Digital Aerial Mapping

A Hands-On Course

Martin Vermeer[1]   Getachew Tesfaye Ayehu[2]

18th September, 2021

[1] martin.vermeer@aalto.fi

[2] sget2006@yahoo.com

Cover image: "Earthrise" from the Moon as photographed by Lunar Orbiter I on August 23, 1966. The spacecraft carried a complete aerial photography system for doing a photogrammetric survey of the lunar surface in preparation of the manned Apollo missions, possibly the first extraterrestrial use of photogrammetry ever. The system, inside a pressure tank, was completely analogue however, using film emulsion which was processed chemically and electronically scanned, after which the images were transmitted telemetrically to Earth.

Also, the rendering of the images, after reception and recording on video tape, was analogue back in 1966. Only recently, thanks to the Lunar Orbiter Image Restoration Project (LOIRP), has the imagery been digitally restored from the old tapes that were about to be thrown away...

All of humanity at the moment of exposure, alive and dead, is within the frame of this image.
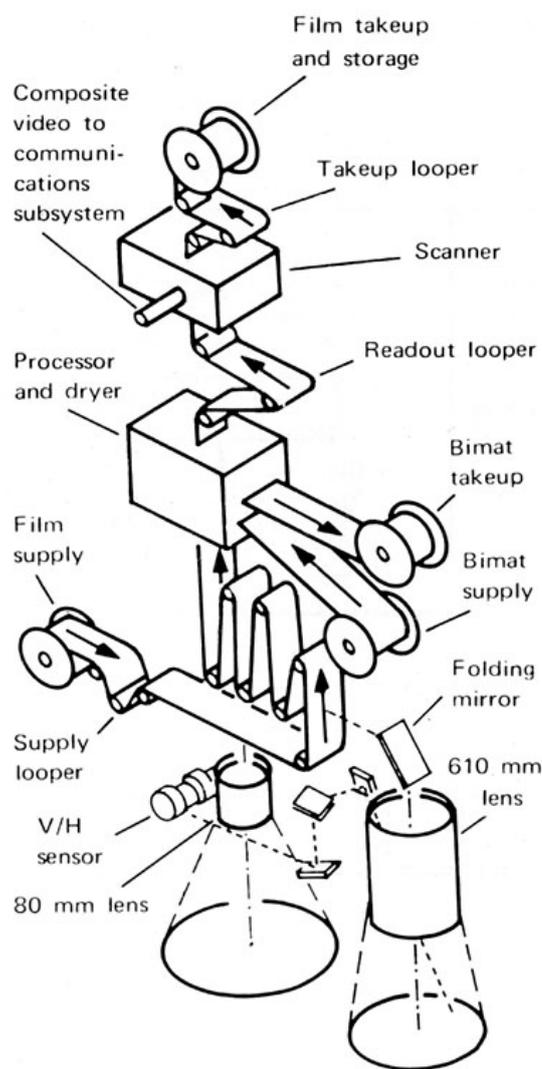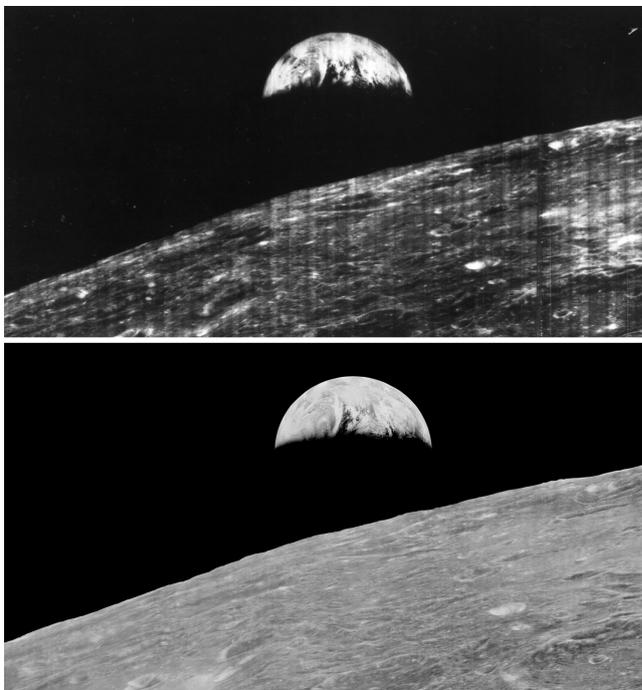

FIGURE 1.—*Photographic subsystem.*

Preface

This book grew out of a series of lectures the first author, professor of geodesy at Aalto University's Department of the Built Environment, held as a guest lecturer at Bahir Dar University's Institute of Land Administration in Ethiopia. The intensive course, delivered over two weeks full time, was to give the students hands-on experience using digital aerial photogrammetry software in the service of mapping the country for economic development. We did so using two different digital-photogrammetry software packages: the commercial package ERDAS Imagine™, which contains LPS, the Leica Photogrammetry Suite; and the open-source package e-foto, which was developed in an academic environment in Brazil. The hardware, fifteen digital photogrammetry workstations with ERDAS pre-installed, was an earlier donation by SIDA, the Swedish development co-operation agency.

Ethiopia is hardly the only country in need of modern map-making technology, and personnel understanding its operation and utilisation, for its development. This is the motive for developing the lecturing material into a book, aimed at giving the reader everything (short of a laptop) to get started.

Helsinki, 18$^{\text{th}}$ September, 2021,

Martin Vermeer, Getachew Tesfaye Ayehu

Acknowledgements

Contents

Chapters

List of Tables

List of Figures

Introduction

1

Ever since humankind has studied the precise figure of the Earth and the lands on her surface, humans have dreamt of rising above the Earth in order to see in a single view what otherwise can only be inferred by laborious measurements and calculations. The invention of aviation — first by lighter-than-air airships, later by motorised aircraft — has made this dream a reality. During the last century, aerial mapping and aerial measurement — *photogrammetry*, the science and art of performing precise geodetic measurements from photographic images — has played a critical role in mapping countries aspiring to economic progress and prosperity.

Much of the equipment needed in photogrammetry has until recently been highly specialised, difficult and expensive to produce. This continues to be the case for the aerial vehicles used, although there, the development of small and inexpensive unmanned aerial vehicles (UAV) or "drones" is opening up new perspectives. In the field of processing the aerial imagery obtained, new opportunities have however been created by the digitisation of the photogrammetric tool chain.

This booklet offers an introduction to digital aerial photogrammetry based on freely-available software running on ordinary personal computers. It is intended to be suitable for self-study, but may also be part of an academic curriculum. In addition to this text, an Intel-based 64-bit personal computer or laptop, and the Open Source software intended to be used with the text, only a pair of anaglyphic glasses (red-cyan) is needed.

Using aerial photogrammetry for constructing orthophoto maps is mission critical for the development of a *cadastral system*, as well as in support of large *infrastructure construction* works, in many rapidly industrialising and urbanising economies. Also *zoning*, the regulation of

construction activities especially in built-up areas with a view to orderly land use, is critically dependent on accurate maps. Aerial photogrammetry is still the only technology offering the geodetic mapping precision required especially in urban and peri-urban areas.

The digital photogrammetry software e-foto (Brito et al., 2011) was developed at the State University at Rio de Janeiro for the express purpose of teaching aerial photogrammetry using standard personal computers, without the expense and licence hassle of commercial photogrammetric software packages. It offers a largely complete tool chain from camera calibration and interior orientation over exterior orientation to aerotriangulation and feature extraction, and a less complete one for digital terrain model construction and ortho-rectification aimed at the construction of orthophoto maps. The package is already well sufficient for educational use, and will serve as the basis for this tutorial.

1.1 Learning outcomes

After completion of the course the students will understand the basics of photogrammetry, primarily traditional photogrammetry using photographic imagery, but also airborne laser scanning. They will be able to

1. Plan an aerial photography or airborne laser scanning mission: compile a valid flight plan based upon requirements, terrain, weather, and climatic conditions.

2. Set up the processing of aerial imagery in a photogrammetric digital work station: form 3-D models by performing interior and exterior orientation of images, and extract terrain co-ordinates in a given external reference frame.

3. Perform with understanding the extraction of digital height contours and other mappable features from stereo models.

4. Practice the art of designing working orthophoto maps, combining in a legible, clear and aestetically attractive way information from the photos and external information in a way that is useful to the map user.

5. Perform, or organise the work of, the creation of orthophoto maps from aerial photographs and externally provided information, using a photogrammetric digital work station, and possibly exploiting height information provided by airborne laser scanning.
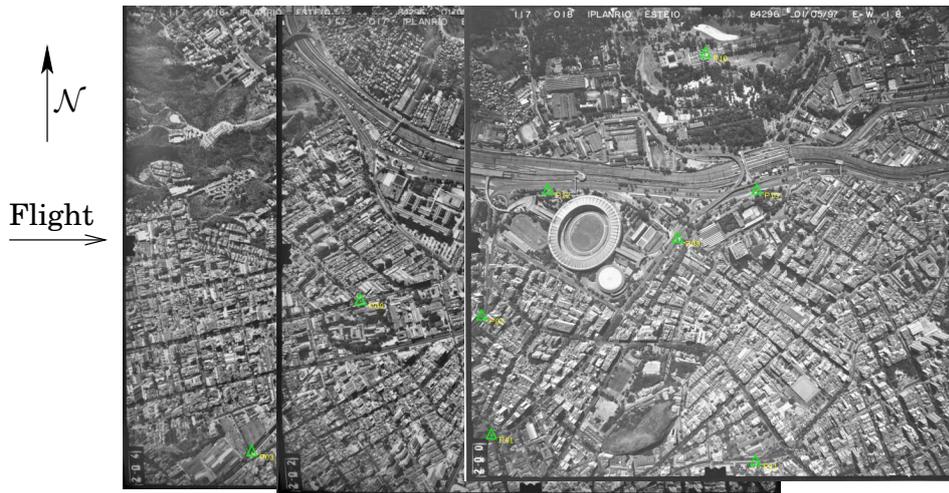
FIGURE 1.1. The Rio de Janeiro, Brazil, imagery (3).

They will leave the course with a broader understanding of the history of photogrammetry, its various applications (cadastral, base mapping, construction support, zoning, etc.) and technologies (traditional, laser scanning, space data, etc.).

The total course load is estimated at 80 hours or 3 ETCS credits.

1.2 Imagery provided

Besides the software e-foto, three sets of images are provided:

1. a set of three aerial images from Rio de Janeiro is kindly provided by the e-foto team, as a test set for e-foto. Flying height was some 4000 feet (1200 m). The images are 2810×2810 pixels in size.

2. a set of five very large ($12\,000 \times 12\,000$ pixels) images from the Ringroad II project in the Greater Helsinki area was provided by

FIGURE 1.2. The Finnish Ringroad II, Helsinki area, imagery (5).

FIGURE 1.3. The Bahir Dar, Ethiopia, imagery (6).

the municipality of Espoo and the mapping consultancy FM International Oy FINNMAP. The images were obtained from 2000 feet (600 m) flying height. These may be freely used for educational purposes.

3. a set of six digital images for Bahir Dar city is provided obtained with a much smaller camera.

1.3   Supporting material provided

Provided are

1. Camera calibration documents:

    (a) for the camera used in Finland:

       `Wild_RC30_camera_calibration_certificate.pdf`.

    (b) For the camera used in Brazil:

       `1997_cert_calib_1_2.jpg` and `1997_cert_calib_2.2.jpg`.

    (c) The camera calibration data for Bahir Dar is included in this text, table 4.4.

2. *Maps* of ground control points (GCP):

    (a) for Finland: `Control_points_images.pdf`.

    (b) For Brazil: `GCP_DESC.pdf`.

    (c) For Bahir Dar: `GCP_bahirdar.pdf`.

3. *Lists* of ground control point co-ordinates:

    (a) for Finland: `control_points_keha2_harjoitus.txt`.

    (b) For Brazil: `GCP_DESC.pdf`, the same file as containing the maps.

    (c) For Bahir Dar: `GCP-bahir-dar.txt`. The list for Bahir Dar is also given in table 6.1.

1.4 Software provided

The e-foto software is provided in packaged form, both for Linux and for Windows. This software is 64-bit and will only run on a 64-bit computer and operating system with an Intel or compatible CPU.

The original downloadable packages are on the software repository SourceForge, for both Windows and Ubuntu Linux: (E-foto).

1.5 Background materials

On the Internet a wealth of learning materials on photogrammetry, aerial mapping and related subjects may be found, for example (Vicente). The hope is nevertheless that the text presented here fills a niche.

Aerial photogrammetry and its history

2.1 Aerial photogrammetry

As described in Kraus (2007), photogrammetry "allows one to reconstruct the position, orientation, shape and size of objects from pictures [. . .]". These pictures may have been produced by the traditional process of exposure on a light-sensitive emulsion on glass plates or film, followed by development or processing using chemicals, to bring out the visible image. Nowadays they may also have been captured as pixel arrays by digital image sensors. A third, also modern, technique is *laser scanning*, where every pixel (picture element) has *distance information* added, as obtained from the travel time of laser light.

In this text we will restrict ourselves to *aerial* photogrammetry, in which the image sensor has been mounted on an aircraft. Much of what we have to say will apply also to *terrestrial* photogrammetry, however, and to the processing of satellite imagery.

The science and art of photogrammetry is first of all about the *geometry* of the scene being imaged and mapped: the purpose is the production of *geometrically correct map information*, in the form of actual maps (paper or digital), or sets of digital co-ordinate data. However, also *radiometric data* may be the object of study; this is called *remote sensing*. The borderline between these two is fluid, e. g., photogrammetry may result in the production of colour or greytone maps — maps depicting radiometric data — from aerial photographs through the process of *ortho-rectification*. All these are objects of our study.

2.2 Historical overview

It is perhaps not surprising that the history of photogrammetry points back as far as Leonardo da Vinci, perhaps farther. Leonardo explored

optics and geometry, including the phenomenon of perspective which is also used in the visual arts. Albrecht Dürer again constructed (1525) devices enabling the making of perspective drawings — the precursors of cameras. Stereoscopy came not long after that.

From the 19th century onward, the development of photogrammetry went in parallel with that of photography, the automated capture of imagery on, at first, a light sensitive emulsion in a *camera*.[1] Aviation, at first by balloon, enabled aerial imagery to be acquired. As a pioneer should be mentioned the French officer Aimé Laussedat (1819–1907), who from 1849 to 1858 conducted extensive research, both practical and theoretical, into the use of photography for topographic mapping. Platforms used by him back then were balloons and kites, but the method proved too cumbersome for practical use.

Motorised aviation changed everything. Figure 2.1 shows the standard way of mapping a large area using vertical photographs that overlap, in order to obtain three-dimensionality.

The secret of extracting three-dimensional terrain information is in being able to *reconstitute* the three-dimensional situation that existed in the terrain, including the locations of the aircraft in air space at the moments when every photograph was being taken — though obviously at a much smaller scale. The most straightforward approach to this was taken in the *Multiplex* reconstitution instrument, where every image exposure location was physically represented by a *slide projector*, with an actual slide copy of the image inside. Clearly, positioning the projector's projection centre and orienting the projector correctly — that is corresponding to where the aircraft was and how the camera was pointing at the moment of exposure — was quite a challenge!

The projected terrain image was then "scanned" in three dimensions by an operator wearing red-and-cyan anaglyphic glasses moving a screen-and-pen device around on their desk.

The important related invention of the *floating mark* was made by Franz Stolze (1836–1910, a German inventor better known for his gas turbine related work) in 1892, exploiting the human capacity for stereo vision in order to directly measure 3-D co-ordinates in a photogrammetric stereo model.

---

[1]*Camera obscura*, latin for "dark room", an ancient device for capturing perspective images.
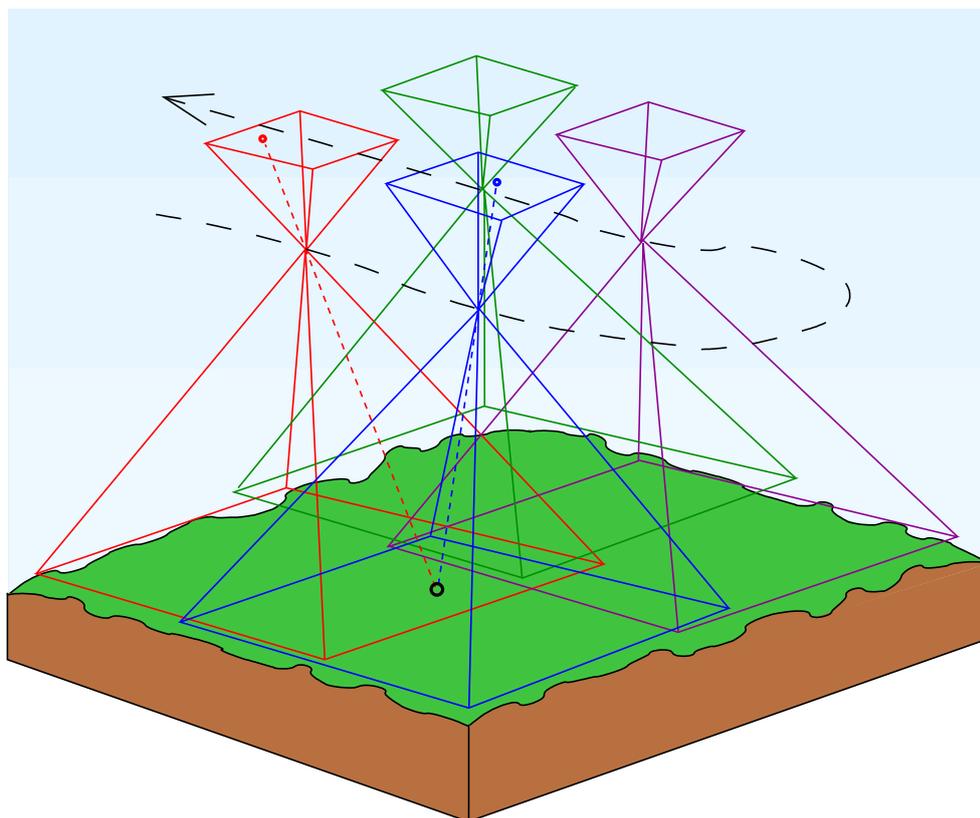
F<small>IGURE</small> 2.1. Aerial photography. Images are colour coded. The dashed line is the aircraft path connecting the image exposure locations — meaning the locations in space where the camera optical centre was when the image was exposed — obtaining two strips of two images each. Each terrain point will show in *at least two* images.

Skipping much preliminaries brings us to the 1920s and 1930s, when aerial photogrammetric techniques began to be adapted in production settings, for the production of accurate topographic base maps over large areas. Pioneers worth mentioning are Willem Schermerhorn,[2] Vilho Petter Nenonen,[3] and Yrjö Väisälä.[4] This — the production of topographic base maps — continues to be the largest field of application, volume-wise,

---

[2]Willem Schermerhorn (1894–1977) was the father of Dutch (and Dutch-Indies!) photogrammetry, internationally renowned. He also served as Dutch prime minister.

[3]Artillery General Vilho Nenonen (1883 – 1960) invented, together with academician Yrjö Väisälä, the horizon camera bearing his name, a photogrammetric camera capturing not only the terrain underneath but also the horizon; a first attempt at in-flight camera orientation determination, well predating current inertial measurement units.

[4]Yrjö Väisälä (1891–1971) was an academician, astronomer, physicist and universal genius who also contributed to Finnish photogrammetry.

FIGURE 2.2. An array of Multiplex devices. Photo © The National Map, U.S.
Geological Survey.

of aerial photogrammetry.

Before the computer era, the processing of aerial imagery was done
in large, expensive optomechanical devices; analogue computers, as it
were. The era of the personal computer has relegated all of this to
history, the stuff of science museums. In a way this is a pity, as this
application stretched the use of optics and fine mechanics to the limit
— and sometimes beyond it. But today, anyone with a €500 laptop,
€0.50 stereo glasses, smart software (which may be free!) and a little
perseverance and willingness to learn can become the operator of a
complete digital photogrammetric work station. The current text is
designed to help you make this journey.

2.3   Photogrammetric workflow

The photogrammetric workflow is nowadays almost completely digital.
The only analogue elements remaining are the optics of the aerial camera,
and still the paper of some of the output, like maps and reports, although
those may also be digital, and will so be more and more in the future.

