DGAP Notes

Dirk Stallmann

March 15, 2008
DGAP Version 3.2.0

Contents

Chapter 1

Introduction

The program DGAP[1] implements the photogrammetric method of bundle adjustment invented by Helmut Schmid and Duane Brown. Based on the central perspective the geometric relationship between the image and the object are determined by the least squares method. The method is also known as "bundle block adjustment" or shorter "bundle adjustment".

[1]= "Dirks General Analytical Positioning". DGAP is closely related to the bundle program "General Analytical Positioning" (GAP) by Horst A. Beyer [Beyer, 1992] and the batch-version "Scott's GAP" (SGAP) by Scott O. Mason [Mason, 1994] (Institute of Geodesy and Photogrammetry, Swiss Federal Institute of Technology Zurich). The development of DGAP starts with the simpilyfied reimplementation of GAP in the programing language C++. The mathematical approach, the in- and output and many options and parameters are borrowed from GAP.

Chapter 2

Program In- and Output

The program in- and output is done by textfiles in DOS or UNIX format. Filename extensions are not used by the program but can be helpful to indicate the file contents.

The textfiles consisting of several input fields. The input fields are separated by white spaces[1] and are interpreted depending on the expected data type. The four data types are following rules for C++/STL data types:

Data type	C++/STL type	Example
boolean value	bool	`0 1`
integer numbers	int	`42 -4711`
double precision numbers	double	`-123.4 1.0e+31`
string	char[], string	`abc AP-set stereo_a`

The reading of the data types following the C/C++-rules: Binary numbers are represented by `0` (off) and `1` (on). Strings cannot contain white spaces. Therefore use the underscore `_` or the minus-character `-` to seperate the words. For a better distinction between keywords and names it is recomended to use the underscore in names.

2.1 Configuration/Project File

The configuration or project file (.cfg or .prj) integrate all input files.

File structure:

```
<keyword> <filename>
...
[stop-dep]
```

`<keyword>` defines the expected type of the input file given by `<filename>`. The keyword `stop-dep` at the end of file is optional. Any following information after `stop-dep` will be ignored.

For the input files prefer the following order:

1. parameters/options
2. cameras

[1]Standard C white spaces: blank, tabulator, vertical tabulator, newline, formfeed, carridge return.

3. image connections
4. orientation parameters, object points, additional parameters etc.

Keyword	File type
`params`	parameters/options
`ls-params`	least-squares parameters/options
`network-design-params`	network design parameters/options
`camera`	camera data
`camera-ads`	ADS40 camera data
`image`	image connections
`ext-ori`	orientation parameters
`orientation-data-params`	orientation data parameters/options for line scanner
`orientation-data`	orientation data for line scanner
`orientation-data-odf`	ADS40 orientation data (ODF-format)
`image-crds`	observed image coordinates
`obj-crds`	object coordinates
`ctrl-supp`	control support
`ap-set`	additional parameters
`pos-cor`	position correction parameters
`att-cor`	attitude correction parameters
`imu-misal`	IMU misalignment parameters
`stop-dep`	end of data

Table 2.1: Keywords: configuration/project file

Listing 2.1: Example test.cfg

```
params          test.pr
ls-params       test.lsp
camera          test.ca
image           test.im
image-crds      test.ic
ext-ori         test.eo
obj-crds        test.oc
stop-dep
```

2.2 Parameters/Options File

The parameters/options file (.pr) contains parameters and options for printing, output and debugging.

File structure:

```
<pr_chk_appr> <pr_chk_rays> <pr_inter> <pr_ic>
<pr_obj_crds> <pr_ext_ori> <pr_ap_sets> <pr_pos_cor> <pr_att_cor> <pr_imu_mis>
<db_upari> <db_inter>
<pr_aps_nx> <pr_aps_ny>
<fname_normal_matrix_struct>
<fname_normal_matrix>
```

```
<fname_right_side_vector>
<fname_solution_vector>
```

