches.
  - Match filtering:
    - RANSAC E-matrix w threshold on distance from epipolar line
    - Break up feature chains at views with multiple projections (median 4%)
- 614k auto object points, 1.42M auto image points

2	3	4	5	6	7	8	9	10	>10
486k	87k	28k	8k	3k	1.2k	517	293	156	146
79.2%	14.1%	4.6%	1.3%	0.5%	0.2%	0.08%	0.05%	0.03%	0.02%

ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Bundle block adjustment
  - Transform to local cartesian CS
  - Rigid fine-transformation to GCPs
  - Observations:
    - Manual GCP & CP image obs. (given)
    - Automatic feature point image obs.
  - Unknowns: EORs, IORs, polynomial radial distortion 3rd degree, object points
  - Robust loss function
  - Until max auto image residual < 3pix:
    - Discard object points with small maximum intersection angle
    - Discard image observations with large residuals
    - Reduce scale of robust loss function
  - Replace robust loss function with squared loss
  - 598k auto object points
  - 1.38M auto image points

ISPRS ICWG I/Vb Image Orientation Benchmark A/A

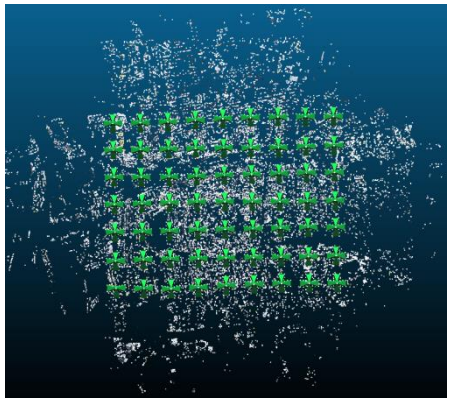
- Bundle block adjustment, cont.
- σ_0 : 0.65pix
- Image residual norms:

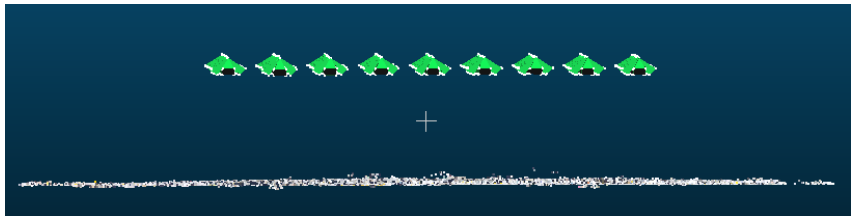
[pix]	min	median	max
Auto	0.0	0.22	3.0
GCP	0.3	1.28	2.44

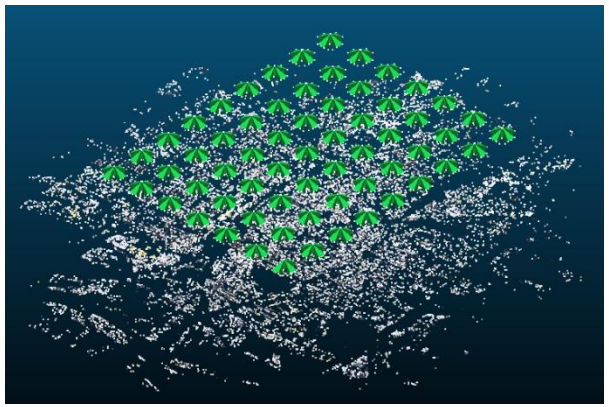
- Omega/Phi:

[gon]	min	median	max
ω	-1.00	-0.11	0.34
φ	0.09	0.14	0.70

- IORs, ADPs within reasonable bounds







ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Maxima of absolute values of correlation coefficients ρ:

Cam	x_0	y_0	z_0	r^3	x_0@	y_0@	z_0@	r^3@
163 (nadir)	0.40	0.57	0.93	0.11	$z_{0,148}$	$z_{0,159}$	Z_0	φ
148 (front)	0.51	0.99	0.99	0.12	κ	$z_{0,148}$	$y_{0,148}$	ω
145 (right)	0.32	1.00	1.00	0.16	φ	$z_{0,145}$	$y_{0,145}$	φ
147 (back)	0.55	0.99	0.99	0.09	κ	$z_{0,147}$	$y_{0,147}$	ω
159 (left)	0.29	0.99	0.99	0.09	φ	$z_{0,159}$	$y_{0,159}$	φ

ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Object space is mostly flat
- All nadir images share the same κ
  - z_0 and Z_0 highly correlated
- Same rotation about optical axis for oblique images
  - y_0, z_0 of each camera fully correlated
- Do not estimate IOR, but use lab calibration!

ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Constant IOR:
  - σ_0 : 0.65pix (same)
  - Image residual norms (smaller for GCPs!):

[pix]	min	median	max
Auto	0.0	0.22	3.0
GCP	0.26	0.91	1.74

	163 (nadir)	148 (front)	145 (right)	147 (back)	159 (left)
r^3	0.853	-0.004	0.365	0.196	0.317
σ_{r3}	0.07	0.09	0.09	0.08	0.09
$\lvert r^3 \rvert / \sigma_{r3}$	12.7	0.0	4.0	2.6	3.7

ISPRS ICWG I/Vb Image Orientation Benchmark A/A

- Constant IOR:
  - EOR std.devs.: ≈ 5cm, ≈ 3 mgon
  - Median relative offsets of oblique cameras & σ_{MAD}
  - Inconsistent
  - Mostly insignificant

	148 (front)	**145 (right)**	**147 (back)**	**159 (left)**
med_X	0.082	-0.191	-0.120	-0.090
med_Y	-0.124	0.006	-0.009	-0.175
med_Z	-0.399	-0.296	-0.154	-0.265
$\sigma_{MAD,X}$	-0.21	0.11	0.13	0.14
$\sigma_{MAD,Y}$	-0.12	0.08	0.14	0.10
$\sigma_{MAD,Z}$	-0.09	0.64	0.11	0.07

Fast & Efficient Decimation of Tie Points

- For large bocks, too many tie points may prohibit/hinder the adjustment due to limited RAM
- Most (e.g. 80% for benchmark) tie points observed in only 2 views
- 2-folds should not simply be discarded, because:
  - Images at block outline need them
  - Weakly textured areas
- Project:
  - 5-camera rig
  - 42k images, 4.9 µm per pixel, 25 / 50 mm focal length
  - GSD 12cm, 70%/60% overlap, 600m flying height, nadir footprint ca. 600 x 900 m^2
  - GPS/INS inaccurate, cams not synchronized
- Features detected/matched in sub-blocks using Pix4D
- 1.03M tie points - too many to estimate all EORs, IORs, ADPs at once.
- How many are really necessary, and which ones?

Fast & Efficient Decimation of Tie Points

- Decimate tie points in object space
  - Grid overlay in X,Y-plane, cell size 100m
  - For each cell, select the tie point detected in most images (multiplicity)
  - Discard all other tie points
  - 85k tie points (8%), 613 images non-orientable, all oblique.
- Decimate in object space for each camera separately
  - 427k (42%) tie points, 50 images non-orientable, all oblique.
- Decimate in image space
  - Grid overlay in x,y-plane of each camera, 5x3 grid
  - For each camera
    - For each cell
      - If cell is empty, select tie point with highest multiplicity
      - Insert tie point into resp. cells of other cameras, if visible

Fast & Efficient Decimation of Tie Points

- Decimate in image space, cont.
  - 144k (14%) tie points, all images orientable

	1 (nadir)	2 (oblique)	3 (oblique)	4 (oblique)	5 (oblique)
min	14	15	14	15	8
median	101	34	37	34	36
max	296	138	158	138	148

- Tie points with high multiplicity mostly in nadir images!
- Decimation in object space is steered by nadir images.

Outlook

- Benchmark dataset:
  - Investigate relative orientation of cameras on the rig
- Tie point decimation:
  - Investigate effect of decimation on precision of estimated parameters
  - How much decimation can be done without affecting them?

Thanks for your attention!