The flow may be visualised as in figure 2.4.
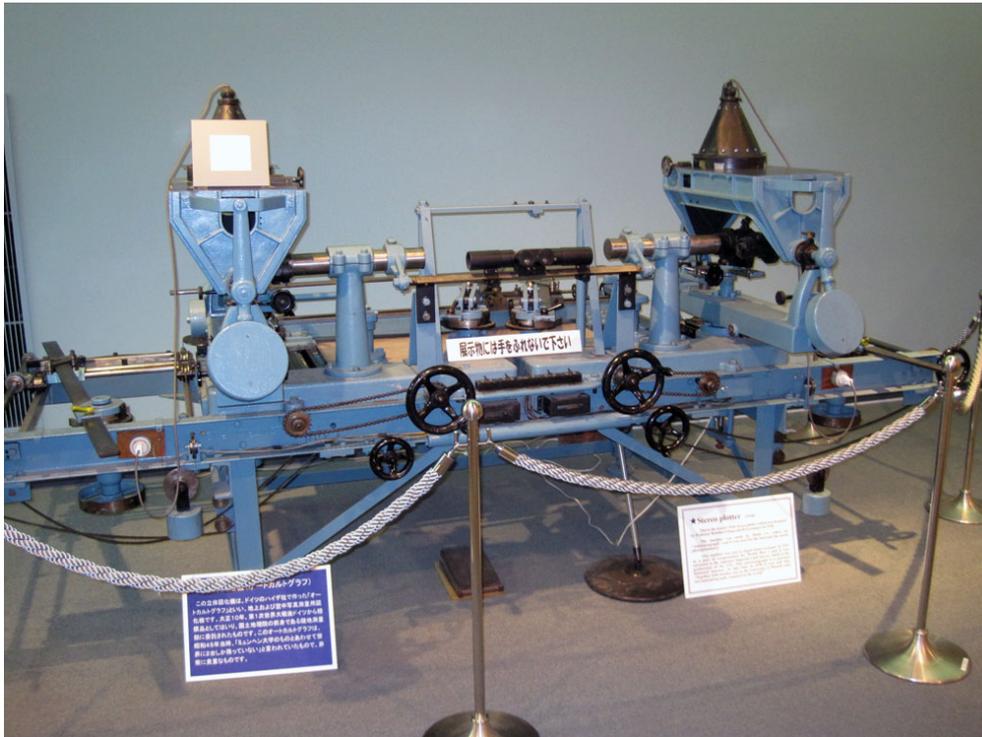
FIGURE 2.3. A traditional stereo plotter, now on display, in Japan. Note the two big hand wheels and the foot wheel for moving the "floating mark" horizontally and vertically, respectively. Photo © National Geospatial Technical Operations Center, U.S. Geological Survey. http://ngtoc.usgs.gov/images/kari-japan07.jpg.

2.4 Theoretical preliminaries

2.4.1 The least-squares method

Numerical photogrammetry is an application of geodetic adjustment theory, the theory of least squares. This theory, which goes back to Gauss and Legendre, allows us to determine or *estimate* unknown quantities — elements of a *vector of unknowns* x — if observations are given as elements of a vector of observations $\underline{\ell}$.

It is assumed that we have available a greater number of observations than there are unknowns:

$$\underline{\ell} = \begin{bmatrix} \underline{\ell}_1 \\ \underline{\ell}_2 \\ \vdots \\ \underline{\ell}_n \end{bmatrix}, \qquad \mathsf{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{bmatrix},$$

where m is the number of unknowns and $n > m$ the number of observations available.

Digital image   Camera   Control point
                 calibration   co-ordinates

Extract
image   Interior   Measure
co-ordinates   orientation   tie points

                 Exterior
                 orientation   Bundle block
                                  adjustment

Digital   Compute
terrain   terrain
model   co-ordinates

                                  Stereo
Orthorectification feature
                    extraction

Orthophoto map
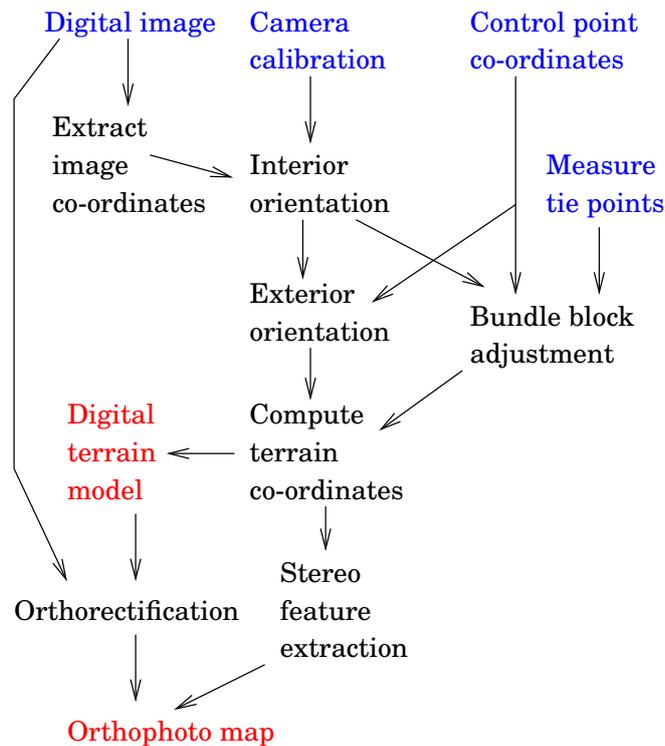
FIGURE 2.4. Photogrammetric workflow. Blue, input data; black, processes; red,
          outputs. Simplified.

In a real situation in aerial photogrammetry, we have to describe the measurement geometry mathematically using our knowledge of how a camera works geometrically. Typically all kinds of small corrections, like the camera calibration model, the effect of the atmosphere and of the curvature of the Earth's surface, are taken care of in advance; after that, the description of the observation geometry, or *observation equations*, can be relatively simple.

Note that the observations $\underline{\ell}$ are given here as *stochastic quantities*. This is because, being measured quantities, they are typically uncertain, meaning that they contain a small *measurement uncertainty*. For example image co-ordinates x, y measured in the camera co-ordinate system may be uncertain by, say, a hundredth of a millimetre:

$$x = 93.255 \pm 0.01\,\text{mm}, \qquad y = -53.345 \pm 0.01\,\text{mm}.$$

The uncertainty, or *standard deviation*, or *standard error*, in this case amounts to $\pm 0.01\,\text{mm}$. The symbol we use for this is σ.

sigma $\sigma\Sigma$

We describe the uncertainty of the whole observation vector $\underline{\ell}$ by a

variance-covariance, or *variance*, *matrix*:

$$\text{Var}\{\underline{\ell}\} = \Sigma = \Sigma_{\ell\ell} = \begin{bmatrix} \sigma_1^2 & & & \\ & \sigma_2^2 & & \\ & & \ddots & \\ & & & \sigma_n^2 \end{bmatrix}.$$

This matrix Σ is very often *diagonal*, meaning that the different obser- sigma $\sigma\Sigma$ vations $\underline{\ell}_i$ are assumed *statistically independent* of each other. Each diagonal element represents the square of the standard error of the corresponding observation: for example if the standard error of observation ℓ_i is $\pm 0.01\,\text{mm}$, then $\Sigma_{ii} = \sigma_i^2 = (0.01\,\text{mm})^2 = 10^{-4}\,\text{mm}^2$.

Very often the standard errors of all the observations ℓ_i are similar, and are all very small numbers. In this case it is advantageous to write

$$\text{Var}\{\underline{\ell}\} = \sigma_0^2 Q_{\ell\ell},$$

where $\sigma_0 = 0.01\,\text{mm}$ is the *standard deviation* or *error of unit weight*, and

$$Q_{\ell\ell} = I_n$$

— the unit matrix of dimension $n \times n$ — is called the *weight coefficient matrix*. This approach is attractive even if the uncertainties σ_i of the observations are not all the same. In that case Q is no longer a unit matrix,

$$Q_{\ell\ell} = \Lambda_n,$$

a general diagonal matrix, where nevertheless the diagonal elements will be close to unity. This is convenient for numerical work.

2.4.2 The observation equations

The general symbolic form of the system of observation equations is

$$\underline{\ell} + \underline{v} = F(\widehat{x}), \tag{2.1}$$

in which

- $\underline{\ell}$ is the vector of observations as discussed,
- $\widehat{x}$ is the vector of *least-squares estimates* of the unknowns x. The "hat" symbol is generally used to indicate statistical estimates,
- $\underline{v}$ is the vector of *residuals*: the difference between the actually observed values $\underline{\ell}$ and the "theoretical observed values" $\widehat{\ell} = F(\widehat{x})$

implied by our least-squares solution $\widehat{x}$. In other words, it describes *how good* this solution is: we wish these residuals to be as small as possible. The least-squares approach achieves this in a sense that we will discuss. Residuals are an essential part of the quality control of geodetic and photogrammetric networks.

The least-squares solution is defined as the solution that *minimises the sum of the squares of the residuals*, i.e.,

$$\sum_{i=1}^{n} \underline{v}_i^2 = \min.$$

Here we shall use a slightly more general rule, a *quadratic form* to be minimised:

$$\underline{v}^{\mathsf{T}} Q_{\ell\ell}^{-1} \underline{v} = \sum_{i=1}^{n} \sum_{j=1}^{n} \left[Q_{\ell\ell}^{-1} \right]_{ij} \underline{v}_i \underline{v}_j = \min.$$

Here we see the inverse of the weight coefficient matrix, called the *weight matrix*:

$$P = Q_{\ell\ell}^{-1}.$$

Use of this form considers that the different observations $\underline{\ell}_i$ may have different uncertainties, and therefore should be assigned different *weights* in the least-squares calculation: the greater the uncertainty of an observation, the lower its weight should be.

About the vector of unknowns x, it typically contains such things as

1. the location co-ordinates and orientation angles of the camera at the moment each image was taken

2. the location co-ordinates of any auxiliary points that are visible in multiple images, so-called *tie points*, chosen and measured in the images to strengthen the geometric interconnection between images

3. the location co-ordinates of any other interesting but unknown points appearing in two or more images.

2.4.3 Linearisation

The general form of the observation equations, equation 2.1, is *non-linear*, and no ready methods exist for obtaining the least-squares solution $\widehat{x}$ from a given $\underline{\ell}$, $Q_{\ell\ell}$. Before use in actual computations we have to *linearise* these equations, which is done as follows.

$$y = f(x)$$

Linearization interval

$$y = y^{(0)} + a\left(x - x^{(0)}\right)$$

$y^{(0)}$

$x^{(0)}$
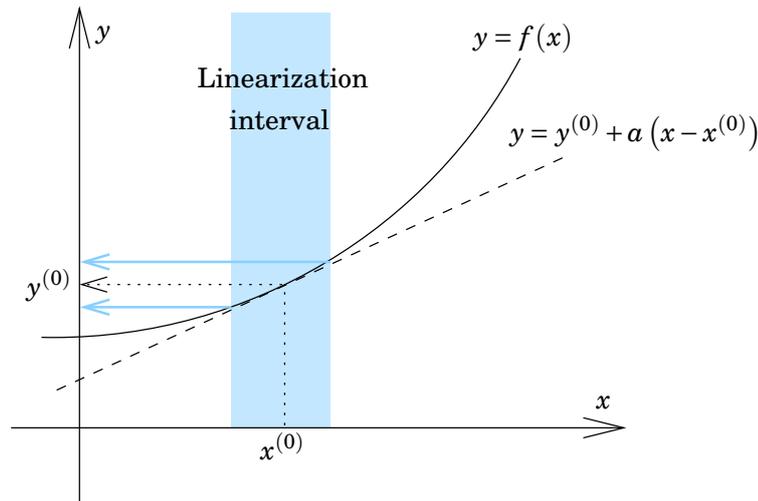
FIGURE 2.5. Linearisation in one dimension.

We consider that photogrammetry is a very precise technique: measured co-ordinates x, y in the images can be obtained to within a hundredth of a mm, though the values themselves, and the image size, may be many centimetres. Also co-ordinates in the terrain will be determined to within a few cm even for flight heights of hundreds of metres or more.

What this means is that the observation equation model function $F(\cdot)$ will be *very nearly linear in the argument range of interest*. See figure 2.5. We are allowed to linearise it.

1. Choose *approximate values* for the vector of unknowns x: call this vector $\mathsf{x}^{(0)}$.

2. Compute, for this vector of approximate values, corresponding approximate values for the vector of observations $\underline{\ell}$, as follows:

$$\ell^{(0)} = F\left(\mathsf{x}^{(0)}\right). \tag{2.2}$$

3. Expand $F(\widehat{\mathsf{x}})$ into a *Taylor expansion* using $\mathsf{x}^{(0)}$ as the origin:

$$F(\widehat{\mathsf{x}}) = F\left(\mathsf{x}^{(0)}\right) + \mathcal{J}_F\left(\mathsf{x}^{(0)}\right)\left(\widehat{\mathsf{x}} - \mathsf{x}^{(0)}\right) + \ldots \tag{2.3}$$

in which

$$\mathcal{J}_F = \begin{bmatrix} \frac{\partial}{\partial x_1}F_1 & \frac{\partial}{\partial x_2}F_1 & \cdots & \frac{\partial}{\partial x_n}F_1 \\ \frac{\partial}{\partial x_1}F_2 & \frac{\partial}{\partial x_2}F_2 & \cdots & \frac{\partial}{\partial x_n}F_2 \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial}{\partial x_1}F_n & \frac{\partial}{\partial x_2}F_n & \cdots & \frac{\partial}{\partial x_n}F_n \end{bmatrix}$$

is the so-called *matrix of Jacobi*, the matrix of all partial derivatives between the components of F and the components of x.

4. Now, substituting equation 2.3 into equation 2.1 and subtracting equation 2.2 from it yields

$$\left(\underline{\ell} - \ell^{(0)}\right) + \underline{v} = \mathcal{J}_F\left(x^{(0)}\right)\left(\widehat{x} - x^{(0)}\right),$$

or with the notations

$$\Delta\underline{\ell} = \underline{\ell} - \ell^{(0)},$$
$$\Delta\widehat{x} = \widehat{x} - x^{(0)},$$
$$A = \mathcal{J}_F\left(x^{(0)}\right),$$

this becomes

$$\Delta\underline{\ell} + \underline{v} = A\,\Delta\widehat{x}, \tag{2.4}$$

the *linear form* of the observation equations.

Very often in writing equation 2.4 we leave the deltas (Δ) out (but remember that these quantities are a lot smaller than the original quantities $\underline{\ell}, \widehat{x}$, which is numerically convenient!). The matrix A is called the *design matrix* of the problem. It expresses the linear dependence that exists between the unknowns and the observations, *at or close to the geometry that actually existed at the time of observation*.

2.4.4   The least-squares solution

The least-squares solution now follows straightforwardly:

$$\widehat{x} = \left(A^{\mathsf{T}}Q_{\ell\ell}^{-1}A\right)^{-1}A^{\mathsf{T}}Q_{\ell\ell}^{-1}\underline{\ell},$$

and for the uncertainties of the estimates of the unknowns $\widehat{x}$ we obtain as their variance matrix:

$$\mathrm{Var}\{\widehat{x}\} = \Sigma_{\widehat{x}\widehat{x}} = \sigma_0^2\left(A^{\mathsf{T}}Q_{\ell\ell}^{-1}A\right)^{-1},$$

sigma $\sigma\Sigma$ with σ_0 the earlier defined standard error of unit weight, a "typical" uncertainty of a single observation.

All this is readily codeable.

We mention separately that in the often occurring case where the observations are statistically independent of each other and of equal uncertainty — sometimes the abbreviation *i.i.d.*, independent and identically distributed, is used for this —, the weight coefficient matrix is $Q_{\ell\ell} = I_n$, as we saw in subsection 2.4.1. Then the above equations simplify to

$$\widehat{x} = \left(A^{\mathsf{T}}A\right)^{-1}A^{\mathsf{T}}\underline{\ell}, \qquad \mathrm{Var}\{\widehat{x}\} = \Sigma_{\widehat{x}\widehat{x}} = \sigma_0^2\left(A^{\mathsf{T}}A\right)^{-1}.$$

2.5   Content of this text

This tutorial is accompanied by hands-on exercises using the software e-foto, an open-source digital photogrammetric work station package developed in Brazil (Brito et al., 2011). This software is more limited in scope than commercially available packages, but allows many useful photogrammetric workflows. It has the merit of being free and thus accessible to many. Also, it is simpler in structure and less automated, making it more suitable for teaching and self-study. Finally, it is available for other operating systems besides Windows™.

Somewhat complementary to this are open-source software packages for remote sensing and geographic information systems (GIS) use. An example of the latter is Quantum GIS (Linifiniti Consulting CC, 2008, maintained),[5] also called QGIS. This open-source package is more like a framework connecting other open-source products for more specific tasks. It is very extensive and unfortunately rather tricky to install due to its many dependencies. Like the commercial package ERDAS Imagine™,[6] it integrates a geo-referencing solution, which is however based on the open-source library package proj4 ((PROJ contributors, 2021)) which may also be used separately. This library again is based on the global compilation of reference systems — thousands of them! — originally maintained by the European Petroleum Survey Group (EPSG) and found at (EPSG codes).

Further useful open-source packages are GDAL, which converts a great many raster and vector image formats used in remote sensing and geographic information systems into each other, and Monteverdi/OTB, a package for processing remote-sensing imagery, especially from satellites.

2.6   Introduction to e-foto

E-foto is a software developed at the State University at Rio de Janeiro, Brazil, for the purpose of hands-on teaching digital aerial photogrammetry on ordinary personal computers. Development started in 2002; originally a number of independent modules were developed, which have been integrated into a complete photogrammetric workflow. This integration is currently still somewhat incomplete.

E-foto is distributed under a *free software licence*; thus there are no

---

[5]This excellent open-access training manual was originally developed in South Africa.

[6]A product of Intergraph™.

restrictions on installing this software even on home computers in private use. It is intended, however, for use in an academic environment, where the cost of commercial software licences might be limiting, while learning the basics of digital photogrammetry doesn't require these advanced commercial packages.

The home web site of e-foto is at (E-foto home).

In this tutorial, we describe the use of e-foto using as examples:

1. the test set of three images over Rio de Janeiro, collected in 1997, and

2. the test set of five images over Greater Helsinki collected in 2004.

Screenshots shown in this booklet are taken sometimes from the Brazilian processing, sometimes from the Finnish processing. Also, these screenshots may be from versions of e-foto older than the latest ones, and the user interface has undergone (minor) changes. Please don't be confused.

Data include

- the digital photographs,

- camera calibration data,

- a list of control points with their terrain co-ordinates, and

- descriptions of the control points, allowing them to be found in the digital images.

In this tutorial, we describe use of e-foto on the Linux operating system, Ubuntu 12.04 or 14.04. Versions for Windows™ are available, and a MacOS™ version is in preparation.

Technical note: for Linux a 64-bit version is available from the e-foto web site. The packages are in the .deb format, which will install on Debian and derivatives, like Ubuntu. In order to be able to process large aerial images — larger than 8192×8192 pixels — it is necessary to use the 64-bit version, which will only run on 64-bit Linux,[7] which again will only run on 64-bit hardware.

Also for Windows a 64-bit version is currently available. The 32-bit versions are being phased out.

We start up e-foto by typing at the command prompt ($) in a terminal window

---

[7]When self-compiling, the source tree will automatically compile a 64-bit version on 64-bit Linux.
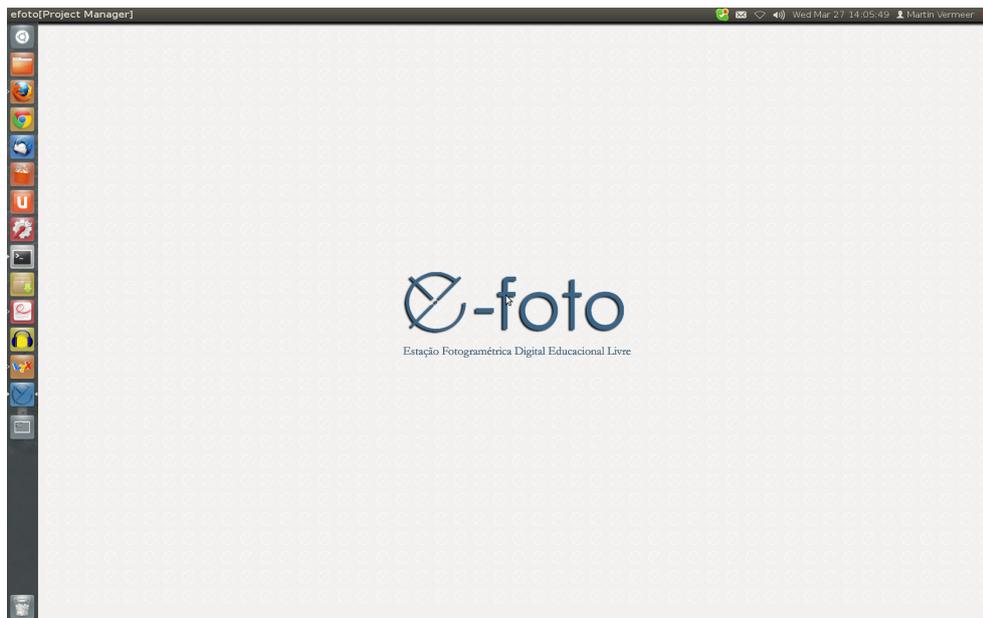
FIGURE 2.6. E-foto start screen.

```
$ e-foto
```

and pressing <Enter>. The start screen comes up (figure 2.6).