Parameter	Data type	Description	Default value
`<pr_chk_appr>` `<pr_chk_rays>` `<pr_inter>` `<pr_ic>`	bool	Print the check of approximations of object point. Print number of rays per object point. Print intersection results. Print observed image coordinates.	0 0 0 0
`<pr_obj_crds>` `<pr_ext_ori>` `<pr_ap_sets>` `<pr_pos_cor>` `<pr_att_cor>` `<pr_imu_mis>`	bool	Print unknown parameters during the iterations: object coordinates, orientation parameters, additional parameters, position correction, attitude correction and IMU-misalignment	0 0 0 0 0 0
`<db_upari>` `<db_inter>`	bool	Print UPARI-table. Print more information about the intersection.	0 0
`<pr_aps_nx>` `<pr_aps_ny>`	int	Print the effects of additional parameters. Set the number of grid points in x and y directions.	0 0
`<fname_normal_matrix_struct>`	string	Output filename of normal equation structure	none
`<fname_normal_matrix>`	string	Output filename of normal equation matrix	none
`<fname_right_side_vector>`	string	Output filename of right side vector	none
`<fname_solution_vector>`	string	Output filename of solution vector	none

Table 2.2: Parameters/options file

Using the word `none` to avoid the output in a file.

Listing 2.2: Example test.pr

```
0 0 1 1
0 1 0 0 0 0
0 0
0 0
none
none
none
none
```

2.3 Least-Squares Parameters/Options File

The settings in the least-squares parameters/options file effects the least-squares adjustment.

File structure:

```
<sigma0>
<max_iter>
<conv_chk> <conv_eps>
<chk_obj> <chk_pcc> <chk_rot>
<conv_obj> <conv_pcc> <conv_rot>
<smin> <smax>
<smin_u> <smax_u>
<unksup_wt> <constr_wt>
<ccoef_lim>
<incr_crd> <incr_rot>
<t_quantil>
<atpv_lim>
<res_lim>
<unit_objc> <unit_angle>
<adj_interface>
<ap_derivs>
```

Parameter	Data type	Description	Default value
`<sigma0>`	double	A priori standard deviation of unit weight in m	10^{-6}
`<max_iter>`	int	Maxium number of iterations	10
`<conv_chk> <conv_eps>`	bool	General convergence option and convergence limit	1 10^{-16}
`<chk_obj> <chk_pcc> <chk_rot>`	bool	Special convergence options for object points, projection centres and rotation angles	0 0 0 0
`<conv_obj> <conv_pcc> <conv_rot>`	double	Convergence limit for object points, projection centres and rotation angles in m resp. rad	0.001 0.001 10^{-6}
`<smin> <smax>`	double	Lower and upper limit of standard deviations σ_{min} and σ_{max} for observed parameters	10^{-30} 10^{+30}
`<smin_u> <smax_u>`	double	Standard deviations for unknown resp. eliminated/fixed parameters $\sigma_{min,u}$, $\sigma_{max,u}$	10^{-31} 10^{+31}
`<unksup_wt> <constr_wt>`	double	Weights for elimination of unknown parameters and for contraints	10^{-10} 10^{-10}
`<ccoef_lim>`	double	Output limit for correlations coefficents	0.8
`<incr_crd> <incr_rot>`	double	Increments for numerical derivatives for object point and rotation parameters in m resp. rad (unused)	10^{-0} 10^{-3}
`<t_quantil>`	double	Quantil of the t-distribution $t_{\infty,1-\alpha/2}$ for test of significance with the error propability α	1.96
`<atpv_lim>`	double	Limit for the adjustment test $\mathbf{A}^{\mathrm{T}}\mathbf{P}\mathbf{v} = \mathbf{0}$	10^{-4}
`<res_lim>`	double	Limit for the output of residuals of image coordinates in m	10^{-5}
`<unit_objc> <unit_angle>`	string	Keyword for units for printing object coordinates and rotation parameters	`m rad`
`<adj_interface>`	string	Keyword of the method for solving the normal equations. See table 2.6	`native`

continued on next page

<ap_derivs>	string	Keyword of the method for building the partial derivatives of the additional parameters. See table 2.7	`image-coords-plus-aps`

Table 2.3: Least-squares parameters/options

Keyword	Unit
`um`	micrometer
`mm`	millimeter
`cm`	centimeter
`m`	meter

Table 2.4: Keywords: unit lengths

Keyword	Unit
`rad`	radiant
`deg`	degree
`gon`	gon/grads

Table 2.5: Keywords: angular units

Keyword	Method
`native`	GenLib2-library
`lapack`	LAPACK/BLAS-library

Table 2.6: Keywords: methods for solving the normal equations

Keyword	Method
`image-coords`	from observed image coordinates
`collinear-equation`	from the collinear equation
`image-coords-plus-aps`	from observed image coordinates and additional parameters