The main menu in the top left of the window (see figure 2.7) offers the options Project, Execute and Help.

Under Execute, you see the various modules that you may execute on the photogrammetric data at your disposal. We will discuss each of these further on. The logic connecting these modules (the names used vary a bit) is given in figure 2.8.

First of all, however, you need to set up a *project*, containing information on how to find your data, as well as auxiliary information on your camera (calibration data) etc. This setting-up procedure is discussed in a later

FIGURE 2.7. E-foto main menus.

```
                    ┌──────────────┐
                    │  Start e-foto │
                    └──────────────┘
                           │
                 Y      ╱─────╲      N
              ┌────────  New?  ────────┐
              │        ╲─────╱         │
              ▼                        ▼
      ┌──────────────────┐    ┌──────────────────┐
      │ Create project file │  │  Load project file │ ◄──┐
      └──────────────────┘    └──────────────────┘     │
              │                        │               │
         ╱─────────╲   Digital         │               │
        ╱  Camera?  ╲──────────┐       ▼               │
        ╲           ╱          │  ┌────────────────────┐│
         ╲─────────╱           │  │ Phototriangulation  ││
       Film │                  │  └────────────────────┘│
      ┌────────────────────┐   │       │                │
      │ Interior orientation │  │       │                │
      └────────────────────┘   │       │                │
              │                 │       │                │
              ▼                 ▼       │                │
      ┌────────────────────┐◄───┘       │                │
      │ Exterior orientation │           │                │
      └────────────────────┘  3-D co-ordinates           │
              │          ╲       │                        │
              ▼           ╲      ▼                        │
      ┌──────────────┐     ╲ ┌────────────────┐          │
      │ Stereo plotting│     ╲│ DEM extraction │          │
      └──────────────┘       └────────────────┘          │
         Map features              │                      │
              │                    ▼                      │
              │          ┌──────────────────────┐         │
              │          │ Orthophoto mapping    │         │
              │          └──────────────────────┘         │
              │              Map background               │
              ▼                    │                      │
      ┌────────────────────────────────────┐              │
      │     Save updated project file       │ ─ ─ ─ ─ ─ ─ ┘
      └────────────────────────────────────┘
```
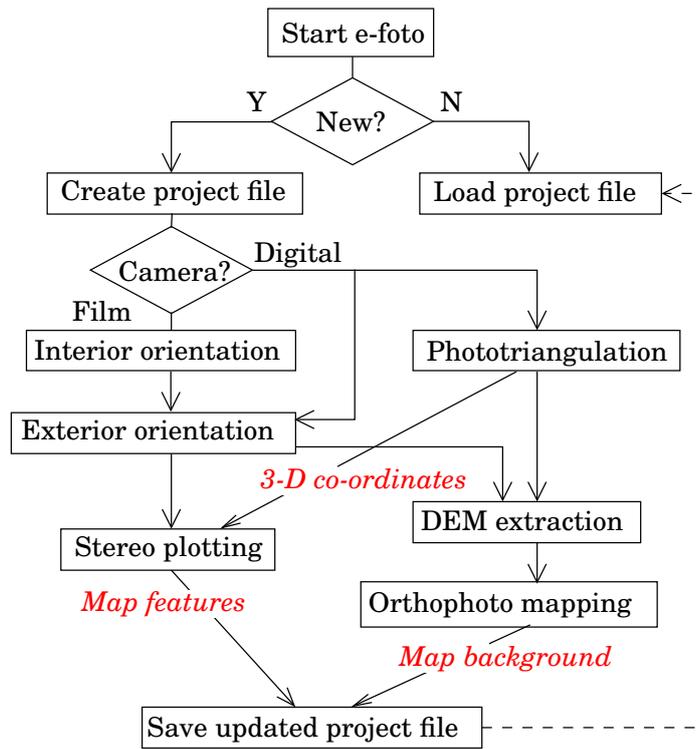
FIGURE 2.8. E-foto modules and processing logic.

chapter.

Aerial photography and mapping

3.1 Atmospheric reduction

If aerial imaging is done from a significant height above the ground, also atmospheric refraction needs to be modelled. For vertical imaging, the correction model used will be symmetric around the vertical axis: a standard atmosphere model may be used which will be quite good enough in view of the smallness of this correction.

The refractive index n of air can be obtained from any of a number of models called "standard atmospheres". For our purpose, correcting vertical photogrammetry, we may use an approximate expression.

We assume the atmosphere to be *horizontally stratified*, figure 3.2, and apply Snell's[1] law, figure 3.1. The stratification can be approximated by a *sequence of discrete steps* in refractive index n, see figure 3.2. Then we
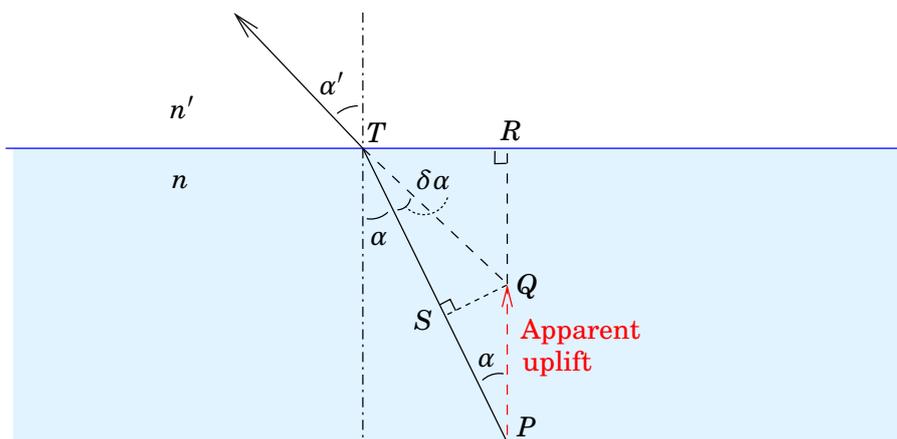
FIGURE 3.1. Snell's law: change in angle of incidence at a refractive-index interface.

---

[1]Willebrord Snell van Rooyen, "Snellius", $1580 - 1626$, Dutch astronomer, physicist, mathematician and geodesist.
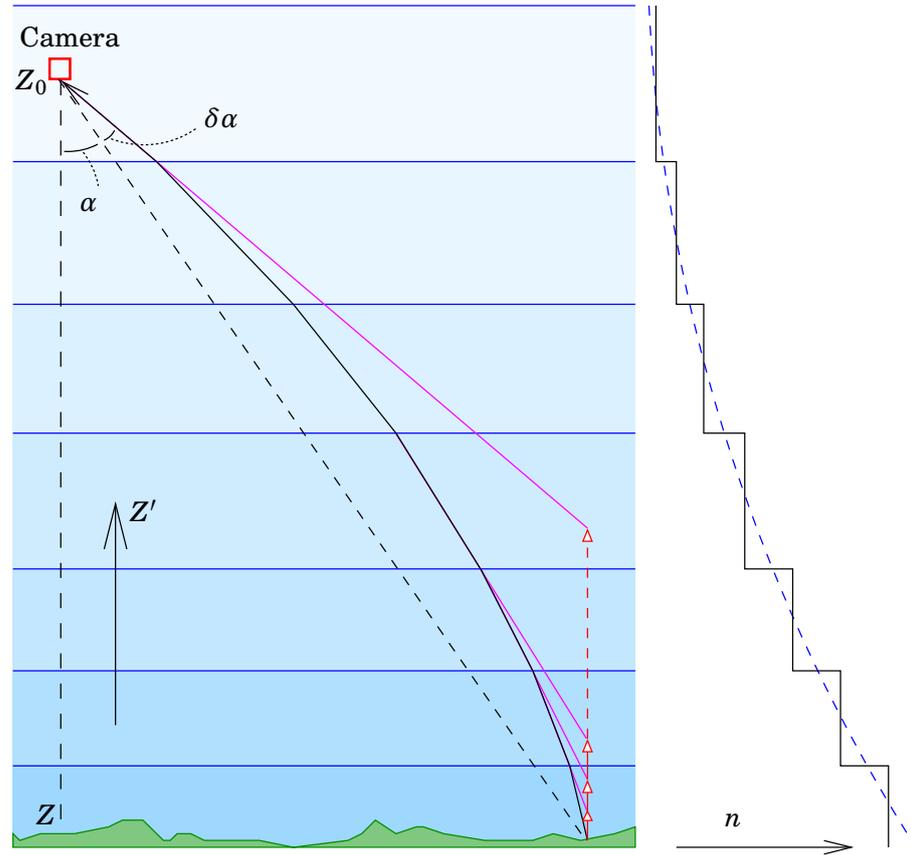
FIGURE 3.2. Atmospheric stratification and ray propagation.

have, with Snell's law for a *single* refractive-index interface, that is, a step change in refractive index by the ratio of n' and n:

$$\frac{PQ}{PR} = \frac{PS/\cos\alpha}{PT\cos\alpha} = \frac{PS}{PT\cos^2\alpha} = \frac{PT - QT\cos\delta\alpha}{PT\cos^2\alpha} \approx \frac{PT - QT}{PT\cos^2\alpha}.$$

Here we have assumed that the change in direction $\delta\alpha$ is small compared to the ray direction α itself.

Snell's law gives

$$\frac{n}{n'} = \frac{\sin\alpha'}{\sin\alpha} = \frac{RT/QT}{RT/PT} = \frac{PT}{QT},$$

So

$$\frac{PQ}{PR} = \frac{PT - (n'/n)PT}{PT\cos^2\alpha} = \frac{1 - n'/n}{\cos^2\alpha}$$

$$\implies PQ(\alpha) = \frac{1 - n'/n}{\cos^2\alpha}PR = \frac{(n - n')/n}{\cos^2\alpha}PR \approx \frac{n - n'}{\cos^2\alpha}PR = -\frac{\Delta n}{\cos^2\alpha}PR,$$

for n close to unity. In the small-angle limit for α we have

$$PQ(0) = -\Delta n\, PR \tag{3.1}$$

and

$$PQ(\alpha) = \frac{PQ(0)}{\cos^2 \alpha}. \qquad (3.2)$$

This is the *apparent uplift* of a terrain point, expressed in the height difference PR between terrain and refractive-index interface, produced by a *refractive-index increment* at this interface. In the following we write this height difference as $PR = Z' - Z$.

Now, for a number of interfaces $k = 1, \ldots, K$, expression 3.1 should be summed to obtain the total uplift:

$$PQ_{\text{tot}}(0) = -\sum_{k=1}^{K} PR_k \Delta n_k,$$

and, in the limit for a continuous refractive-index gradient, the sum becomes an integral:

$$PQ_{\text{tot}}(0) = -\int_{n(0)}^{n(Z_0)} PR(n)\, dn = -\int_0^{Z_0} PR(Z') \frac{dn(Z')}{dZ'} dZ'.$$

Now let us assume that the index of refraction n — or rather, the expression $n - 1$ — is an *exponential function of height* Z', with a scale height τ:

tau τT

$$n(Z') - 1 = C \exp\left(-\frac{Z'}{\tau}\right).$$

Then, the *derivative* of the refractive index

$$\frac{dn(Z')}{dZ'} = \frac{d}{dZ'}\left(n(Z') - 1\right) = -\frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right).$$

With this

$$PQ_{\text{tot}}(0) = \int PR \frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right) dZ' = \int (Z' - Z) \frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right) dZ',$$

and we must evaluate this integral all the way through the atmosphere:

$$\int \overbrace{(Z' - Z)}^{u} \overbrace{\frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right)}^{v'} dZ' =$$

$$= \overbrace{(Z' - Z)}^{u} \overbrace{\left[-C \exp\left(-\frac{Z'}{\tau}\right)\right]}^{v} - \int \overbrace{1}^{u'} \overbrace{\left[-C \exp\left(-\frac{Z'}{\tau}\right)\right]}^{v} dZ' =$$

$$= -(Z' - Z) C \exp\left(-\frac{Z'}{\tau}\right) - \tau C \exp\left(-\frac{Z'}{\tau}\right) =$$

$$= -\left[(Z' - Z) + \tau\right] C \exp\left(-\frac{Z'}{\tau}\right)$$

using integration by parts, annotated in the u, v formalism: $\int u v' dx = [uv] - \int u' v\, dx$. C is a still unknown constant. Now from sea level $Z = 0$ to outer space $Z_0 = \infty$ this becomes

$$I_0^\infty (0) = \int_0^\infty (Z' - 0) \frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right) dZ' = C\tau^2,$$

and we know this, the total atmospheric delay in zenith for visible light from sea level to space, to be approximately $\delta Z^0 \approx 2.4\,\mathrm{m}$. It follows that $C = \delta Z^0 \cdot \tau^{-1}$.

Now the same integral, not from sea level to outer space, but from terrain level Z to flight level Z_0, will be given by

$$I_Z^{Z_0}(0) = \int_Z^{Z_0} PQ(0)\, dZ' = \int_Z^{Z_0} (Z' - Z) \frac{C}{\tau} \exp\left(-\frac{Z'}{\tau}\right) dZ' =$$
$$= \left[-\left((Z' - Z) + \tau\right) C \exp\left(-\frac{Z'}{\tau}\right) \right]_Z^{Z_0} =$$
$$= \delta Z^0 \left(\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right) \exp\left(-\frac{Z_0}{\tau}\right) \right).$$

Now the radial offset in the image becomes, introducing again the dependence on α of the apparent uplift found above, equation 3.2:

$$\delta \alpha_{\mathrm{atm}} = \overbrace{\frac{\cos\alpha}{Z_0 - Z}}^{\text{distance}} \cdot \overbrace{\sin\alpha}^{\text{foreshortening}} \cdot$$

$$\cdot \overbrace{\frac{\delta Z^0}{\cos^2 \alpha} \left(\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right) \exp\left(-\frac{Z_0}{\tau}\right) \right)}^{I_Z^{Z_0}(\alpha)}$$

$$\implies \delta r_{\mathrm{atm}} = \frac{c}{\cos^2 \alpha} \delta \alpha_{\mathrm{atm}} =$$

$$= \frac{c \tan \alpha}{\cos^2 \alpha} \cdot \frac{\delta Z^0}{Z - Z_0} \left(\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right) \exp\left(-\frac{Z_0}{\tau}\right) \right) =$$

$$= \frac{r}{\cos^2 \alpha} \cdot \frac{\delta Z^0}{Z_0 - Z} \left(\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right) \exp\left(-\frac{Z_0}{\tau}\right) \right). \quad (3.3)$$

Here, c is the calibrated focal length of the camera. we may use, e. g., $\tau = 8.4\,\mathrm{km}$ and $\delta Z^0 = 2.4\,\mathrm{m}$. These values may be adjusted to better correspond to known local conditions during imaging.

It may not be obvious, but the right-hand side in equation 3.3 is to first order linear in $Z_0 - Z$ and vanishes at $Z_0 - Z = 0$. Note that both the expression in big parentheses and its first derivative vanish at $Z_0 = Z$,

making it a to first order quadratic expression in $Z_0 - Z$ there:

$$\frac{d}{dZ}\left[\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right)\exp\left(-\frac{Z_0}{\tau}\right)\right]_{Z=Z_0} =$$

$$= \left[-\frac{1}{\tau}\exp\left(-\frac{Z}{\tau}\right) + \frac{Z_0 - Z}{\tau^2}\exp\left(-\frac{Z_0}{\tau}\right) + \frac{1}{\tau}\exp\left(-\frac{Z_0}{\tau}\right)\right]_{Z=Z_0} =$$

$$= -\frac{1}{\tau}\exp\left(-\frac{Z_0}{\tau}\right) + \frac{Z_0 - Z_0}{\tau^2}\exp\left(-\frac{Z_0}{\tau}\right) + \frac{1}{\tau}\exp\left(-\frac{Z_0}{\tau}\right) = 0.$$

A very similar geometric derivation, but presented even more simply, is found in Schut (1969).

Kraus (2007) give a more precise expression from the literature:

$$\delta r_{\mathrm{atm}} = \frac{r}{\cos^2\alpha}K,$$

with

$$K = \frac{0.00241}{Z_0}\left(\frac{Z_0^2}{Z_0^2 - 6Z_0 + 250} - \frac{Z^2}{Z^2 - 6Z + 250}\right). \qquad (3.4)$$

Some theoretical background to this is given in Gyer (1996).[2]

In figure 3.3 we compare our own simple, horizontally stratified exponential model above with the model of Gyer (1996) — which gives tabulated values that were computed based upon the Standard Atmosphere 1976 (Public Domain Aeronautical Software, 2014) and spherical geometry —, and with the formula 3.4 given in Kraus. We see that, for

---

[2]Gyer gives an approximate formula, equation 24, for a horizontally stratified atmosphere (in our notation):

$$\delta\alpha_{\mathrm{atm}} = \frac{\tan\alpha}{Z_0 - Z}\int_Z^{Z_0}\frac{1}{2}\frac{\left(n(Z') - n(Z_0)\right)\left(n(Z') + n(Z_0)\right)}{n^2(Z_0)}dZ' \approx$$

$$\approx \frac{\tan\alpha}{Z_0 - Z}\int_Z^{Z_0}\left(n(Z') - n(Z_0)\right)dZ'.$$

If we substitute here our exponential atmospheric model

$$n(Z') - 1 = \frac{\delta Z^0}{\tau}\exp\left(-\frac{Z'}{\tau}\right)$$

we obtain for the integral above

$$\int_Z^{Z_0}\left(n(Z') - n(Z_0)\right)dZ' = \delta Z^0\left(\left[-\exp\left(-\frac{Z'}{\tau}\right)\right]_Z^{Z_0} - \frac{Z_0 - Z}{\tau}\exp\left(-\frac{Z_0}{\tau}\right)\right) =$$

$$= \delta Z^0\left(\exp\left(-\frac{Z}{\tau}\right) - \exp\left(-\frac{Z_0}{\tau}\right) - \frac{Z_0 - Z}{\tau}\exp\left(-\frac{Z_0}{\tau}\right)\right) =$$

$$= \delta Z^0\left(\exp\left(-\frac{Z}{\tau}\right) - \left(\frac{Z_0 - Z}{\tau} + 1\right)\exp\left(-\frac{Z_0}{\tau}\right)\right).$$

This is seen to lead to a result identical to equation 3.3, though derived very differently.