Table 2.7: Keywords for methods to build the partial derivatives of additional parameters

Listing 2.3: Example test.lsp

```
6.0e-6
10
0   1e-16
0   0   1
0.001   0.001   1.57e-6
1e-30   1e+30
1e-31   1e+31
1e+10   1e+10
0.80
1e0   1e-3
1.96
1e-4
```

```
0.
m   gon
lapack
image-coords-plus-aps
```

2.4 Network Design Parameters/Options File

Parameters and options related to the photogrammetric network design.

File structure:

```
<min_rays_point>
<min_rays_image>
<pr_freq_rays>
[stop-dep]
```

Parameter	Data type	Description	Default value
<min_rays_point>	int	Minimal number of rays per object point	2
<min_rays_image>	int	Minimal number of rays per image	3
<pr_freq_rays>	bool	Print the absolute frequency of rays per object point	0

Table 2.8: Network design parameters/options

Listing 2.4: Example test.pnp

```
2   3
1
```

2.5 Camera File

The input format depends on the camera type:

Keyword	Camera type	Image coordinate system
`camera-frame`	frame camera	metric
`camera-ccd`	digital camera	pixel
`camera-pushbroom`	pushbroom camera	pixel
`camera-pushbroom-calib`	calibrated pushbroom camera	pixel

Table 2.9: Keywords: camera type

File structure for frame camera:

```
camera-frame
<camera_id> <camera_name>
```

```
<c> <xp> <yp>
<sx> <sy>
```

Parameter	Data type	Description	Unit
<camera_id>	string	Camera ID to identify the camera	-
<camera_name>	string	Camera name or comment	-
<c>	double	Camera constant c	mm
<xp> <yp>	double	Principal point coordinates x_p, y_p	mm
<sx> <sy>	double	Image size in s_x and s_y-direction	mm

Table 2.10: Frame camera parameters

Listing 2.5: Beispiel test.ca

```
camera-frame
101   RMK
150.000   0.000   0.000
230.   230.
stop-dep
```

File structur for digital cameras:

```
camera-ccd
<camera_id> <camera_name>
<c> <xp> <yp>
<nc> <nr>
<sc> <sr>
```

Parameter	Data type	Description	Unit
<camera_id>	string	Camera ID to identify the camera	-
<camera_name>	string	Camera name or comment	-
<c>	double	Camera constant c	mm
<xp> <yp>	double	Principal point coordinates x_p, y_p	mm
<nc> <nr>	int	Number of columns and rows of the sensor	Pixel
<dc> <dr>	double	Pixel size/spacing in column und row direction Δc, Δr	µm/Pixel

Table 2.11: Digital camera parameters

File structure for pushbroom cameras:

```
camera-pushbroom
<camera_id> <camera_name>
<c> <xp> <yp>
<nc>
<dc> <dr>
<x0>
```

Parameter	Data type	Description	Unit
<camera_id>	string	Camera ID to identify the camera	-
<camera_name>	string	Camera name or comment	-
<c>	double	Camera constant c	mm
<xp> <yp>	double	Principal point coordinates x_p, y_p	mm
<nc>	int	Number of pixel in sensor line	Pixel
<dc> <dr>	double	Pixel size/spacing in column und row direction Δc, Δr	µm/Pixel
<x_0>	double	x_0-position of the sensor line	mm

Table 2.12: Pushbroom cameras

File structure for calibrated pushbroom cameras:

```
camera-pushbroom-calib
<camera_id> <camera_name>
<c>
<nc>
<dc> <dr>
<x_cal> <y_cal>
...
```

Parameter	Data type	Description	Unit
<camera_id>	string	Camera ID to identify the camera	-
<camera_name>	string	Camera name or comment	-
<c>	double	Camera constant c	mm
<xp> <yp>	double	Principal point coordinates x_p, y_p	mm
<nc>	int	Number of pixel in sensor line	Pixel
<dc> <dr>	double	Pixel size/spacing in column and row direction Δc, Δr	µm/Pixel
<x_cal> <y_cal>	double	Calibrated pixel position for every pixel in the sensor line	mm

Table 2.13: Calibrated pushbroom camera parameters

2.6 Image Connections File

The entries in the image connections file (.im) determines the relation between the observed image coordinates, the orientation parameters respectifly the camera. Furthermore the keyword specify the used mathematical model:

- photogrammetric model,
- extended photogrammetric model,
- DLT,

- affin transformation or
- Direct GeoReferencing (DGR) for pushbroom cameras.