Photogrammetric refraction, 45 degrees
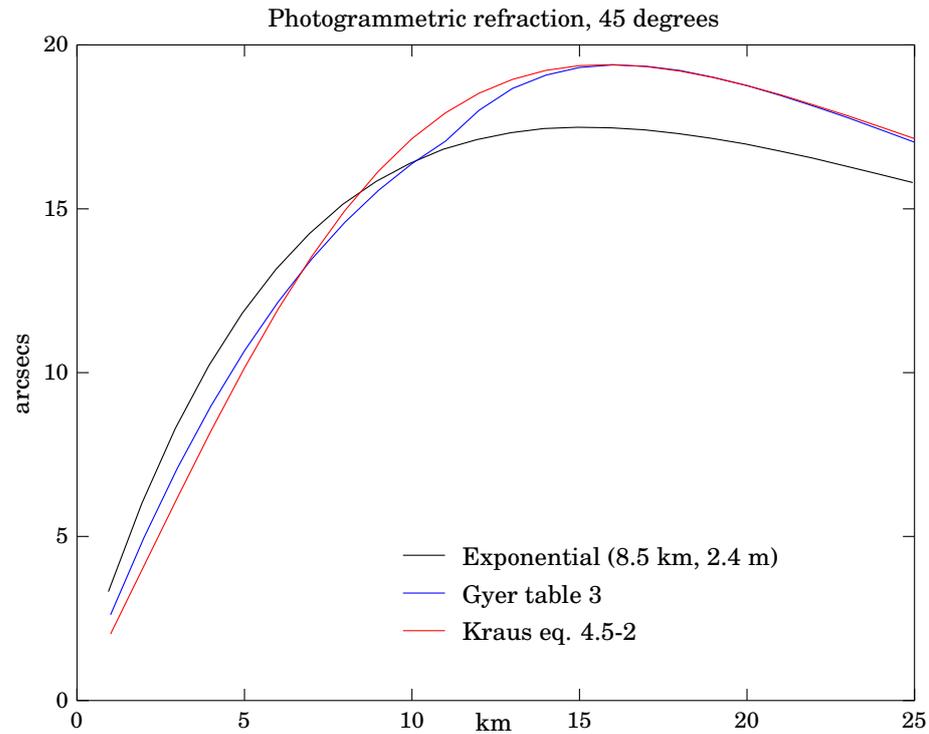
FIGURE 3.3. Comparison of atmospheric-refraction models.

common aircraft heights, our model performs no worse than the differences seen between the more sophisticated models. And this is *before* taking into account the influence of air pressure variations and climatic temperature differences.

3.2 Earth curvature reduction and local project reference

The *Earth's curvature* behaves, within the small area of an aerial mapping project, like the curvature of a sphere with a radius equal to the mean radius of the Earth, say 6371 km. The heights Z (or H) of the ground control points, which constrain the co-ordinates of the photogrammetric solution, are given above the *geoid*, i.e., the figure of the Earth, also known as mean sea level. Commonly used geodetic heights, such as *orthometric heights*, are heights above mean sea level. Also *normal heights*, while differing slightly from orthometric heights, may be understood as heights above sea level.

The horizontal co-ordinates X and Y of the ground control points — which also constrain the co-ordinates of the photogrammetric solution — can be understood as *map projection co-ordinates*, produced by projecting geocentric co-ordinates $\left(\overline{X}, \overline{Y}, \overline{Z}\right)$ onto a map projection surface,
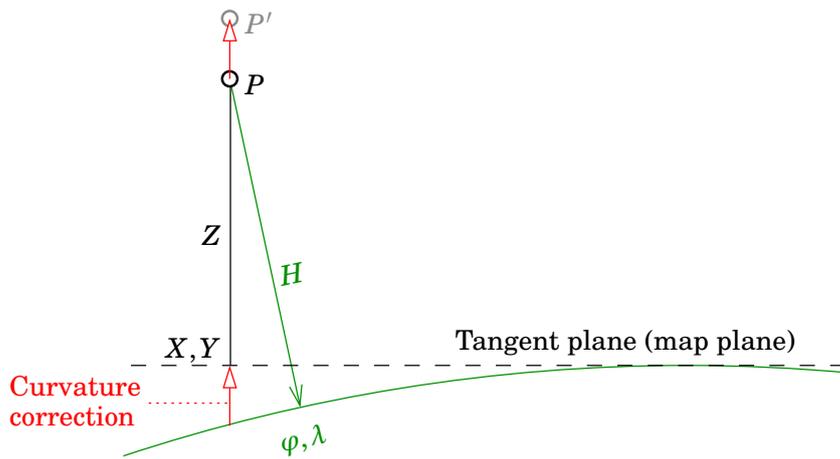
FIGURE 3.4. How to interpret the "Earth-curvature reduction".

for example the UTM (Universal Transverse Mercator) projection on the geocentric GRS80 ellipsoid, for a chosen central meridian.

The *Earth-curvature effect*, like the atmospheric effect, is radially symmetric: it produces a vertical offset of

$$\delta Z_{\text{curv}} = \frac{d^2}{2R},$$

which in the image becomes visible as

$$\delta r_{\text{curv}} = \frac{r}{f}\frac{f}{Z_0 - Z}\delta Z_{\text{curv}} = \frac{r}{Z_0 - Z}\frac{d^2}{2R} = \frac{r^3}{f^2}\frac{Z_0 - Z}{2R},$$

with f the camera focal length, $d = r\dfrac{Z_0 - Z}{f}$ the distance on the ground from the camera footpoint.

Applying an Earth-curvature reduction makes it possible to do the adjustment computation of a photogrammetric block in local map projection and vertical co-ordinates, "as if" these were three-dimensional rectangular co-ordinates. figure 3.4 shows how this works: the co-ordinates in which point P is given, X, Y, H, do not together form a rectangular system. "Reducing" the observations means that we pretend to have observed point P' instead, which actually is at a vertical height H above the *tangent plane* (you can think of it as the plane onto which the map projection projects the Earth's surface), not the curved sea-level reference surface or geoid. And now the geometry is rectangular and high-school geometry and the Pythagoras theorem and such just work!

This is somewhat dangerous, however, and only as correct as the assumption that the geoidal surface in the project area resembles a sphere. Also one should always keep in the back of one's mind that, in these co-ordinates proper, the rectangular geometry is actually not valid.

3.3   GNSS and IMU measurements in aerial acquisition

Nowadays (the last ten-fifteen years) it is common in aerial photogramme-try to position the photographing aircraft in real time, use a combination of on-board GPS (Global Positioning System) — or some other global satellite positioning system. Such systems are generically called GNSS, Global Navigation Satellite Systems — and an inertial measurement unit (IMU). The units are integrated by Kalman filter and provide, in real time, the following information for the locations at which images are recorded:

1. The location of the camera optical centre $\left(\overline{X}_0, \overline{Y}_0, \overline{Z}_0\right)$ in a global geocentric reference frame, typically in a geocentric ITRF co-ordinate reference frame as provided by the GNSS. Before use, these three-dimensional co-ordinates need to be in the same reference frame as the given points in the terrain, the *ground control points* (GCP).

2. The attitude or orientation of the camera, in the form of Euler[3] attitude angles $(\omega, \varphi, \kappa)$ relative, typically, to the axes of a local North-East-Up (NEU) co-ordinate frame.

In airborne laser scanning, real-time positioning of the scanning aircraft is mandatory. As a result, the laser scan imagery (point cloud) is obtained (after transformation) initially in a geocentric reference frame. From there, it may be transformed to map projection co-ordinates X, Y and heights above sea level Z as used locally in the project area, see section 3.2.

We will have more to say in subsection 17.1.2 on the problem of system calibration connected to airborne laser scanning but relevant to photogrammetry in general.

It must be noted that the on-board GNSS device will give aircraft positions in the form of geocentric co-ordinates, rectangular $\left(\overline{X}_0, \overline{Y}_0, \overline{Z}_0\right)$,

or equivalently, as geographical co-ordinates φ, λ and height h relative to a *reference ellipsoid*, typically that of WGS84.[4]

As our project co-ordinates relate to the local terrain situation, the

---

[3]Leonhard Euler ($1707 - 1783$) was a Swiss universal genius.

[4]In fact, the reference ellipsoid of WGS84 is called GRS80; however, everybody calls that WGS84, and the parameter values that are used for them are the same to sub-millimetre accuracy.

following *conversions* are therefore necessary:

$$
\begin{bmatrix} \overline{X}_0 \\ \overline{Y}_0 \\ \overline{Z}_0 \end{bmatrix} \xleftrightarrow{\text{ellipsoid}} \begin{bmatrix} \varphi \\ \lambda \\ h \end{bmatrix} \longrightarrow \begin{cases} \begin{bmatrix} \varphi \\ \lambda \end{bmatrix} \xrightarrow{\text{map proj.}} \begin{bmatrix} X \\ Y \end{bmatrix} \\ \quad h \xrightarrow{\text{geoid model}} H = Z \end{cases}
$$

Here,

- ○ the information on the *map projection* used in defining the *map co-ordinate system* of the ground control points, and of the whole aerial mapping project, must be provided correctly to the processing software, and

- ○ a good enough *geoid model* must also be provided in order to convert the GNSS-produced heights h, which are relative to the reference ellipsoid, to the heights above sea level Z (or H) used locally in the project.

The inertial measurement unit (IMU) again typically gives orientation angles (Euler angles) $(\omega, \varphi, \kappa)$. The pitch and roll angles ω, φ are relative to the local horizon. however, the heading (yaw) angle κ is referenced to the direction of the *local meridian*, whereas what we need is a reference to the *Map North* of the map projection used (remember we use map projection co-ordinates X, Y as our working co-ordinates for the aerial mapping project), i.e., the direction of the projection's *central meridian*. The difference between the two is known as the *meridian convergence*, and it can be approximately computed by

omega $\omega\Omega$ phi $\varphi\phi\Phi$ kappa κK

$$
\delta\kappa = (\lambda - \lambda_0)\sin\varphi,
$$

where λ is our longitude (East is positive), λ_0 is the longitude of the central meridian of the map projection used, and φ is our latitude. The correction needed is now

$$
\kappa \;\; = \;\; \kappa_{\text{IMU}} + \delta\kappa,
$$

See figure 3.5. This gives us the orientation angles κ that we can use for example as input for our aerotriangulation computation, either as initial or approximate values, or — with appropriate weights — as prior constraints.

No corrections to the other two orientation angles are needed, as they refer to the aircraft axes, not any meridian direction. Also they describe deviations from the local vertical, which coincides everywhere — after applying the Earth-curvature reduction! — to the rectangular or Cartesian Z-axis of our computation geometry.
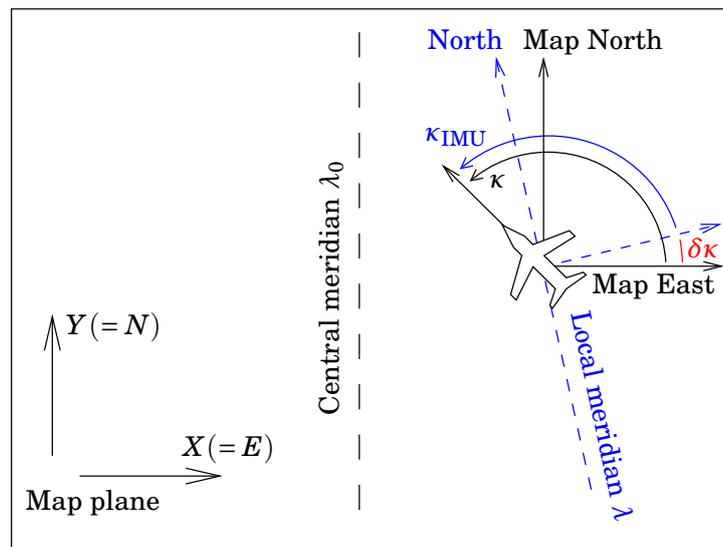
FIGURE 3.5. Meridian convergence and transformation of IMU angles to Map North. In this diagram it is assumed for the definition of κ that the operator is facing port, see section 8.9.

4 Camera types and camera calibration

4.1 Camera and sensor types, spectral areas

As camera types we may distinguish:

1. Traditional (frame) cameras, using light-sensitive emulsion on film

2. Digital cameras, using one or several CCD arrays

3. "Pushbroom" or "whiskbroom" sensors

   Examples: the SPOT satellite's HRV pushbroom, Landsat's Thematic Mapper sensor. Animated illustration: (NASA Earth Observatory)

4. Many others, like non-metric cameras.

4.1.1 Traditional (frame) cameras

A very popular size for these aerial cameras is focal length 6 inches / 152 mm, image size 9×9 inches / 22.5×22.5 cm. If the resolution is 0.01 mm, i.e., the pixel size is 0.02 mm (using the rule of thumb that resolution is half the pixel size) this means an image of 11250×11250 pixels. These cameras have four or eight fiducial marks around the edges and corners of the image field, which get exposed on the film of each frame to allow interior orientation.

Because digital cameras of sufficiently large format are something of the very recent past, there are large bodies of aerial imagery in existence which was collected on film, but which nevertheless is today processed digitally. Analogue (film) imagery is converted to digital imagery using *scanners*. These are similar to the scanners used for converting your old holiday pictures to computer format, but

- their image field is way larger, and
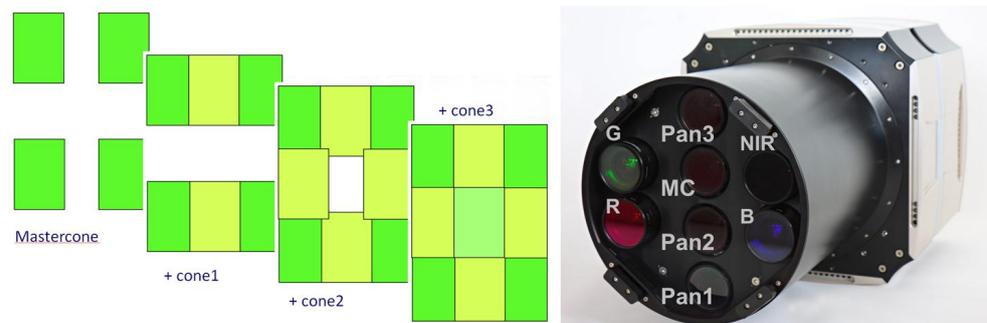
FIGURE 4.1. A modern digital aerial photogrammetric camera (Gruber et al., 2012). Note the mosaicking of the panchromatic image using four camera units, and the separate red, green, blue (RGB) and near infrared camera units. Image © Vexcel Imaging GmbH, Austria.

- their geometric precision is high, preserving the precision with which the original image was captured on film.

Professional photogrammetric image scanners are quite expensive, and are useless once the historical stock of imagery has been converted. For this reason, they, and skills in using them, only exist in a few places.

Digital photogrammetry based on scanned analogue imagery is called *soft-copy photogrammetry*.

4.1.2   Digital cameras

For a long time, these cameras typically offered smaller image sizes than the traditional cameras, having both a shorter focal length and a smaller image field. In practical use this means that one has to fly more flight lines at a lower aircraft altitude, which costs money. Only recently have digital aerial cameras offered an image size in pixels similar to traditional film cameras.

An example of a modern digital aerial camera is the Vexcel UltraCam™ Eagle (Gruber et al., 2012), which offers an image size of 104×68 mm, at a pixel size of $5.2\,\mu$m (micrometre), i.e., 0.0052 mm, giving $20\,010 \times 13\,080$ pixels, fully comparable to traditional film cameras. The focal length for panchromatic (i.e., black-and-white) image capture is 210 mm. Colour is provided using separate camera units with lower resolution.

Digital cameras typically use a CCD (charge-coupled device) array as their image sensor. The challenge is to make the image sensor field of view suitably large, given that it is a semiconductor chip cut from a silicon die of given, limited size. One approach is mosaicking multiple

CCD devices, or even multiple cameras, into one. It will be clear that this creates calibration challenges. A more obvious approach, also taken in the UltraCam, is making everything smaller, starting with the pixel size.

Recently (2015), Leica published their DMC III digital aerial camera, the first large format camera to use CMOS technology instead of CCD (Mueller and Neumann, 2016). Unlike the UltraCam, it provides the panchromatic image using a single camera unit with a focal length of 92 mm, a pixel size of 3.9 µm, and an image size of no less than 25 728 × 14 592 pixels. As the name says, this is a greytone image; additional camera units are used at a focal length of 45 mm and a pixel size of 6.0 µm, image size 8 956 × 6 708 pixels, adding red, green, blue and near infrared at lower resolution.

It is clear that digital image collection offers its own, significant advantages:

1. No need to purchase film media, only a lot of hard-disk storage space is needed.

2. The workflow is cut short, especially eliminating manual stages: no more processing of films followed by scanning.

3. No contribution to image noise from film grain.

4. Digital sensors have a greater *dynamic range*, meaning that details both within very dark and very bright areas in the same image are captured.

5. Digital sensors also capture small contrasts better and do not compress them like film does. This leads to the effective resolution achieved to be superior to scanned film images of nominally better pixel resolution.

6. More flexibility in collecting spectral information. One can separately collect pan-spectral (black-and-white) information at high resolution, and colour information at lower resolution. And one is not limited to the traditional RGB (red-green-blue) visual colours plus near infrared: one can specify and collect any spectral band, and any number of them, up to hyperspectral imaging (as in remote sensing). The UltraCam doesn't use this possibility yet though.

In recent years, the field of aerial mapping has almost completely switched to using digital cameras.

4.1.3 "Pushbroom" and "whiskbroom" sensors

Of these, worth mentioning separately are the SPOT and Landsat push-broom sensors. As this is satellite data, resolution typically is in the tens of metres, and they are of limited usefulness for geometric photogramme-try. Yet, for example Google Earth™ is largely based on imagery from the Landsat 7's ETM+ sensor, its panchromatic channel with a resolution of 15 m. This data, which covers the globe, is freely available.

A pushbroom sensor works by using a line of sensors perpendicular to the flight direction of the satellite. A whiskbroom sensor instead uses a single sensor with a small, rapidly rotating mirror, which allows the sensor to scan a line on the Earth's surface which is also perpendicular to the direction of flight of the satellite. As the satellite moves, successive lines are scanned, together building an image in a strip or band on both sides of the ground path. As such, there are no individual images, rather a long, unbroken stream of imagery. Also, instead of orientation elements for individual images, exterior orientation derives orientation elements as a continuous function of either time, or distance traveled by the satellite along its track. This is a very different paradigm.

4.1.4 Unmanned aerial vehicles

A technology that only recently has come of age and may warrant consid-eration, is the use of an unmanned aerial vehicle or "drone" as the sensor platform. This has the advantage of being much less expensive than flying a real aircraft, but at the cost of a limited payload capacity. For example, traditional aerial cameras are way too bulky for such a platform. But if the smaller image size of a commercial off-the-shelf digital camera is acceptable for the application, usually from a lower flying height, this is certainly an option. For geodetically precise mapping of large areas it is however less suitable.

4.2 Camera calibration

[1] It is necessary to *calibrate*[1] the camera used for obtaining aerial imagery, in order to be able to relate image co-ordinates in a camera-bound system — let us say, (x, y) — to real-world co-ordinates in the terrain (X, Y, Z).

---

[1]Often one speaks of "camera resection", as the term "calibration" may also refer to radiometric calibration.

Calibration alone is not sufficient for this, but it is a necessary condition. Calibration is done in a special laboratory set-up.

The simplest of cameras are *metric cameras*; in the absence of imaging errors, what they do is project points in the terrain through a single point, the *optical centre*, onto a flat imaging surface where an emulsion or light sensitive optical array is placed. For such an ideal camera, three parameters, or sets of parameters, need to be determined in calibration:

- ○ the *focal length* of the camera — this also determines the scale at which the aerial mapping takes place, and its resolution on the ground
- ○ the location x_P, y_P of the *principal point* of the camera in relation to the set of fiducial marks at the edge of the image plane
- ○ the locations, in millimeters of x and y, of the fiducial marks in the camera co-ordinate system.

Normally, however, the objective lens of the camera is not quite optically perfect; experience has shown that remaining distortion in the image tends to be radially symmetric around the principal point, and can be described by a small number of parameters. These are also included in the camera's calibration certificate, as a *table of offsets* in the image plane away from the principal point. See the below calibration results for an example.

From this table it is possible to derive the radial distortion by fitting a simple polynomial to the table values, as follows:

$$r' = r + k_0 r + k_1 r^3 + k_2 r^5 + \cdots$$

Because of symmetry considerations, only odd powers of r are included. Two or three terms typically are sufficient. Using this polynomial is preferable to using the calibration table directly, as the calibration measurements themselves contain significant uncertainty. The fit procedure diminishes the effect of this.