The mathematical models are only valid within special combinations of orientation parameters and camera types:

Keyword	Model	Orientation parameter	Camera type
`image-frame`	photogrammetric	`ext-ori-xxx`	`camera-frame`, `camera-ccd`
`image-frame-ext`	extended photogrammetric	`ext-ori-xxx`	`camera-frame`, `camera-ccd`
`image-dlt`	Direct Linear Transformation	`dlt-params`	`camera-frame`, `camera-ccd`
`image-affine-transf`	affine transformation	`affine-transf-params`	`camera-frame`, `camera-ccd`
`image-scanner`	DGR for line scanner	Orientation data set	`camera-pushbroom`, `camera-pushbroom-calib`, `camera-ads`

Table 2.14: Keywords: mathematical model

File structure:

```
<image_type> <image_id> <station_id> <camera_id>
...
[stop-dep]
```

Parameter	Data type	Description
<image_type>	string	Mathematical model. Keyword see table 2.14
<image_id>	string	Image ID to identify the image
<station_id>	string	Station ID to identify the station
<camera_id>	string	Camera ID to identify the camera

Table 2.15: Image connections

Listing 2.6: Example test.im

```
image-frame   1   1   101
image-frame   2   2   101
stop-dep
```

2.7 Image Coordinates File

File structure for image coordinates with individual standard deviation:

```
indiv-sdev
<image_id> <scale>
```

```
<point_id> <x> <y> <sx> <sy>
...
stop-dep
```

File structure for image coordinates with common standard deviation:

```
common-sdev <sx> <sy>
<image_id> <scale>
<point_id> <x> <y>
...
stop-dep
[stop-dep]
```

File structure for pixel coordinates with individual standard deviation:

```
indiv-sdev
<image_id> <off_x> <off_y>
<point_id> <x> <y> <sx> <sy>
...
stop-dep
```

File structure for pixel coordinates with common standard deviation:

```
common-sdev <sx> <sy>
<image_id> <off_x> <off_y>
<point_id> <x> <y>
...
stop-dep
[stop-dep]
```

Parameter	Data type	Description	Unit
`<image_id>`	string	Image ID	-
`<off_x> <off_y>`	double	Offset (only if pixel coordinates)	pixel
`<scale>`	double	Scale factor (only if image coordinates)	none
`<point_id>`	string	Point ID	-
`<x> <y>`	double	Image or pixel coordinates	m or pixel
`<sx> <sy>`	double	Standard deviation thereof	m or pixel

Table 2.16: Image/pixel coordinates

Listing 2.7: Example test.ic

```
indiv-sdev
1   1e-3
  11   16.012   79.963   0.006   0.006
  13   88.560   81.134   0.006   0.006
  31   13.362  -79.370   0.006   0.006
  33   82.240  -80.027   0.006   0.006
```

```
  12   51.758   80.555   0.006   0.006
  21   14.618   -0.231   0.006   0.006
  22   49.880   -0.782   0.006   0.006
  23   86.140   -1.346   0.006   0.006
  32   48.035  -79.962   0.006   0.006
stop-dep

2   1e-3
  11  -73.930   78.706   0.006   0.006
  13   -5.252   78.184   0.006   0.006
  31  -79.122  -78.879   0.006   0.006
  33   -9.887  -80.089   0.006   0.006
  12  -39.953   78.463   0.006   0.006
  21  -76.006    0.036   0.006   0.006
  22  -42.201   -1.022   0.006   0.006
  23   -7.706   -2.112   0.006   0.006
  32  -44.438  -79.736   0.006   0.006
stop-dep
stop-dep

4711    0.035    0.962   0.006   0.006
```

2.8 Orientation/Transformation Parameters File

File structure for exterior orientation parameters using rotation angles about the three axes:

```
indiv-type
<eo_type>
<station_id> <uangle> <time>
<Xo> <Yo> <Zo>
<sXo> <sYo> <sZo>
<a1> <a2> <a3>
<sa1> <sa2> <sa3>
...
[<stop-dep>]
```

with `eo_type = {ext-ori-opk-rot | ext-ori-pok-rot | ext-ori-opk-fix | ext-ori-australis}.`

File structure for exterior orientation parameters using unit quaternion:

```
indiv-type
ext-ori-quaternion
<station_id> <time>
<Xo> <Yo> <Zo>
<sXo> <sYo> <sZo>
<q0> <q1> <q2> <q3>
<sq0> <sq1> <sq2> <sq3>
...
[<stop-dep>]
```

File structure for DLT parameters:

```
indiv-type
dlt-params <station_id>
<L1> <sL1>
<L2> <sL2>
...
<L11> <sL11>
...
[<stop-dep>]
```

File structure for affine transformation parameters:

```
indiv-type
affine-transf-params <station_id>
<A1> <sA1>
<A2> <sA2>
...
<A8> <sA8>
...
[<stop-dep>]
```

File structure for exterior orientation parameters with common type:

```
common-type
<eo_type> <unit_angle> <time_flag> <matrix_flag> <sXo> <sYo> <sZo>
<sa1> <sa2> <sa3>
```

If `<matrix_flag> = 0` the rotation parameters are expected:

```
<station_id> [<time>] <Xo> <Yo> <Zo> <a1> <a2> <a3>
...
[<stop-dep>]
```

If `<matrix_flag> = 1` the elements of the rotation matrix are expected:

```
<station_id> [<time>] <Xo> <Yo> <Zo> <r11> ... <r33>
...
[<stop-dep>]
```

Parameter	Data type	Description	Unit
`<eo_type>`	string	Parametrisation with rotation angles. Keywords see table 2.18	-
`<uangle>`	string	Angular unit of rotation angles. Keywords see table 2.5	-
`<time_flag>`	bool	Flag if (GPS-)time is not given (0) or given (1)	-

continued on next page

<matrix_flag>	bool	Flag if rotation angle (0) or rotation matrix (1) given: <r11>, <r12>, ... <r33>	-
<station_id>	string	Station ID to identify the set of orientation/-transformation parameters	-
<time>	double	(GPS-)time	s
<Xo> <Yo> <Zo>	double	Projection centre coordinates	m
<sXo> <sYo> <sZo>	double	Standard deviations of projection centre coordinates	m
<a1> <a2> <a3>	double	Rotation angles	<uangle>
<sa1> <sa2> <sa3>	double	Standard deviations of rotation angles	<uangle>
<q0> <q1> <q2> <q3>	double	Quaternion elements	1
<sq0> <sq1> <sq2> <sq3>	double	Standard deviations of quaternion elements	1
<L1> ... <L11>	double	DLT parameters	m bzw. 1
<sL1> ... <sL11>	double	Standard deviations of DLT parameters	m bzw. 1
<A1> ... <A8>	double	Affine transformation parameters	m bzw. 1
<sA1> ... <sA8>	double	Standard deviations of affine transformation parameters	m bzw. 1

Table 2.17: Orientations/transformation parameters

Keyword	Parametrisation	Number of parameters
ext-ori-opk-fix	Rotation angles in sequence ω-φ-κ about fixed axes	3
ext-ori-opk-rot	Rotation angles in sequence ω-φ-κ about rotated axes	3
ext-ori-pok-rot	Rotation angles in sequence φ-ω-κ about rotated axes	3
ext-ori-australis	Rotation angles in sequence α-ν-κ	3
ext-ori-quaternion	Unit quaternion	4

Table 2.18: Keywords: parametrisation of the rotation matrix

Listing 2.8: Example test.eo

```
indiv-type

ext-ori-pok-rot   1   gon   0.
5000.765    5027.343    1997.357
   1e+31       1e+31       1e+31
 0.00000     0.00000     6.56071
   1e+31       1e+31       1e+31

ext-ori-pok-rot   2   gon   0.
5918.253    5134.023    2022.014
   1e+31       1e+31       1e+31
 0.00000     0.00000     6.89616
   1e+31       1e+31       1e+31
```

```

stop-dep
```