The fit procedure is based on the *least squares method*, cf. subsection 2.4.1. The *observations* $\underline{\ell}$ are the differences, called radial distortions, $\underline{r}'_i - r_i, i = 1, \ldots, n$, collected in a calibration set-up like depicted in figure 4.2. On the right is a grid with precisely known dimensions, which is photographed. We see that the distances r' of the image points from the principal point (PP) of the camera are different from what they would have been if the imaging system were perfect: r. The *unknowns* are
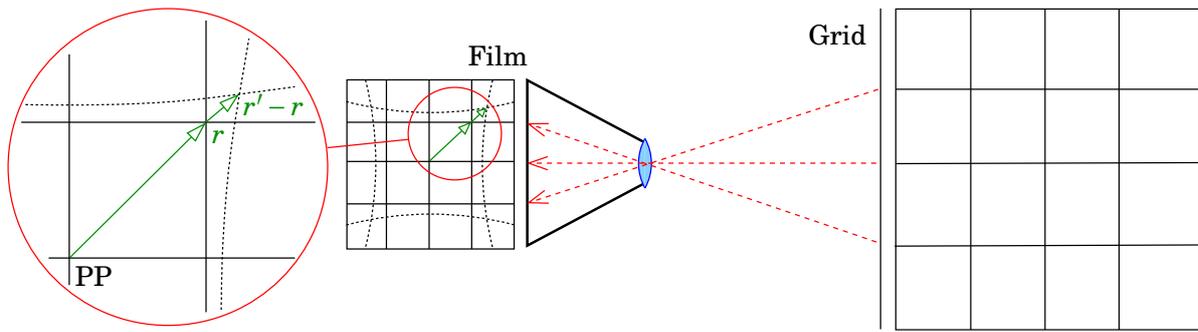
FIGURE 4.2. Camera calibration set-up, concept.

the coefficients $k_0, k_1, \ldots$. The observation equations now become, with residuals $\underline{v}_i$:

$$
\begin{bmatrix}
\underline{r}'_1 - r_1 \\
\underline{r}'_2 - r_2 \\
\vdots \\
\underline{r}'_n - r_n
\end{bmatrix}
+
\begin{bmatrix}
\underline{v}_1 \\
\underline{v}_2 \\
\vdots \\
\underline{v}_n
\end{bmatrix}
=
\begin{bmatrix}
r_1 & r_1^3 & r_1^5 \\
r_2 & r_2^3 & r_2^5 \\
\vdots & \vdots & \vdots \\
r_n & r_n^3 & r_n^5
\end{bmatrix}
\begin{bmatrix}
\widehat{k}_0 \\
\widehat{k}_1 \\
\widehat{k}_2
\end{bmatrix} .
$$

One sees that the design matrix A is made up of powers of r_i, the distances of the image point from the camera's principal point for every measured point in the calibration grid.

Note that one could correct the focal length so that $k_0 = 0$. However, this is not always done. Often, rather the *calibrated focal length* of the camera, symbol c, is chosen such, that the radial distortions $r' - r$ are *minimised over the whole range of r*, from the principal point ($r = 0$) out to the four corners of the camera's image field.

A more advanced model is that of *Duane Brown*[2] (Brown, 1966) — based on earlier, almost forgotten work by Conrady (1919) —, which includes
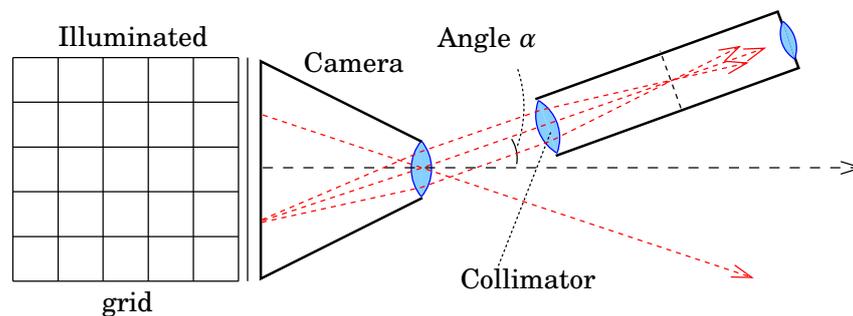
FIGURE 4.3. A practical camera calibration set-up with goniometer table.

[2] Duane Brown (1929–1994) was an eminent American photogrammetrist and geodesist, remembered especially for his contributions to camera calibration.

TABLE 4.1. Calibration data for the Finnish camera: fiducial marks relative to FC, "fiducial cross". "FC" refers to the *central cross*, the origin of all (x, y) co-ordinates. Eight fiducial marks were used, in the geometry given below right, and also in figure 7.1 right side.

Camera type:	RC30	
Lens type and serial no.:	15/4 UAG-S1, 13260	
Focal length:	153.279 mm	

Principal point	x (mm)	y (mm)
of autocollimation (PPA)	−0.012	0.004
of symmetry (PPS)	−0.012	−0.002

Fiducial marks (eight)		
1	105.998	−105.995
2	−106.006	−106.006
3	−105.998	105.995
4	106.005	106.005
5	−0.007	−111.999
6	−112.003	−0.002
7	0.004	112.003
8	112.003	0.005

3	7	4
6	FC	8
2	5	1

decentering distortion described by further parameters $P_1, P_2, \ldots$. For plate co-ordinates x, y *reckoned from the principal point*, it gives

$$x' = x + x\left(k_1 r^2 + k_2 r^4 + \cdots\right) + \left(P_1\left(r^2 + 2x^2\right) + 2P_2 xy\right)\left(1 + P_3 r^2 + \cdots\right),$$

$$y' = y + y\left(k_1 r^2 + k_2 r^4 + \cdots\right) + \left(2P_1 xy + P_2\left(r^2 + 2y^2\right)\right)\left(1 + P_3 r^2 + \cdots\right),$$

where $r^2 = x^2 + y^2$. Here it has also been assumed that the k_0 term is subsumed in the focal length.

Realistic camera calibration set-ups are not so simple. One problem with figure 4.2 is that the illuminated grid would have to be at infinity, and very large. A more realistic — though outdated — approach is given in figure 4.3 using a precise goniometer table; see Optical Metrology Centre (undated).

Alternatively, *field calibration* is used more and more — it has the merit of calibrating camera behaviour under realistic use conditions. Also using the starry sky as Nature's illuminated grid, with precisely known stellar positions, has been tried.

TABLE 4.2. Calibration data for the Finnish camera: radial distortion in micrometres (μm). Tabulated separately for the four semi-diagonals. Normally these values are fitted with a model as described above, yielding parameters k_i, P_j. Right, numbering of the semi-diagonals used.

Radius	Semi-diagonals				Mean
mm	1	3	2	4	
10	0.8	0.5	0.7	0.6	0.5
20	1.2	0.4	1.1	0.7	0.8
30	1.2	0.0	1.4	−0.1	0.6
40	1.0	−0.1	1.2	−0.4	0.4
50	0.8	−0.8	1.0	−0.6	0.1
60	0.3	−0.6	1.2	−0.9	0.0
70	0.8	−0.3	0.7	−0.1	0.2
80	0.3	0.4	0.8	0.4	0.4
90	1.2	0.9	0.7	0.5	0.8
100	1.1	1.0	0.8	0.8	0.9
110	1.7	0.7	0.2	1.4	1.0
120	2.4	1.0	0.0	1.7	1.2
130	1.1	1.1	0.1	1.2	0.8
140	−2.3	0.5	−2.4	−2.3	−1.6
148	−3.3	−2.8	−4.3	−5.9	−4.0
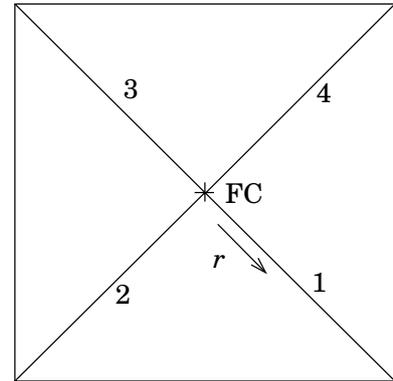
4.3 Camera examples

4.3.1 The Finnish camera

As an example we give the calibration values of a Wild camera owned by the mapping consultancy firm FM International Oy FINNMAP and used for acquiring imagery in the Ringroad II project around Helsinki. The calibration was performed by SwissOptic AG in Heerbrugg, Switzerland on January 18, 2000. See tables 4.1 and 4.2.

4.3.2 The Brazilian camera

The calibration data for this camera is given in table 4.3. It is seen that the pixel size is quite a bit larger, 0.08 mm, than for the Finnish camera, 0.02 mm, see sub-section 4.1.1. The original negatives being of the same size — some 22 cm — this difference is due to coarser scanning. It may also be that the digital images, which are offered by the e-foto development team as test images, have been reduced after scanning for better manageability.

TABLE 4.3. Calibration data for the Brazilian camera. Units assumed are powers of mm throughout. Four fiducial marks were used, in the geometry given below right, and also in figure 7.1 left side.

Camera:	Carl Zeiss RMK A 15/23	
Serial number:	137 474	
Lens:	Pleogon A2	
Serial number:	137 504	
Image size (cm):	23	23
Scanned image size (pixels):	2895	2838
Pixel size:	0.08	0.08
Calibrated focal length:	153.528 mm	

Principal point:	x (mm)	y (mm)
	-0.063	-0.037

Distortion coefficients:	
k_0	$-2.667 \cdot 10^{-5}$
k_1	$-1.813 \cdot 10^{-9}$
k_2	$-5.877 \cdot 10^{-14}$
k_3	$-5.909 \cdot 10^{-18}$
P_1	$-6.278 \cdot 10^{-7}$
P_2	$-7.172 \cdot 10^{-7}$

	3	
2	FC	1
	4	

Fiducial marks (four):	x (mm)	y (mm)
1	113.000	0.016
2	-113.006	0.018
3	0.004	113.015
4	0.007	-112.975

4.3.3 The Ethiopian camera

The data for the camera used in the Bahir Dar imaging flight is given in table 4.4. This is a fairly small digital camera.

We see that the sensor size in the cross-flight (x) direction is 4490 pixels, i.e., less than a quarter of the sensor size of the (scanned) Finnish imagery, which is no less than 11000 pixels (see section 4.1.1). This means that mapping a given area at the same resolution will require over *four times as many flight lines*, making the aerial imaging part of the project four times more expensive.

Now, in reality, this camera will be much smaller and lighter than a big film camera like the Leica RC30, which can weigh in at 100 kg and consume hundreds of watts of electric power from the aircraft on-board power system (Leica Geosystems, 2002). Therefore, it will be

TABLE 4.4. Calibration data for the Bahir Dar camera.

Camera:	IDM_300_ICS2_Calibrated	
Sensor size (pixels):	4490	3364
Pixel size (mm):	0.012	0.012
Focal length (mm):	55.1802	
Principal point	x (mm)	y (mm)
of autocollimation (PPA):	0.0910	−0.1496
of symmetry (PPS):	0.0000	0.0000
Distortion coefficients:		
k_1		$-2.4153 \cdot 10^{-5}$
k_2		$8.9472 \cdot 10^{-9}$
k_3		$2.6584 \cdot 10^{-12}$
P_1		$1.4468 \cdot 10^{-6}$
P_2		$-3.3601 \cdot 10^{-6}$

possible to fly it on a much lighter aircraft, or even an unmanned aerial vehicle or "drone", which will bring down costs a lot. This remains true even considering that a denser network of ground control points may be required to achieve good geometric strength. Another way to strengthen the geometric strength of the network is to include a large number of *tie points*, for example by automatic tie-point generation.

Flight plan

5

5.1 Problem description

When the purpose is to geometrically map an area, that fixes the basics of the flight plan: the imaging will be *vertical* (meaning that the camera looking straight down, rather than *oblique*), in a *block* consisting of a number of parallel *strips* of overlapping *images*. The critical parameter to be known is the *resolution on the ground* to be strived for: together with the resolution on the film, typically of order micrometres, we then obtain the *image scale*.

Say we want a resolution on the ground of 0.1 m. The film resolution being 0.01 mm, we obtain as the image scale the ratio of these, $1:10\,000$. If we then further know that the camera focal length is $c = 150$ mm — a typical value, but alternatives exist —, we can also directly compute the flight height: $H = 10\,000 \times 0.15$ m $= 1500$ m.

We can now also compute how big an area will appear on each image: if it measures 9 inches (22.5 cm) a side, it will photograph a square 2250 m on a side, i.e., 2.25^2 km^2 or about 5 km^2. But this doesn't yet take the *overlap* into account. In a very crude, approximate formula:

$$A \approx \left(S\overline{d}\right)^2 \left(1 - O_x\right)\left(1 - O_y\right) = d^2\left(1 - O_x\right)\left(1 - O_y\right),$$

where $\overline{d}$ is image linear size (i.e., 22.5 cm), $S = 10\,000$ is the image scale number (i.e., the scale is $1:S$), $d = S\overline{d}$ is the size of the image area on the ground (2250 m) and O_x, O_y are the longitudinal and transversal overlaps, respectively. This formula is asymptotically valid for large (many strips, many frames per strip) blocks.

Choosing $O_x = 0.60$ and $O_y = 0.30$ we obtain for the effective terrain area mapped by one image, with this camera at this resolution:

$$A \approx 1.4 \, \text{km}^2.$$
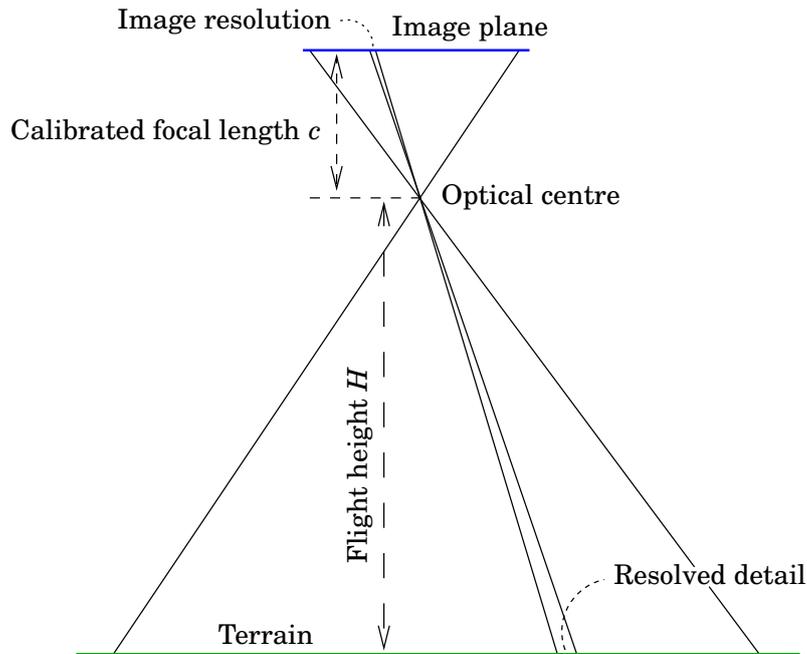
FIGURE 5.1. Relationship between flight height H, camera calibrated focal length c, film or image resolution (typically one-half of pixel size) and required resolution on the ground

Knowing the total size of the area now gives you the amount of film you are going to consume. Figuring out the number of strips, the number of images per strip, and the total flight distance — remember to take generous turns between strips! — takes further work; see the example below. By now, you should be set up to invite tenders from aerial mapping companies.

5.2 Example

As an example we consider a target area of 3×5 km. See figure 5.2. This kind of picture is definitely worth drawing.

Firstly let us assume that we fly along the longest direction of the area, length $D_x = 5$ km. This is generally preferable, as it means that the least amount of time is lost to unproductive turns of the aircraft. Then, we have 5 km to cover with images that are 2.25 km broad. But, remember that the images have to *overlap* by 60%. This means that the first and the last images are "extra". The other images are offset by $(1 - O_x) \cdot d$ from each other; this distance is also the distance between the camera exposure centres and is known as the *image* or *frame separation*.

The positions of the left edges of the images span the range $[0, D_x - d]$.

Example 5.2 43
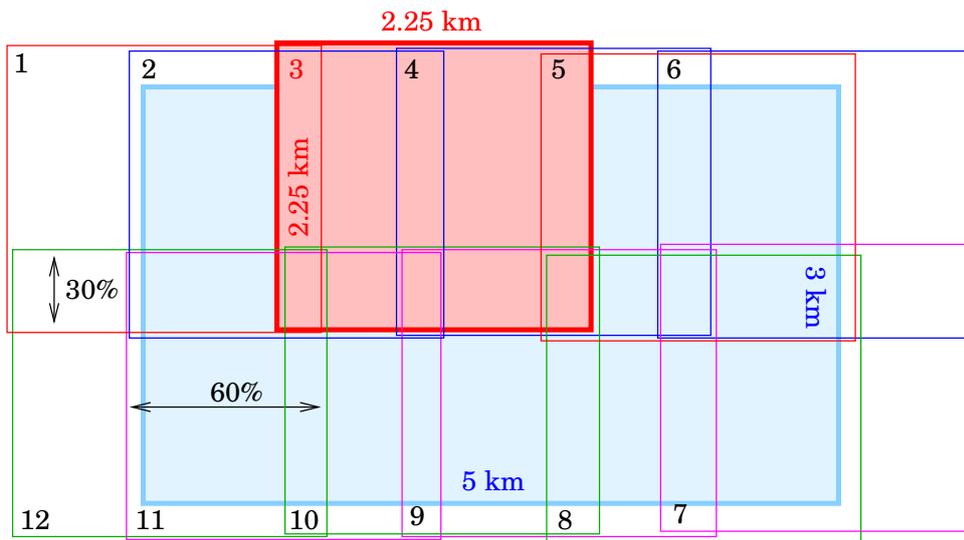
FIGURE 5.2. Flight plan target area.

This gives us the formula for the number of images (exposures) in one flight strip:

$$n_x = \frac{D_x - d}{(1 - O_x) \cdot d} + 1 + 2,$$

where the extra number 2 represents the extra first and last images in the strip (needed to produce stereo view over the whole target area), and the number 1, that the number of edges on the closed interval $[0, D_x - d]$ is one more than the length $D_x - d$ is a multiple of the offset length $(1 - O_x) \cdot d$.

Thus the number of images in a flight strip will be

$$n_x = \frac{5\,\text{km} - 2.25\,\text{km}}{0.9\,\text{km}} + 3 = 6.06.$$

One is tempted to round to 6, by for example slightly reducing the longitudinal overlap. Generally speaking, one should instead round *upward* to the nearest integer.

Obviously the number of strips needs to be 2 in order to cover the 3 km width of the area. Then there will be an overlap of at most

$$O_y = \frac{2 \times 2.25\,\text{km} - 3\,\text{km}}{2.25\,\text{km}} = 0.67.$$

As only 30%, i.e., $O_y = 0.3$, is required, we are again left with "edges" of $2 \times 2.25\,\text{km} - 3\,\text{km} - 0.3 \times 2.25\,\text{km} = 0.825\,\text{km}$ total, or 412 m on each side.

A general formula for the number of strips, analoguous to the above one for the number of images per strip, is

$$n_y = \frac{D_y - d}{(1 - O_y) \cdot d} + 1,$$

Optimized coverage area (uncertainty pink)

Target area

ΔD_x

ΔD_y Raw coverage area
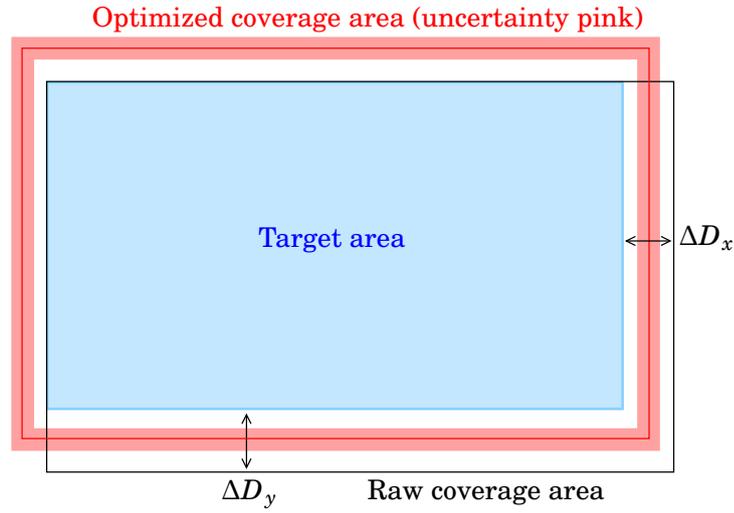
FIGURE 5.3. Placement of coverage area over target area.

with D_y the width of the target area. This yields $n_y = 1.48$, to be rounded upward to 2.

Both n_x and n_y should be *rounded upward* to the nearest integer. Their product $n_x n_y$ equals the total number of images (frames, exposures) for the whole project.