2.9 Object Coordinate File

File structure by individual standard deviations:

```
indiv-sdev
<point_id> <Xo> <Yo> <Zo> <sX> <sY> <sZ>
...
[stop-dep]
```

File structure by common standard deviations:

```
common-sdev
<sX> <sY> <sZ>


<point_id> <Xo> <Yo> <Zo>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
<point_id>	double	Point ID to identify the object point	-
<X> <Y> <Z>	double	Object point coordinates	m
<sX> <sY> <sZ>	double	Standard deviation of the object point coordinates	m

Table 2.19: Object coordinates

Listing 2.9: Example test.oc

```
indiv-sdev
  11   5083.205   5852.099    527.925   1e-31   1e-31   1e-31
  13   5780.020   5906.365    571.549   1e-31   1e-31   1e-31
  31   5210.879   4258.446    461.810   1e-31   1e-31   1e-31
  33   5909.264   4314.283    455.484   1e-31   1e-31   1e-31

  12   5430.762   5877.971    504.192   1e+31   1e+31   1e+31
  21   5145.744   5040.026    504.192   1e+31   1e+31   1e+31
  22   5495.459   5070.680    504.192   1e+31   1e+31   1e+31
  23   5855.069   5102.227    504.192   1e+31   1e+31   1e+31
  32   5558.274   4284.780    504.192   1e+31   1e+31   1e+31

stop-dep

 111   5550.000   4284.000    504.000   1e+31   1e+31   1e+31
```

2.10 Control Support File

Use the Control Support (CS) file (.cs) to overwrite the standard deviations and declare this point to be a check point.

File structure:

```
[#] <point_id> <cs> <cs_in>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
`<point_id>`	string	Point ID to identify the object point	-
`<cs>`	string	CS type. Keywords see table 2.21	-
`<cs_in>`	string	CS in X/Y/Z-coordinate: `x` = X-, `y` = Y-, `z` = Z-coordinate	-

Table 2.20: Control Support

Keyword	Object point types
`in`	new or intersection point
`co`	control point
`ch`	check point

Table 2.21: Keywords: object point types

Listing 2.10: Example test.cs

```
  11   co   xyz
  13   co   xyz
  31   ch   xyz
  33   co   xyz
stop-dep
```

2.11 Additional Parameter File

File structure for physical models:

```
<ap_set_type>
<ap_set_id>
<camera_id>
...
stop-dep
<ap> <sap>
...
[stop-dep]
```

File structure for polynominal models:

```
<ap_set_type>
<ap_set_id>
<camera_id>
...
stop-dep
<b>
<ap> <sap>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
`<ap_set_type>`	string	Additional parameter model. Keywords see table 2.23	-
`<ap_set_id>`	string	ID to identify the set of parameters	-
`<b>`	double	Base in image space. Only for the polynomial models `p12` and `p44`.	mm
`<camera_id>`	string	Camera ID to identify the connected camera	-
`<ap> <sap>`	double	Coefficients and standard deviation of the additional parameter. The number of parameters depends on the selected model (see table 2.23)	

Table 2.22: Additional parameters

Keyword	Model	Number of Params.	Parameters
`inner-or`	inner orientation	3	Δx_p, Δy_p, Δc
`radial-dist`	radial distortion	3	K_1, K_2, K_3
`decentering-dist`	tangential distortion	2	P_1, P_2
`in-plane`	in-plane distortion	12	$a_1, a_2, \ldots, a_{12}$
`out-of-plane`	out-of-plane distortion	3	a_{13}, a_{14}, a_{15}
`brown-eqn-22`	Browns model eqn. 22	29	Δx_p, Δy_p, Δc, K_1, K_2, K_3, P_1, P_2, $a_1, a_2, \ldots, a_7$, $b_1, b_2, \ldots, b_7$, $c_1, c_2, \ldots, c_7$
`brown-eqn-23`	Browns model eqn. 23	21	Δx_p, Δy_p, Δc, K_1, K_2, K_3, $a_1, a_2, \ldots, a_{15}$
`p12`	Ebners orthogonal polygon model	12	$a_1, a_2, \ldots, a_{12}$
`p44`	Gruens orthogonal polygon model	44	$a_1, a_2, \ldots, a_{44}$
`gap`	GAP implementation	10	Δx_p, Δy_p, Δc, s_x, a_1, K_1, K_2, K_3, P_1, P_2

continued on next page

`australis`	Australis implementation	10	Δx_p, Δy_p, Δc, K_1, K_2, K_3, P_1, P_2, b_1, b_2

Table 2.23: Keywords: additional parameter sets

Listing 2.11: Example test.ap