If the computed n_x and n_y are not integers but need to be rounded upward, the imagery will cover more area on the ground than the target area, and there will be "space left", see figure 5.3. The formula for this "space left", on the left and right, and upper and lower, edges, is now "space taken by images" minus "space taken by overlaps" minus again the real dimension of the target area. As follows:

$$\Delta D_x = \Big((n_x - 2)\, d - (n_x - 3)\, O_x d\Big) - D_x =$$
$$= \Big((n_x - 2)(1 - O_x) + O_x\Big) d - D_x,$$
$$\Delta D_y = \Big(n_y d - (n_y - 1)\, O_y d\Big) - D_y = \Big(n_y (1 - O_y) + O_y\Big) d - D_y,$$

half of which is the shift, in both directions, needed to place the target area in the middle of the coverage area. This helps to safeguard full coverage also in the presence of small deviations from the nominal flight plan.

The flight time between exposures is obtained directly from the image separation $\Delta x_{\text{exp}} = (1 - O_x)\, d$, as follows:

$$\Delta t_{\text{exp}} = \frac{\Delta x_{\text{exp}}}{v},$$

where v is the aircraft velocity. Multiply with the total number of exposures $n_x n_y$, and one obtains the *total photography time*.

We want every point in the *whole* target area to appear in two photographs at least. This means that a border area around the target area will be photographed on single images of the project. It is wise to have ground control points also in this border area, as this strengthens the control of the camera location and orientation for these edge exposures.

5.3 Practicalities

Both the above examples assume that the target area of measurement is *rectangular*. Sometimes it is, but rarely exactly so. For approximately rectangular areas, one may try to enclose the area in a rectangle, and then proceed as above. For irregularly-shaped areas, there is no other technique than the manual one.

1. Put a map of the area on your desk, with the outline of the target area drawn on it.

2. Cut, from paper or cardboard, *squares* that are the same size as an image on the ground, but reduced to map scale.

3. Mark on the squares the positions of forward and sideways overlap by straight lines.

4. Now, arrange the squares into a proper geometry, with full, stereo coverage, and overlaps in both directions as specified. Use pieces of sticky material to keep the squares in place.

5. You may need to experiment a little to find the best solution. Do not just minimise the number of images taken, try also to minimise the number of strips, and thus, the number of aircraft turns needed.

6. Record your result, with numbered images, on a planning document and planning map blank. Thus one obtains a *flight map*, which can be given to the pilot and the photographer of the imaging flight. The flight map should contain the flight lines (with the direction of flight indicated) and the individual exposure locations.

In an area with strong mountain slopes (Nepal!), it is not a good idea to fly up and down the slope, so this constrains the flight direction: rather, fly along the height contours at constant height.

About the *timing* of photography: in moderate climate zones it is wise to schedule for early spring, when the snow has left but deciduous trees

Air Strip

10 km

20 km
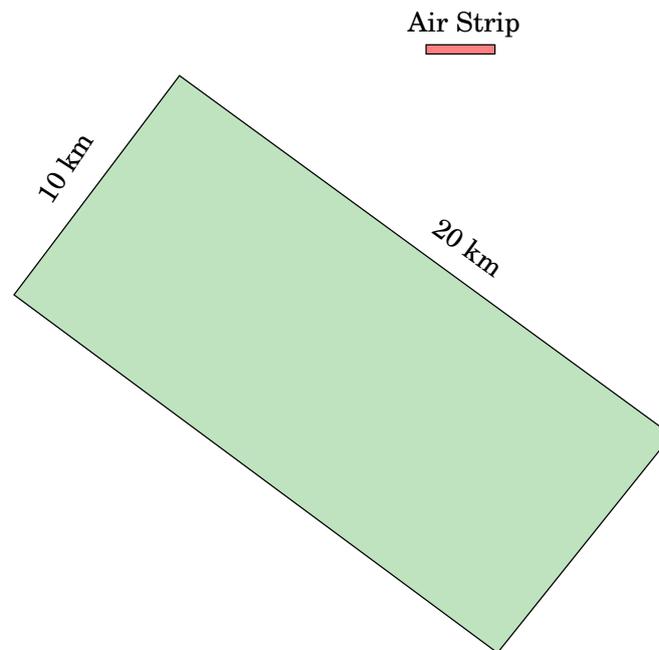
FIGURE 5.4. Area to be mapped in the flight plan exercise.

are still leafless. In the tropics, avoid the rainy (cloudy) season.

The uncertainty of weather is something to always take into account.

5.3.1 Flight Plan Exercise (1)

Given: available camera, required resolution (point accuracy) on the ground. Assume as a rule of thumb that we can achieve a point accuracy (resolution) corresponding to ± 0.5 pixel size.

- We have a film camera with focal length 150 mm and image field of $22\,cm \times 22\,cm$. The size of a pixel on the film is 0.01 mm.
- We have an area to be mapped of $10 \times 20\,km^2$, see figure 5.4.
- We require a point accuracy (ground resolution) of $\pm 5\,cm$ on the ground.

Furthermore, assume

- One image (colour) costs 1000 credits.
- The aircraft flies $400\,km/h$, and one hour of flying costs 100 000 credits.

We need to answer the following questions.

- What has to be your flying height?
- How many strips, of how many images each, do we need to fly (assume 60 % / 20 % overlap). Make a sketch including the path of

the aircraft. Which flying direction do you think is best?

○ Draw ground control and tie points into your sketch to create a good geometry for the aerotriangulation. Explain what makes the solution strong.

○ How many total images will you have to take? Estimate cost for this film material. Discuss colour vs. greytone.

○ Choose a suitable aircraft (discuss!), flying speed, cost.

○ Compute hours in the air assuming $400\,\mathrm{km/h}$ flying speed. Assume that the airstrip is close to the project area, so you can ignore approach and return flights. Estimate cost assuming 100.000 credits per hour.

5.3.2 Flight plan exercise (2)

Now we have, for the same area and ground resolution requirements, a different camera: a digital camera (see subsection 4.3.3) with a focal length of $55.18\,\mathrm{mm}$ and a pixel size of $0.012\,\mathrm{mm}$. The camera image field is 4490×3364 pixels, with *the higher pixel count being in the cross-track direction,* as is customary with digital imaging.

○ We again have an area to be mapped of $10 \times 20\,\mathrm{km}^2$, see picture.

○ We again require a point accuracy (ground resolution) of $\pm 5\,\mathrm{cm}$ on the ground.

Furthermore

○ Images have zero cost (no film is used).

○ Still, the aircraft flies at speed $400\,\mathrm{km/h}$, and one hour of flying costs 100.000 credits.

Answer the following questions:

○ What has to be now your flying height?

○ How broad, in metres, is a single flight strip on the ground?

○ Assume $60\% / 20\%$ overlap. How many flight strips are needed to cover the area?

○ Compute again hours in the air assuming $400\,\mathrm{km/h}$ flying speed, again assuming the airstrip is close to the project area. Estimate cost assuming 100.000 credits per hour. Compare with previous result.

Question: How realistic is the assumption that the same aircraft will
be used for this case as in the first case? Read carefully chapter 4.

Setting up an e-foto project file

Note that the e-foto system is integrated so, that data from a previous module is carried forward to the next module inside the *project definition file*, which has a file name extension of .epp. This means that before executing a module you must *load* the project definition file; after completing it, you must *save* it, so it is available for following modules. These load and save options are available under the Project menu item.

In figure 6.1, we see the different tabs that the Project Manager module offers:

- Project Header, already filled out with data for this project
- Terrain data
- Sensor: here the camera calibration data is entered
- Flight data
- Images: here the path names to the digital images are given
- Points: here, terrain points and their co-ordinates are entered, typically by loading them from a simple text file.

Before trying to change the information in any of the tabs, you need to press the Edit button. When finished, you press the OK button — or, to throw away a failed edit, the Cancel button.

6.1 Project header

Here is entered the general information on the project, including information on the project definition file. Figure 6.1 shows the header page of a previously filled out, saved and re-loaded project file. Not all of this information is required, and the metadata is automatically filled in.
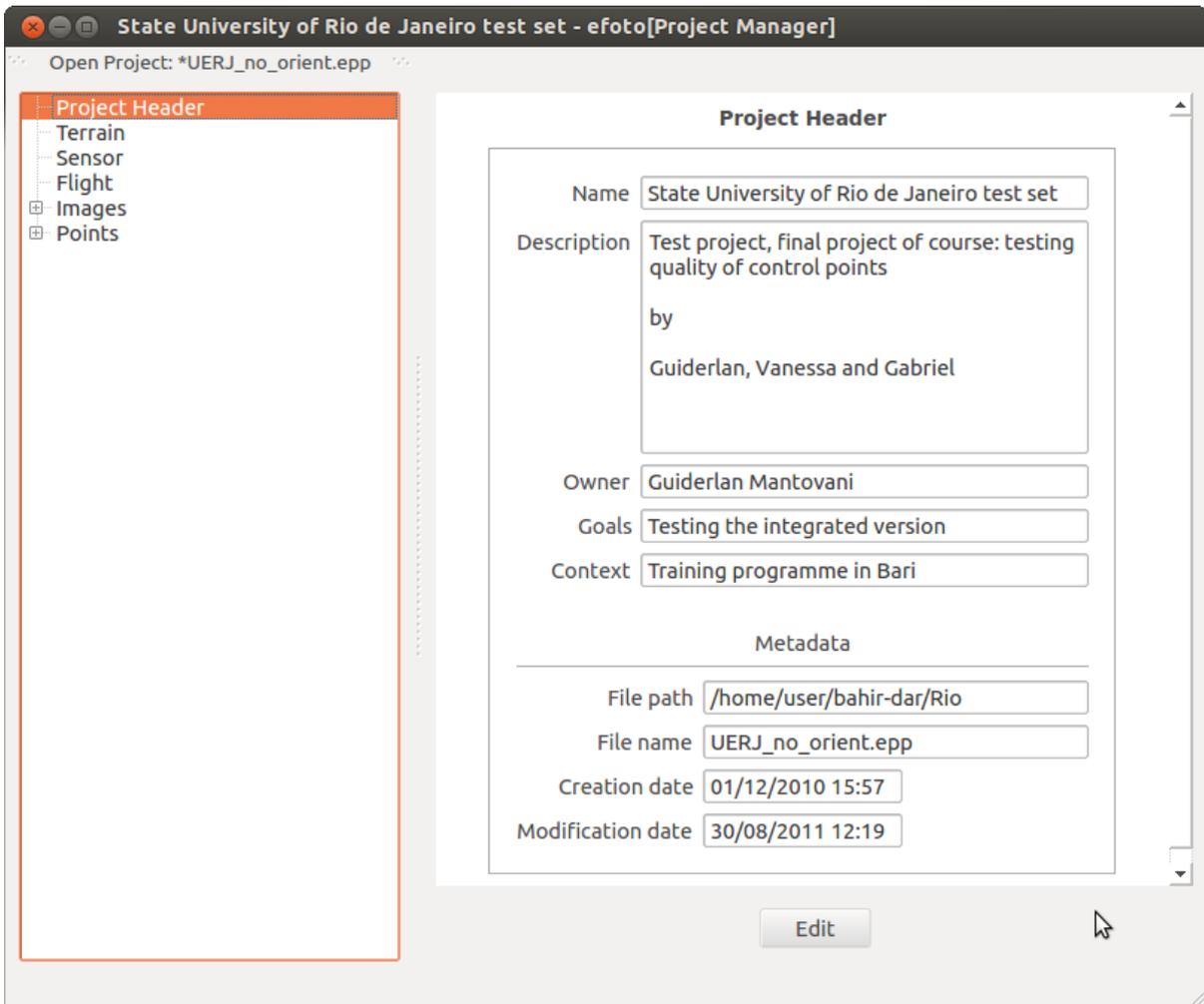
FIGURE 6.1. Loaded project file.

6.2 Terrain

Here we enter general information on the area to be mapped. The interesting fields are:

GRS (Geodetic Reference System): this is pre-filled with WGS84, but other options are SAD69 (South American Datum 1969) and SIRGAS2000 (Sistema de Referencia Geocéntrico para las Américas). Also Corrego Allegre (a Brazilian system) is provided for in the code. Unfortunately all this is hardwired at present.

Outside South America, WGS84 is the prudent choice, and may be taken as an alias for, for example, ITRFyy (International Terrestrial Reference Frame), ETRFyy (European Terrestial Reference Frame) or some similar regional geocentric reference frame. The coordinate differences among all these frames are on the sub-metre
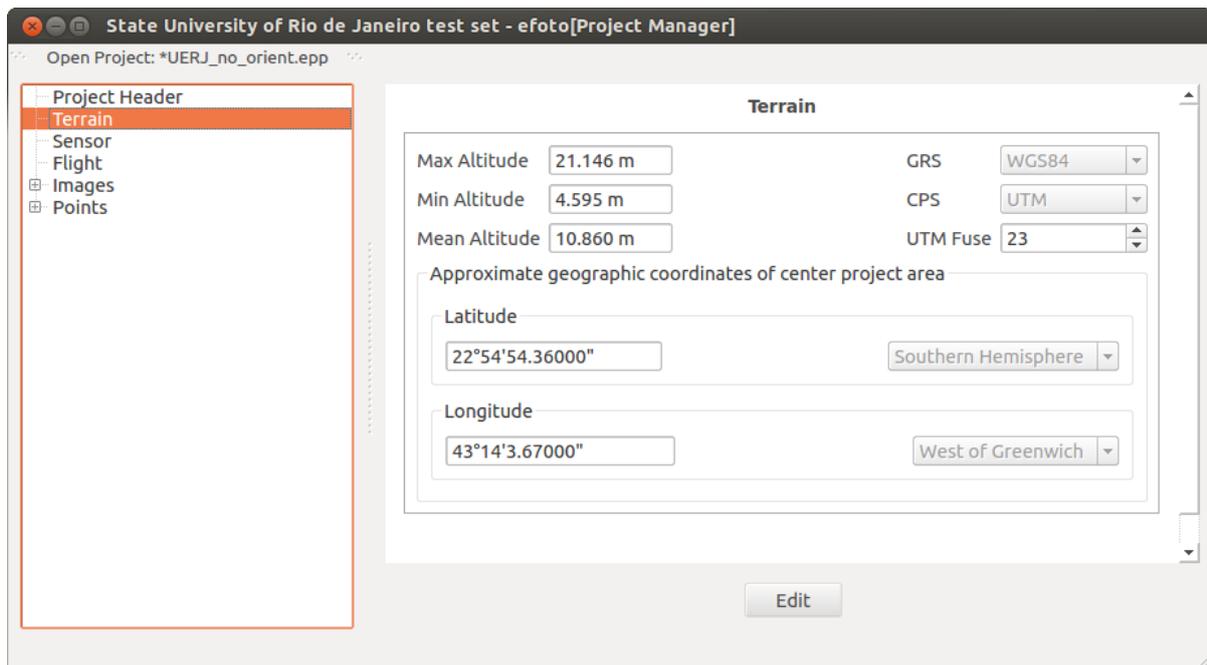
FIGURE 6.2. Entry of terrain parameters.

level, and are constant across typical photogrammetric mapping project areas.

CPS (Cartographic Projection System): pre-filled is UTM (Universal Transverse Mercator), with no alternatives provided.

UTM Fuse (UTM zone): this is the UTM zone number, counting Eastward from the *date line* (180° East, or West). Every zone is six degrees wide. The Western hemisphere comprises zones 1–30, the Eastern hemisphere zones 31–60. So, e.g., zone 35 runs from 24° to 30° with a central meridian of 27°, and zone 37 from 36° to 42° with a central meridian of 39°. It appears that e-foto fills in this field from the longitude data for the project centre point, discussed below.

Latitude (project area centre): enter numerical value in degrees, minutes and seconds.decimals. Choose the correct hemisphere from the pulldown menu.

Longitude (project area centre): enter numerical value in degrees, and again pick the correct hemisphere from the pulldown menu.

It is probably wise to set all these fields to roughly correct or correct values. Maximum, minimum and mean altitude (of the *terrain*, not the aircraft!) are pre-set to zero, and we have observed nothing adverse from

FIGURE 6.3. Sensor data entry form (part).

leaving them so set. Of course in mountainous or high-altitude areas one should enter more realistic values.

In fact, if you don't use an on-board GNSS receiver, probably none of the parameters on reference system, map projection and area location are relevant, and you can safely leave them unset. Your results will be in the same, locally rectangular, XYZ (or east-north-up) system as that in which the ground control point co-ordinates are given, and the software doesn't need to know what system that is — as long as *you* know...

6.3   Sensor

Here we enter first of all the camera description and properties, figure 6.3. The camera *focal length* and co-ordinates of the *principal point* are taken from the calibration certificate of the camera.

Then, lower on the same page (use the scroll bar on the right), we may enter the distortion model coefficients k_i, P_i according to the Brown-Conrady model (Brown, 1966), figure 6.4. Enter only if these have been
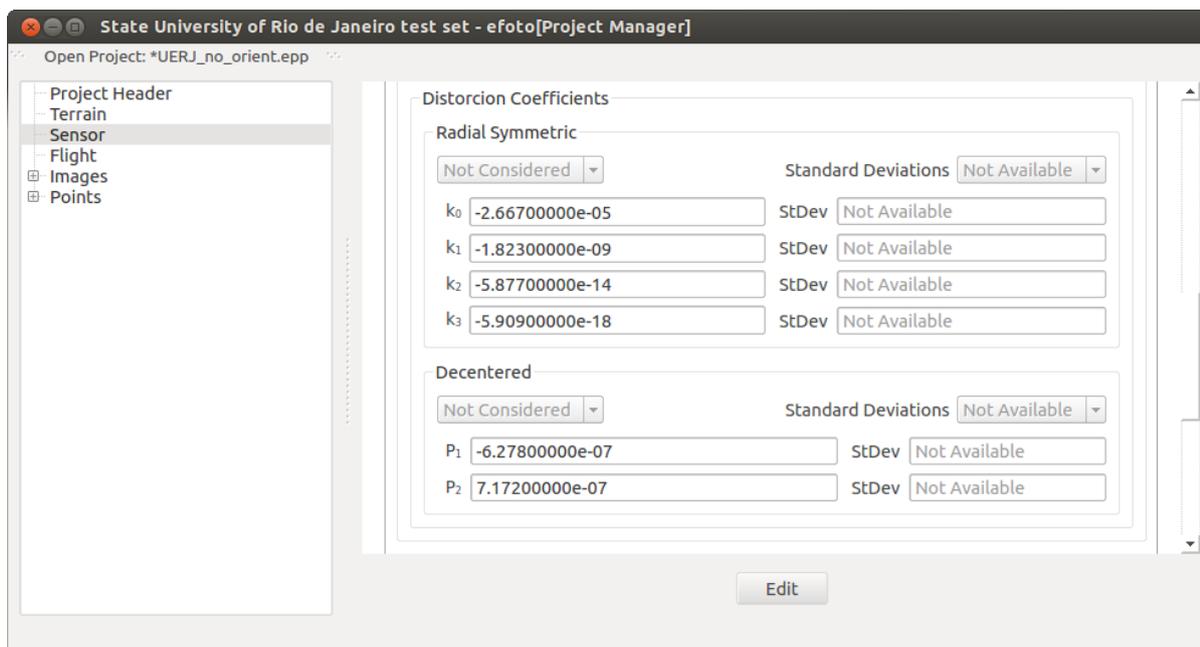
FIGURE 6.4. Entry of camera distortion coefficients.

determined for your camera.

Then, finally the camera co-ordinates of the *fiducial marks*, figure 6.5. These too are provided on the camera's calibration certificate, and are entered here.
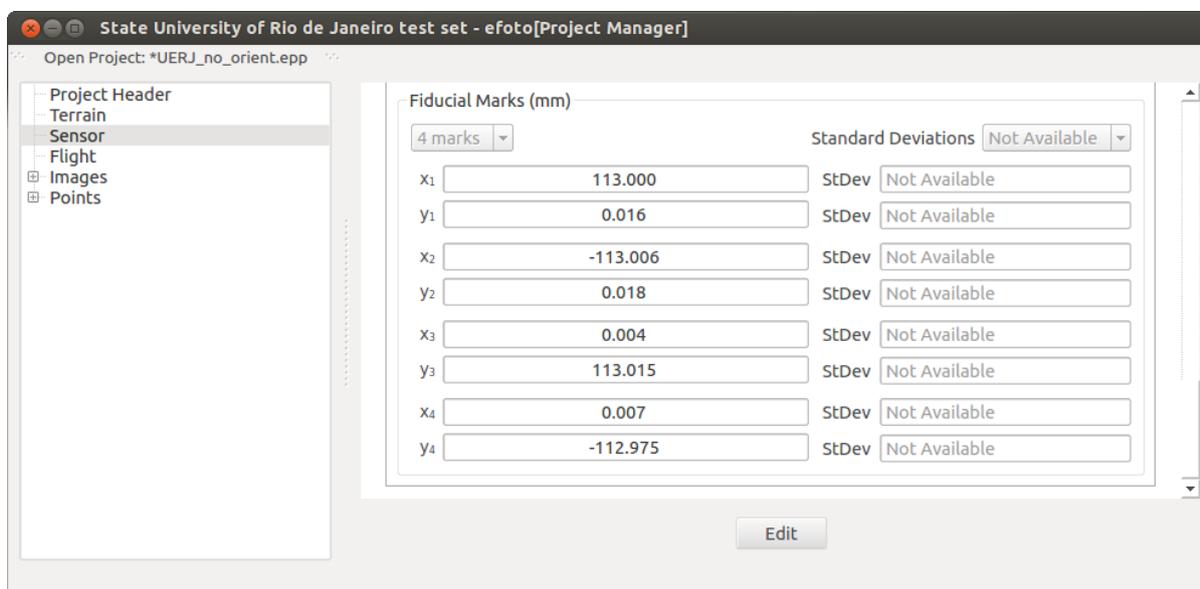
FIGURE 6.5. Entry of fiducial marks co-ordinates in camera frame, from the camera calibration certificate.

FIGURE 6.6. Flight information entry form.

6.4 Flight

Here is entered the photogrammetric flight information, to the extent
that it is known, figure 6.6.

The software uses only the Nominal Scale field for setting up the initial
approximate values of the geometry — the Nominal Flight Altitude value
is not actually used. It is probably wise to enter it though. The values
shown for longitudinal overlap (60 %) and for transversal overlap (30 %)
are typical for aerial mapping flights and are known to give good results.
Here you should enter the values actually used in this project's image
acquisition flight:

- Longitudinal Overlap is the overlap of two successive image frames
  obtained during the flight within a single strip of images,

- Transversal Overlap is the overlap between photographs in adjacent
  strips of images, with the flight direction of the aircraft typically
  turning in between.
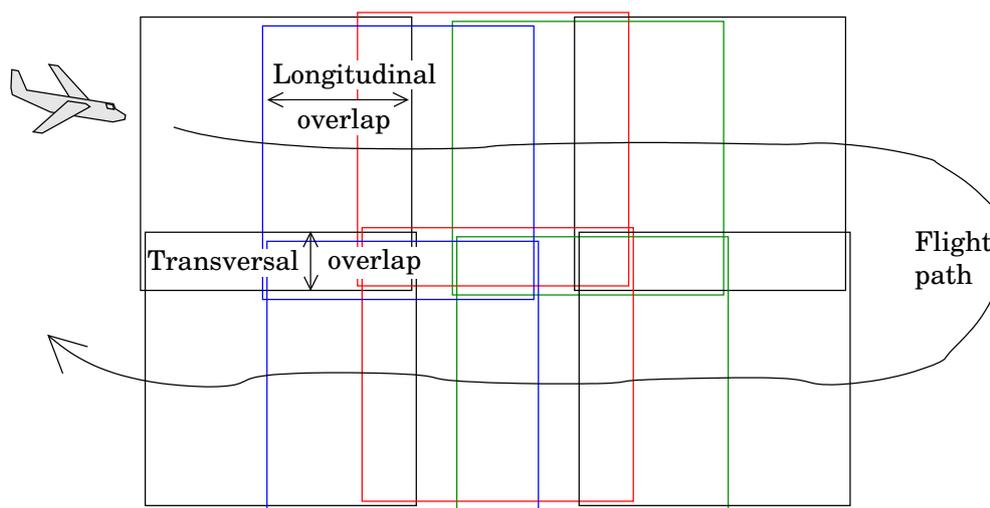
FIGURE 6.7. Flight geometry.

The flight geometry is depicted in figure 6.7.

6.5 Images

The Images tab allows the *importation* (i.e., setting the path names where to find them) of digital images. You can import the whole set in one go.

Note 1: When re-loading a project file, e-foto may complain that it cannot find the images you have just imported. If this happens, open the sub-tab for every image, click on the Edit button at the bottom, and use Select Image to re-set the path to that image. Press OK. After that, all should work OK.

Note 2: When loading a project file that was downloaded from the Internet — like the example project files provided by the e-foto project web page — you should also not expect these file pointers to be correct, as you will have placed your image file in a location on *your own* hard disk, which the project file does not know anything about. You will have to go through the Select image procedure for each image.

Note 3: If you are using the e-foto version of June 30, 2016, you need to *rotate* the images in a graphic editor like the GIMP. The angle of rotation should be the approximate exterior orientation angle κ. For more details, we refer to section 8.9 on exterior orientation.

As the execution of the various modules progresses, the Project Manager of e-foto keeps track of the work: clicking the Images tab will show you for
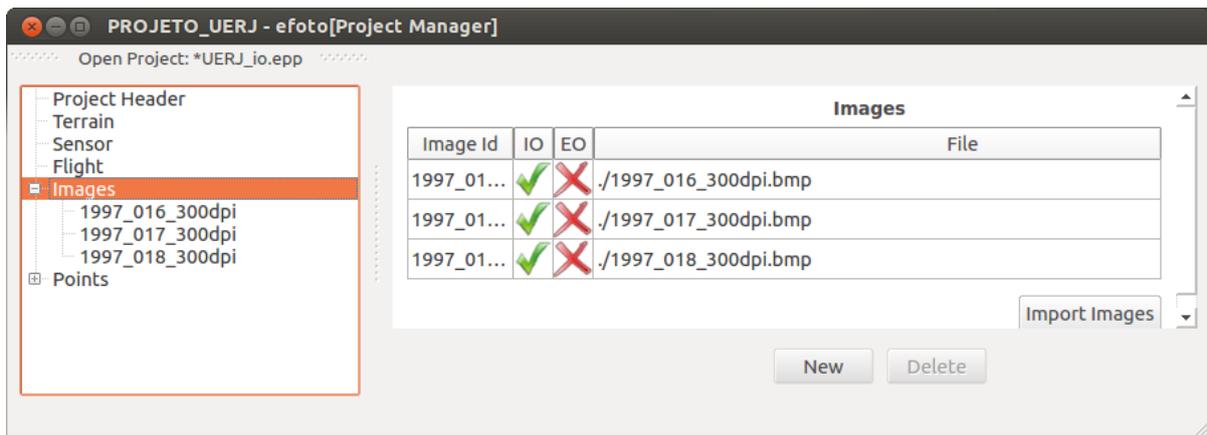
FIGURE 6.8. Project manager of e-foto. Here, interior orientation of three images has been completed, but exterior orientation has not yet begun.

which images *Interior* and *Exterior Orientation* modules were completed. In figure 6.8, interior orientation for all three images has been completed, but exterior orientation has not yet been started.

Click on every Images sub-tab individually to open up a form to be filled out *for only that image*: figure 6.9. You need to press the Edit button at the bottom before being able to fill out the from. Press OK when you are finished (or Cancel to start over again).

As you see, there are fields to enter data on where the images sit on your hard disk, and under that,

- ○ The position of the camera at exposure as determined by an on-board GNSS (Global Navigation Satellite Systems) receiver. If the aircraft has an on-board receiver, click on the button so a little cross appears. After that, you can choose whether the receiver was actually used, and whether the co-ordinates obtained should be used as an initial value to an iterative calculation of the camera position. And after that, the camera position co-ordinates measured, with their uncertainties ("StdDev", for standard deviation).

  Note that the co-ordinates to be entered are named E_0, N_0 and H_0, for "easting", "northing", and "height". These correspond to the exterior orientation camera co-ordinates X_0, Y_0, Z_0 which are listed in the lower box of the form. E-foto is not exactly consistent on notation...

- **Note 4:** For *older* versions of efoto (older than about 2014) you must use the following *workaround* for a bug in the software: If you don't have GPS or INS data with your images, make

FIGURE 6.9. Project entry form for an image.

sure to

1. *enable* the "Ground Coordinates. . ." click box; and

2. choose type: "Not Considered".

3. Do the same for Inertial Navigation System (INS) data, see below.

The need to do this is due to a bug / illogicity in these e-foto software versions, that has been corrected in more recent versions.

- The orientation of the camera at exposure, determined by an IMU (inertial measurement unit). Same remarks apply as for GNSS (i.e., following Note 4 above, if you have an older e-foto version, enable the click box, and choose "Not Considered").

- Metadata about the image file, and

- The results of interior and exterior orientation, or phototriangulation, if these are available already in the project file.

Technical remarks

You might think that if your images are in colour, you could convert them to greytone, which would make it easier for e-foto to handle them especially on a small laptop. However, although the files will be only one third of the original size, e-foto internally stores them still as 32 bits per pixel, so this doesn't really help much.

A better idea is to reduce the images to one quarter size by the following command:

```
$ convert <inputfile> -resize 50% <outputfile>
```

but remember that then you also lose half of your resolution on the ground. For training with the software this is OK, but not for production work. The convert command belongs to the ImageMagick package, which you need to have installed.

Since 2015, only 64-bit versions of the software are offered for download, both for Linux (Debian, Ubuntu) and for Windows.

6.6 Ground control points

Control points can be *loaded* from an external text file, if it is in the *right format*. Alternatively you can enter the data manually by clicking on the New button, and filling in the form that appears. You can *save* your data into a file, and inspect this file to see what the *right format* looks like. It is a simple text columns format that is apparently widely used by other photogrammetric software as well.

The point file format is:

1. Point number

2. Easting

3. Northing

4. Height

FIGURE 6.10. A filled-out Points tab for the Rio de Janeiro project.

and the fields are separated by a *tab symbol* (Control-I or ASCII 9). Remember also that the decimal separator should be a *point*, not a comma, as the e-foto software has not been localised at this point.

Technical remark: the co-ordinate data for the control points must be given *in this order*: first easting, then northing (like in mathematics, first x, then y). Geodesists do it the other way around: first northing (x), then easting (y). If your file is formatted like that, you need to interchange columns:

```
$ awk '{ t=$2 ; $2=$3; $3=t; print }' inputfile > outputfile
```

If the file contains even stranger stuff, like "southings" or "westings", you have a bigger editing job ahead.

6.7   E-foto, reference systems and map projections

As the e-foto software currently only supports a very limited number of geodetic reference systems and only one map projection (Universal Transverse Mercator, UTM), your options are limited if your ground control point co-ordinates are given in some traditional, local, non-geocentric datum. In Finland for example, ground control points may be given in KKJ, the National Map Grid Co-ordinate System, which uses a variant of

```
 ●▢  control_points_keha2_harjoitus.txt (~/bahir-dar/Finland) - VIM          ⊗
3723288^I42068.141^I73465.128^I2.668$
5723116^I42010.831^I73336.363^I14.69$
3623294^I41580.418^I73671.656^I10.848$
3622062^I41405.704^I74018.279^I24.977$
5623124^I41686.585^I73825.315^I8.54$
5623123^I41967.444^I73646.004^I13.03$
5622127^I41984.369^I74086.989^I5.29$
3622454^I41794.457^I74064.578^I12.441$
2623046^I41922.516^I73910.299^I8.415$
3723289^I42077.715^I73867.486^I11.168$
4722001^I42140.65^I74104.025^I11.341$
5622128^I41553.759^I74080.646^I29.28$
4623006^I41528.195^I73051.984^I34.466$
5623115^I41964.582^I7311 863^I9.09$
5624110^I41917.133^I72909.598^I14.63$
3624361^I41933.312^I72632.399^I2.75$
5624106^I41995.51^I72408.668^I1.63$
5624109^I41454.821^I72509.198^I14.37$
3624366^I41615.809^I72852.326^I16.653$
3624156^I41591.34^I72275.699^I2.77$
3624135^I41494.313^I72138.313^I8.222$
3724445^I42097.945^I72067.176^I7.28$
5624104^I41702.372^I72144.328^I9.78$
:set list                                          23,1          All
```

FIGURE 6.11. Ground control point list for the Finnish project in vi text editor.

TABLE 6.1. The ground control points in Bahir Dar. Note that the co-ordinates, obtained by hand-held GPS, are rounded to whole metres.

Number	Easting	Northing	Height
71	0321958	1280485	1812
72	0322001	1279905	1812
73	0321397	1280118	1806
74	0321448	1280648	1807
75	0321527	1281015	1804
76	0321540	1281256	1804
77	0321395	1281478	1805
78	0321341	1281824	1807
79	0321719	1281920	1798
80	0321272	1282164	1806
81	0321335	1282380	1800
82	0321551	1282293	1801
83	0322001	1282290	1804
84	0322067	1282126	1800
85	0321912	1281490	1807
86	0321913	1281156	1809
87	0321930	1280746	1813
88	0322087	1280866	1813

the Gauss-Krüger map projection on the International Ellipsoid of 1924 ("Hayford"), with central meridians of 18, 21, 24, 27, 30 and 33 degrees east. And note that often in local applications, *truncated co-ordinates* are used where leading digits have simply been dropped. . .

Similarly in Ethiopia we have the *Adindan datum*, on the Clarke 1880 (RGS) ellipsoid, using the UTM projection in zones 35, 36, 37 and 38.

In these cases, if you need to bring together ground control point and aircraft image exposure co-ordinates — the latter of which are typically given in WGS84 — one solution is to *transform* the ground control points co-ordinates to WGS84, and then to *project* them into a suitable UTM projection. Alternatively, transform the aircraft co-ordinates to the locally used datum, and project them using the map projection in local use. After that, e-foto will be able to work both with the ground co-ordinates given and with the aircraft co-ordinates provided by GNSS.

A suitable software for doing this transformation and re-projection is proj4, an open-source package that knows a great many different national and regional geodetic datums and map projections — in fact, the whole set compiled by the EPSG, the European Petroleum Survey Group, as documented at, e.g., (EPSG codes).

On the other hand, realistically speaking, these traditional geodetic datums are often non-geocentric and of limited precision, being based on measurements made before the satellite era. If this is the case, you wouldn't be able to use GNSS-provided aircraft co-ordinates as such anyway, but would have to allow the software always to adjust these to agree with the system the ground control points are in. But then, there is no real reason to go through a complicated transformation: just choose, in e-foto's Image tab, for every image, to *disable* the cross mark at "Ground Coordinates of exposure station centre (optical center of sensor)". Then, all computations will be done in local co-ordinates and the geodetic reference system and map projection settings will be ignored.

6.8   Concluding remarks

Although the Project Manager module allows the extensive editing of all project-related data, some things are not accessible to editing. E.g., there appears to be no way of making the system "forget" that an image was exteriorly oriented, or remove a control point from an image if it doesn't actually appear in that image, and was entered accidentally.

TABLE 6.2. Editing the project file, in order to "forget" an image's exterior orientation. Here it is image number ("key") 4 that is edited out *in two places*. This works for the version of June 2016.

```
<exteriorOrientation>
...
<imageEO type="spatialResection" image_key="4">
<Xa>
<X0 uom="#m">41736.60185948185</X0>
<Y0 uom="#m">73101.76725801102</Y0>
<Z0 uom="#m">621.2541973006183</Z0>
<phi uom="#rad">0.02360511107121352</phi>
<omega uom="#rad">-0.0150353187767421</omega>
<kappa uom="#rad">1.55154177028697</kappa>
</Xa>
</imageEO>
...
</exteriorOrientation>
```

```
<spatialResections>
...
<imageSR image_key="4">
<iterations>5</iterations>
<converged>truefalse</converged>
...
</imageSR>
...
</spatialResections>
```

Fortunately the project definition file — the one ending in .epp — is a simple text file, formatted as XML (eXtensible Mark-up Language), a format which is fairly straightforward to understand, and can be edited in a text file.

If you know what you are doing (and, just in case, take a back-up copy of the file), you can *edit* this file in an ordinary text editor. In Ubuntu Linux this means vim, emacs or gedit. See table 6.2.

Interior orientation

7
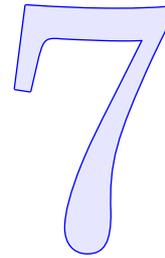
In a film image obtained with an analogue camera, in other words an emulsion on a glass or plastic film base, film co-ordinates (ξ, η) were traditionally measured, in metric units, in a co-ordinate measurement machine. These give the positions of points in the image obtained in a frame connected to the edges of the film. For a scanned image, i.e., an image *file*, each pixel has a pair of row and column numbers. These can be extracted by software, and converted to metric image co-ordinates using the known resolution (in pixels per mm) or pixel size (in μm) of the image. This effectively simulates a co-ordinate measuring machine.

xi $\xi \Xi$
eta ηH

Interior orientation is the process to express co-ordinates measured on the photographic plate or the scanned digital image, so-called *film* or *image co-ordinates*, from the image or film co-ordinate frame into the camera co-ordinate frame, so-called *camera* or *plate co-ordinates*. It establishes the *transformation* between these two co-ordinate frames. This is done by using the *fiducial marks*, markers all around the edge of the camera's image field, which are photographed onto the emulsion together with the terrain image at the moment of exposure. Typically there are either four or eight fiducial marks.

The origin of the camera frame is the *principal point*, the projection of the optical centre of the camera objective lens onto the image plane.

After the interior orientation transformation is derived, it may be used to convert the film or scanned-image co-ordinates (ξ, η) of *any* point in he image to rectangular metric camera co-ordinates (x, y). e-foto does this automatically in the background when one is measuring point co-ordinates in the imagery.
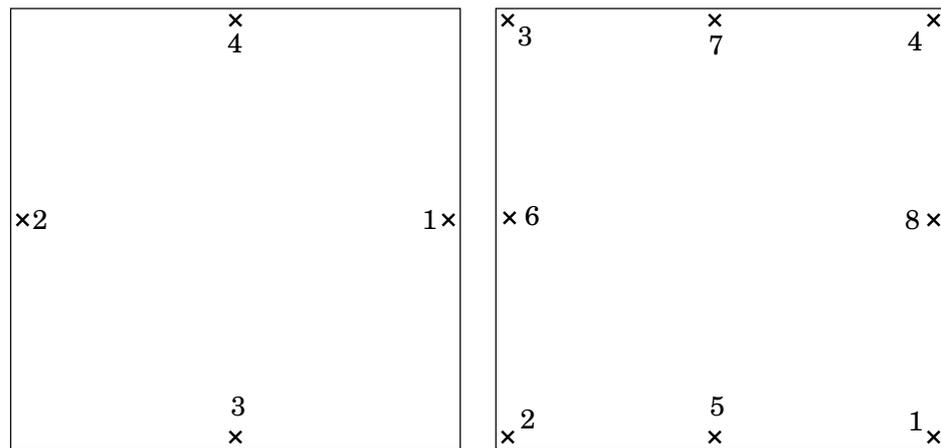
FIGURE 7.1. Fiducial marks of a camera. There may be four or eight fiducial marks.

Note on digital images

Nowadays the use of digital aerial cameras, obtaining digital imagery directly, is common. In such cameras, the image sensor, a CCD (charge-coupled device) array, is an integral part of the camera and rigidly attached to the camera body. For this reason, it is *not* necessary to perform interior orientation on digitally obtained imagery. Only *camera calibration* is needed, and serves for all images obtained with that camera.

If, however, one uses images that were *originally* obtained on film and then scanned, interior orientation *is* needed: the film roll moves — is transported forward — between exposures, and the precise position of the film in the camera image field will vary a little from exposure to exposure. This makes it necessary to *reconstruct* this precise position, using fiducial marks exposed onto the same image and rigidly attached to the camera body, the process known as interior orientation.

7.1 Theory

The relationship between

1. film co-ordinates (ξ, η) — typically obtained in a film measurement device, or in case of digital imagery, by dividing, in the photogrammetry workstation, the position in pixels in both co-ordinate directions by the known pixels-per-centimetre value of the image — and

2. camera co-ordinates (x, y) referring to the camera's principal point as the origin,

can be described in a number of ways, from the very simple to the very complex. A simple model would be a *similarity transformation*:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + K \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix}.$$

Here, the unknowns of the relationship are a scale factor K, a rotation angle θ, and a translation vector $\begin{bmatrix} x_0 & y_0 \end{bmatrix}^{\mathsf{T}}$. This vector describes the offset between the origin (like the bottom left corner) of the photographic plate or digital image co-ordinate system, and the *principal point* of the camera. The principal point is the orthogonal projection of the optical centre of the camera objective lens onto the image plane.

theta $\theta\Theta$

However, more common is an *affine transformation*:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + K \begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix},$$

usually written

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} a_0 \\ b_0 \end{bmatrix} + \begin{bmatrix} 1+a_1 & a_2 \\ b_1 & 1+b_2 \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix}$$

with six free parameters. This allows, besides a rotation and scaling, also for the image to be *skewed*, even in two directions. This means that it depicts a tiny square in pixel space as a tiny diamond or rectangle in the camera co-ordinate frame, and a tiny circle as a tiny ellipse.