```
australis
ap_set1
cam1
stop-dep
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
0.0   1e+31
stop-dep
```

Listing 2.12: Example test2.ap

```
inner-or
subset_inner
cam1
stop-dep
0.0   1e+31
0.0   1e+31
0.0   1e+31

set_io
radial-dist
subset_radial
cam1
stop-dep
0.0   1e+31
0.0   1e+31
0.0   1e+31

decentering-dist
subset_decenter
cam1
stop-dep
0.0   1e+31
0.0   1e+31

in-plane-dist
subset_in_plane
cam1
```

```
stop-dep
0.0   1e+31
0.0   1e+31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31
0.0   1e-31

stop-dep
```

2.12 Position Correction Fie

File structure:

```
<pos_cor_id>
<image_id>
...
stop-dep
<num_coeff>
<coeff> <sdev>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
<pos_cor_id>	string	Positions correction ID	-
<image_id>	string	Image ID of connected image	-
<num_coeff>	int	Number of polynom coefficientes	-
<coeff> <sdev>	double	Polynom coefficient and associated standard deviation	-

Table 2.24: Position correction

2.13 Attitude Correction File

File structure:

```
<att_cor_id>
<image_id>
...
stop-dep
<num_coeff>
```

```
<coeff> <sdev>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
<att_cor_id>	string	Attitude correction ID	-
<image_id>	string	Image ID of connected image	-
<num_coeff>	int	Number of polynom coefficientes	-
<coeff> <sdev>	double	Polynom coefficient and associated standard deviation	-

Table 2.25: Attitude correction

2.14 IMU Misalignment File

File structure for parametrisation using rotation angles:

```
<type>
<image_id>
...
stop-dep
<matrix_flag> <uangle>
<a1> <a2> <a3>
<sa1> <sa2> <sa3>
...
[stop-dep]
```

Parameter	Data type	Description	Unit
<type>	string	Parametrisiation of the rotation matrix. Keyword see table 2.27	-
<image_id>	string	Image ID of connected image	-
<uangle>	string	Angular unit for rotation angle. Keyword see table 2.5	-
<matrix_flag>	bool	Flag if rotation matrix (0) or rotation matrix (1) given: <r11>, <r12>, ... <r33>	-
<a1> <a2> <a3>	double	Rotation angles	<uangle>
<sa1> <sa2> <sa3>	double	Standard deviation of the rotation angles	<uangle>

Table 2.26: IMU misalignment

Keyword	Parametrisation	Number of Parameters
angles-opk-fix	Rotation angles with sequence ω-φ-κ about fixed axes	3

continued on next page

angles-pok-fix	Rotation angles φ-ω-κ about fixed axes	3
angles-opk-rot	Rotation angles ω-φ-κ about rotated axes	3
angles-pok-rot	Rotation angles φ-ω-κ about rotated axes	3
ext-ori-australis	Rotation angles α-ν-κ	3
unit-quaternion	elements of unit quaternion q_0, q_1, q_2, q_3	4
rotation-matrix	Rotationsmatrix $r_{11}, r_{12}, \ldots, r_{33}$	9

Table 2.27: Keywords: parametrisation of the rotation matrix

2.15 Leica ADS40

ADS40 camera files and orientation data in ODF-format [Leica, 2002] can be used directly.

The filenames (`<ads-filename>` , `<odf-filename>`) and IDs (`<camera_id>` , `<ods_id>`) have to specified in the configuration file. File structure:

```
camera-ads <ads-filename> <camera_id>
...
orientation-data-odf <odf-filename> <ods_id>
...
[stop-dep]
```

2.16 Orientation Data Set Parameters/Options File

Special settings common to all orientation data sets.

File structure:

```
<shift>
<interpol>
```

Parameter	Data type	Description	Default value
<shift>	double	globale shift Δr for all orientation data sets (in rows)	0.
<interpol>	string	Keyword for the interpolation method: trunc = round to integer value by truncation of the fractional part, linear = linear interpolation	linear

Table 2.28: Orientation data set parameters/options

Chapter 3

Running dgap

To use DGAP call dgap from the command line:

```
dgap <config.cfg>
```

where <config.cfg> is the name of the configuration file. By calling the program without arguments or with the option -help the usage text (see listing 3.1) is shown.

Listing 3.1: DGAP usage text

```
Usage: dgap [options] cfg_file [stem]
Arguments:
  cfg_file          Configuration file.
  stem              Stem name for output files.
Options:
  -chk_in           Check input only.
  -corr             Evaluate correlation coeffs. of parameters.
  -corr_matrix      Evaluate correlation coeffs. of parameters.
  -covar            Print covariances of parameters.
  -db_upari         Print upari-table.
  -help             Show this text.
  -nointersec       Don't try intersection to calc. object point coords.
  -ori_from_dlt     Calculate EO from DLT parameters.
  -pr_ic            Print image coords.
  -pr_input         Print all input data.
  -pr_oc            Print object points after initialization.
  -prec             Evaluate precision of parameters.
  -resec            Try resection to calc. transf. params.
  -update_cams      Update cameras by AP sets.
  -verb             Gives more information.
  -version          Show program version.
  -wr_aps file      Write adjusted AP sets.
  -wr_cam file      Write cameras in file.
  -wr_eos file      Write adjusted EOs.
  -wr_mis file      Write adjusted IMU misalignment.
  -wr_ocs file      Write adjusted object coords.
  -wr_opa file      Write object point analysis.
  -wr_res file      Write residuals in file.
```

Chapter 4

Command options

Option	Description
-chk_in	Check the input files. Read only the input files and stops the program after reading.
-corr	Computation of correlations. Printing in table form.
-corr_matrix	Computation of correlations. Printing in matrix form. Only for orientation und additional parameters.
-covar	Computation of covariances. Printing in matrix form. Only for orientation und additional parameters.
-db_upari	Print UPARI-Table.
-help	Show usage text.
-nointersec	Don't try intersection to calculate approximations for object point coordinates.
-ori_from_dlt	Calculate inner and exterior orientation parameters from DLT.
-pr_ic	Print input image coordinates. This option has only effect to the listing together with option -pr_input .
-pr_input	Print all input data without image coordinates.
-pr_oc	Print object points after initialisation.
-prec	Computation of inner accuracy (precision).
-resec	Try resection to calculate approximations for orientation/transformation parameters.
-update_cams	Update the inner orientation of the camera by the additional parameters.
-version	Show program version.
-wr_aps file	Write the adjusted additional parameters in file .
-wr_cam file	Write the camera parameters in file .

continued on next page

`-wr_eos file`	Write the adjusted orientation parameters in `file`.
`-wr_mis file`	Write the adjusted IMU misalignment params. in `file`.
`-wr_ocs file`	Write the adjusted Objektkoordinaten in `file`.
`-wr_opa file`	Write the Object Point Analyse (OPA) data in `file`.
`-wr_res file`	Write the residuals plus the observed image coordinates in `file`.

Table 4.1: Program options

4.1 Program Output

The program output take place on the standard output on the console. To redirect the standard output from a command use the > (greater than) symbol followed by the name of the output file. If the file does not already exist it will be created. For example:

```
> dgap config.cfg > run.lst
```

This redirects the standard output from the program so that it goes to the file `run.lst`.

Bibliography

Horst A. Beyer. *Geometric and Radiometric Analysis of a CCD-Camera Based Photogrammetric Close-Range System*. PhD thesis, Eidgenössische Technische Hochschule Zürich, Institut für Geodäsie und Photogrammetrie, ETH Hönggerberg, 8093 Zürich, May 1992. Diss. ETH Nr. 9701.

Leica. *ADS40 Information Kit*. Leica Geosystems, Division for Mapping and GIS, 10840 Thornmint Road, San Diego, CA 92127, USA, April 2002. http://www.gis.leica-geosystems.com.

Scott O. Mason. *Expert System-Based Design of Photogrammetric Networks*. PhD thesis, Eidgenössische Technische Hochschule Zürich, Institut für Geodäsie und Photogrammetrie, ETH Hönggerberg, 8093 Zürich, May 1994. Diss. ETH Nr. 10475.