In other words, the pixel sizes may be different in the x and y directions, and the directions of the rows and columns of pixels may not be precisely perpendicular to each other. Both of these distortions may happen if the film material is not absolutely stable but can stretch; or more likely, if the manufacturing precision of the film scanning device is less than perfect.

Note that, as both the measured film or digital-image co-ordinates and the camera co-ordinates are in metric units, and the axes of both are roughly parallel (with each other and with the edges of the image) it follows that all four, a_1, a_2, b_1, and b_2 are typically *small numbers*. The offsets a_0 and b_0 are larger, denoting the vectorial offset from image measurement origin (typically an image corner) to camera co-ordinate origin, the principal point in the middle of the image.

7.2 Execution

In preparation for interior orientation, the *co-ordinates of the fiducial marks* as given in the calibration certificate (e.g., table 4.1) must have
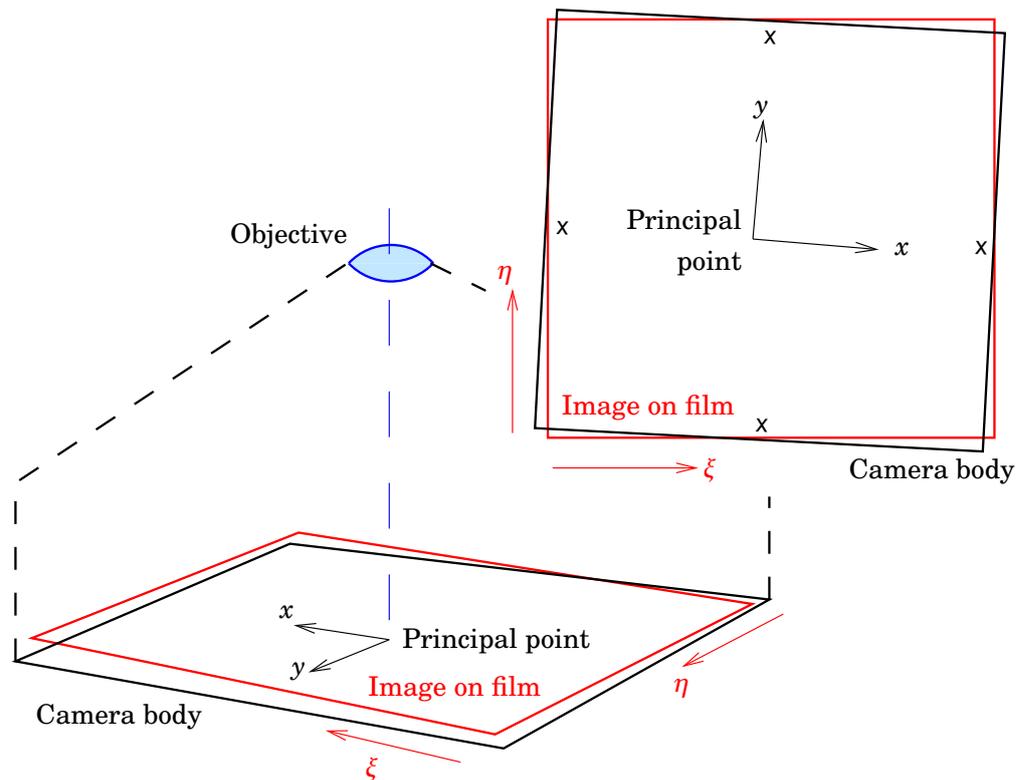
FIGURE 7.2. Camera principal point, camera co-ordinates (x, y), and film co-ordinates (ξ, η).

been entered — as must, of course, be all other pertinent camera calibration data, like the calibrated focal length and the radial-distortion table (table 4.2).

The data for executing the interior orientation is collected from the image in the software used; by first loading the image and its calibration data, and then placing the mouse cursor on each of the fiducial marks in turn. Clicking the mouse will register the film co-ordinates (ξ, η) for the fiducial mark in question; the camera co-ordinates (x, y) *for the same mark* were loaded from the calibration data.

As we have six unknown parameters, and four or eight fiducial marks, i.e., eight or sixteen conditions, the determination of the parameters is *overdetermined*. Software for interior orientation solves for the parameters by a *least-squares estimation* procedure. This procedure will produce, as outputs:

1. a table of *residuals*, one for each fiducial mark, showing how well the interior orientation solution obtained is able to place the fiducial marks of the oriented image on the fiducial marks of the camera's image plane — in other words, how well we have succeeded. The

presence of large residuals will indicate that we did something wrong.

2. A value for the *variance of unit weight* σ_0^2, as well as a value for its square root, the *standard error of unit weight*, also root-mean-square error or RMSE, σ_0. These numbers are also a measure for how well the interior orientation has succeeded. They too tell us, if they are unduly large, that we must have done something wrong.

sigma $\sigma\Sigma$

If we find a very large σ_0 value, the first thing to do is look at the table of residuals. Often the presence of a very large residual for one of the co-ordinates, or both, of one fiducial mark will indicate something wrong with that fiducial mark. Check for clerical errors in the calibration data entry; then, re-do the measurement of that point, and re-do the interior orientation. Now σ_0 should be suitably small. *Do not accept* your result until this is the case.

7.3 Interior orientation with e-foto

Interior orientation is the process to express the digital image's co-ordinate frame ("film co-ordinate frame") into the camera co-ordinate frame, deriving the transformation between the two co-ordinate frames, as described in detail above. This section will guide you through e-foto's Interior Orientation photogrammetric project stage, or module, in a step-by-step fashion. It requires a project file that was saved from the previous module, the Project Manager module.

7.3.1 Getting started

If you start the e-Foto software, the start screen comes up (figure 2.6). The main menu offers the options Project, Execute and Help. Using the Interior Orientation module requires that you load a project file created and saved by the Project Manager module: its new project setup process including the entry of the photogrammetric camera calibration certificate data, and entering the paths to the digital images to be used. If you haven't done so yet, we refer you to the tutorial on the Project Manager module.

1. Select the Project menu item from the main menu.

2. Load the previously saved project definition file using the "Load file", "Last project", or "Open file" option, depending on e-foto version. The file has an extension .epp. See figure 7.3.
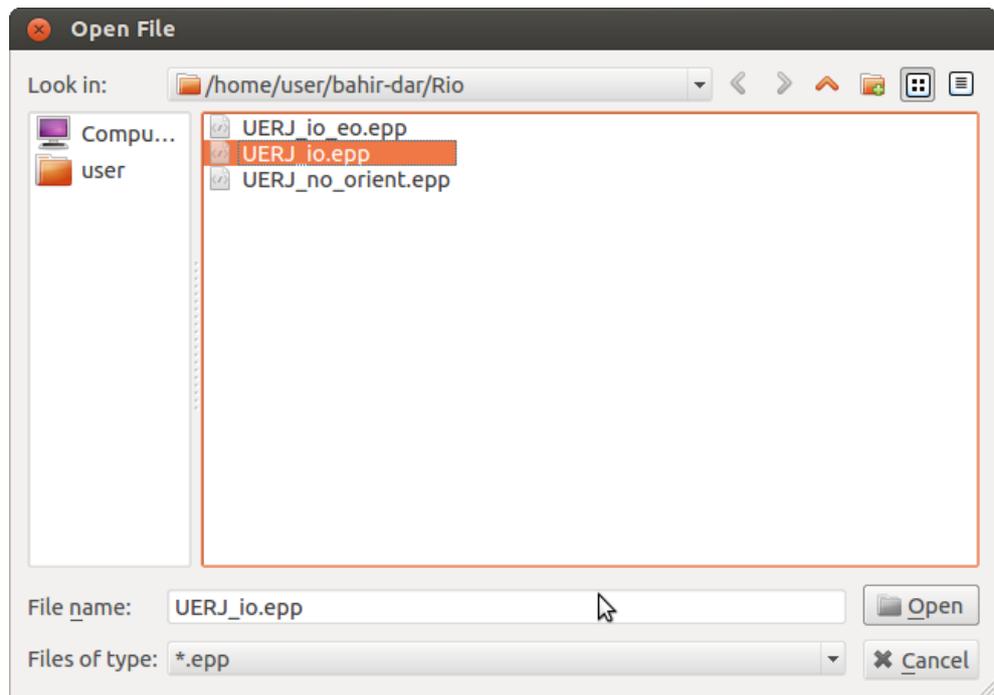
FIGURE 7.3. Loading a previously saved project file.

If you select the project file, the screen shown in figure 6.1 will come up.

Now choose Execute ▷ Interior Orientation. You're ready to go!

7.3.2   Starting interior orientation

A window will be presented asking you to select your image, figure 7.4. Please select the first image in your set (this will be the one presented for selection).

Next, you have to identify the *fiducial marks* according to their numbers.

Now you should move the mouse, and click, every fiducial mark in turn. The followings steps must be executed in sequence, for every fiducial mark:
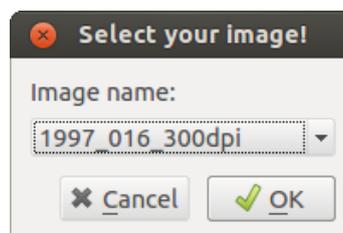
FIGURE 7.4. Select your image.
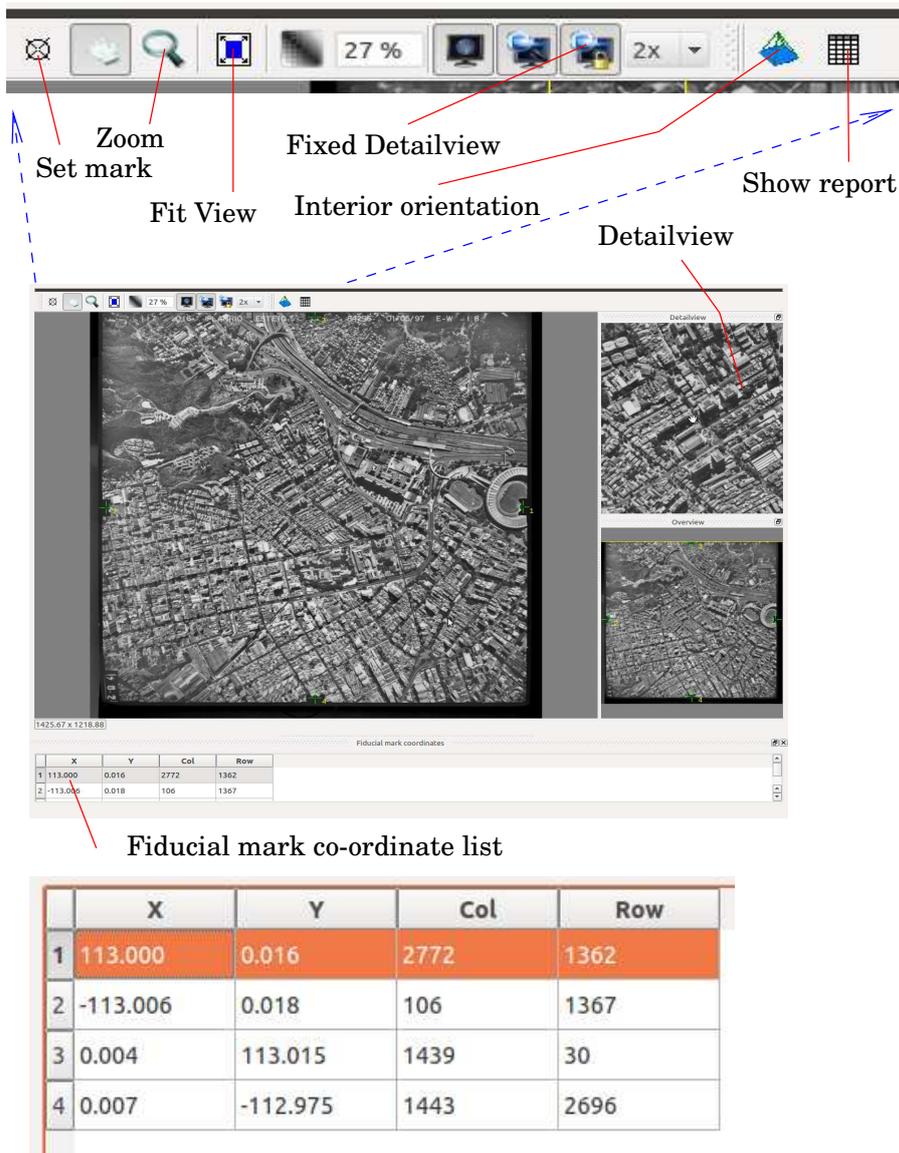
FIGURE 7.5. Interior orientation main view.

1. Click on the Zoom icon on the toolbar.

2. Make sure that the icon Fixed Detailview has been pressed down; click on it if needed.

3. In the big picture, move the mouse cursor to the current fiducial mark that you wish to measure.

4. Use the mouse to draw a small, yellow rectangle around the fiducial mark. Click the top left corner, pull the mouse, with button held down, to the bottom right corner — see how a little yellow rectangle forms —, and release mouse button. Now the fiducial mark and its surroundings should show up magnified on the screen.

FIGURE 7.6. Fiducial marks in the Detailview at 8× magnification: left an edge mark, right a corner mark.

5. *Verify* that, in the fiducial mark co-ordinate list, below the image, the row of the table belonging to the current fiducial mark is the active one; if not, *click* on the row you wish the co-ordinates to be entered into the fiducial mark co-ordinate list.

   Note: if you have measured a fiducial mark wrong, or not precisely enough, you can also go back and re-measure by clicking on the table row for that mark. The values in the row will be overwritten as you measure.

6. Now move the mouse cursor *precisely* on top of the fiducial mark, and *click*. The co-ordinate values for this fiducial mark are entered into the co-ordinate list below the image, and the list cursor jumps to the next row.

7. Clicking on the Fit View toolbar button should give you back the over-view of the full image, ready to proceed to the next fiducial-mark measurement.

Repeat for every fiducial mark, until all marks (four or eight) have been measured.

Then
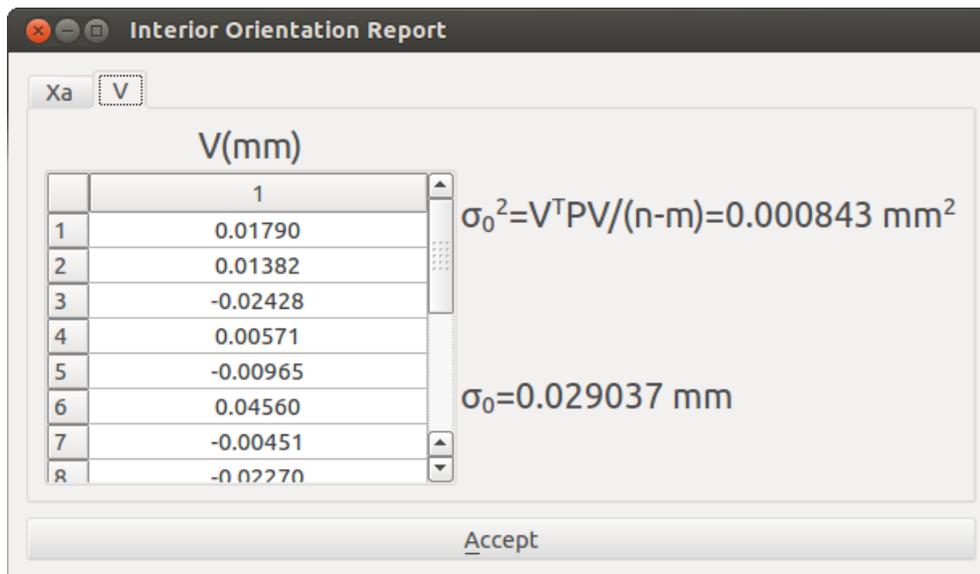
FIGURE 7.7. Interior orientation report, showing residuals.

1. Press the toolbar icon Interior Orientation. Note, in the output, the *standard error of unit weight* (also known as root-mean-square error or RMSE) σ_0. Your experience with this imagery should tell you how small this number can be if everything is correct, usually of order $0.01\,\text{mm}$.

2. Press the toolbar icon View report, and the tab "V". This will display the residuals of the fiducial mark co-ordinates. They should all be suitably small.

3. If you are happy, press Accept.

7.3.3 Completing interior orientation

- You have to repeat the interior orientation procedure for all images in your set.

- After every image, or whenever you need to interrupt your work, remember to *save* your intermediate results by exiting Interior Orientation (by closing its main window by clicking on the little cross in the corner) and then, in the main menu, clicking Project ▷ Save file.

- You may interrupt your work at any time. Interior orientation results that have been approved by clicking Accept in the Interior Orientation report, and after that, saved in the Project menu, will be preserved.

- If you are unhappy about the quality of some of your work (it improves with training!), you may return to Interior Orientation, open

the problem image, re-do the fiducial marks you are unhappy with — you may set the fiducial mark to be updated by clicking on the appropriate row in the fiducial mark co-ordinate list — re-accept and re-save.

Under the Project main menu item in the Images tab you can see which images have their interior orientation completed and saved into the project file, figure 6.8.

7.4 Debugging interior orientation

It is rather likely that, at first attempt, your interior orientation may not succeed and will produce a much too big value for σ_0, the root-mean-square error. There may be several reasons for this:

1. One or more of the values you entered from the camera calibration document into the co-ordinate table of fiducial marks (table 4.1) is in error.

2. Your measurement of the image co-ordinates of one or more of these fiducial marks is in error.

3. Your *identification* of the fiducial marks — that is the system of numbering used — was wrong, and the values you measured in the image, and the values tabulated in the calibration table, do not correspond. This will lead to huge residuals; check carefully that, in the images, the fiducial marks are indeed numbered as you assumed when measuring.

Carefully go through the numbers for each fiducial mark again, starting with the point producing the largest residuals. The RMSE value σ_0 should be small, e.g., order $0.01\,\text{mm}$ for the Finnish imagery; don't settle for anything bigger than this. Only once repeated attempts to bring the largest residual further down fail, is it time to move on. The quality of the results from the whole project depends on you getting this stage right.

Exterior orientation
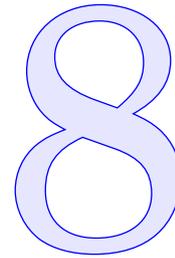
Exterior orientation, or *spatial resection*, refers to establishing the connection between terrain co-ordinates — three-dimensional, say (X, Y, Z), with the Z axis pointing up — and the camera co-ordinates (x, y) for one given image. It requires a sufficient number of *ground control points* $i = 1 \ldots n$ to be given both in the terrain, $(X_i, Y_i, Z_i), i = 1 \ldots n$, and on the photographic plate, $(\xi_i, \eta_i), i = 1 \ldots n$, or equivalently, in camera co-ordinates (x_i, y_i).

8.1 Theory

The unknowns, or elements, or parameters, to be recovered by exterior orientation are:

- The location co-ordinates (X_0, Y_0, Z_0) of the camera, in the same terrain co-ordinate system also used for the ground control points.

- The camera *orientation angles* $(\omega, \varphi, \kappa)$ also relative to the axes (X, Y, Z) of the terrain co-ordinate system, defined by a "corkscrew rule": ω rotates, around the X axis, Y towards Z; φ rotates, around the Y axis, Z towards X; and κ rotates, around the Z axis, X toward Y. The (X, Y, Z) axes triad is *right-handed*, i.e., a corkscrew turning from X to Y moves forward along the Z axis.

omega $\omega\Omega$
phi $\varphi\phi\Phi$
kappa κK

In the usual situation in aerial photography, where the camera looks straight down and the terrain underneath has only small height (elevation) differences, one can show that while recovery of the camera's height co-ordinate Z_0, and its rotation angle κ around the Z axis, can be done precisely, both the horizontal co-ordinates X_0, Y_0 and the tilt angles ω, φ can *not* be recovered very precisely: the recovered values are much more uncertain and are correlated or anti-correlated, X_0 with φ, Y_0 with ω. It is only because aerial cameras have such a wide view angle, that
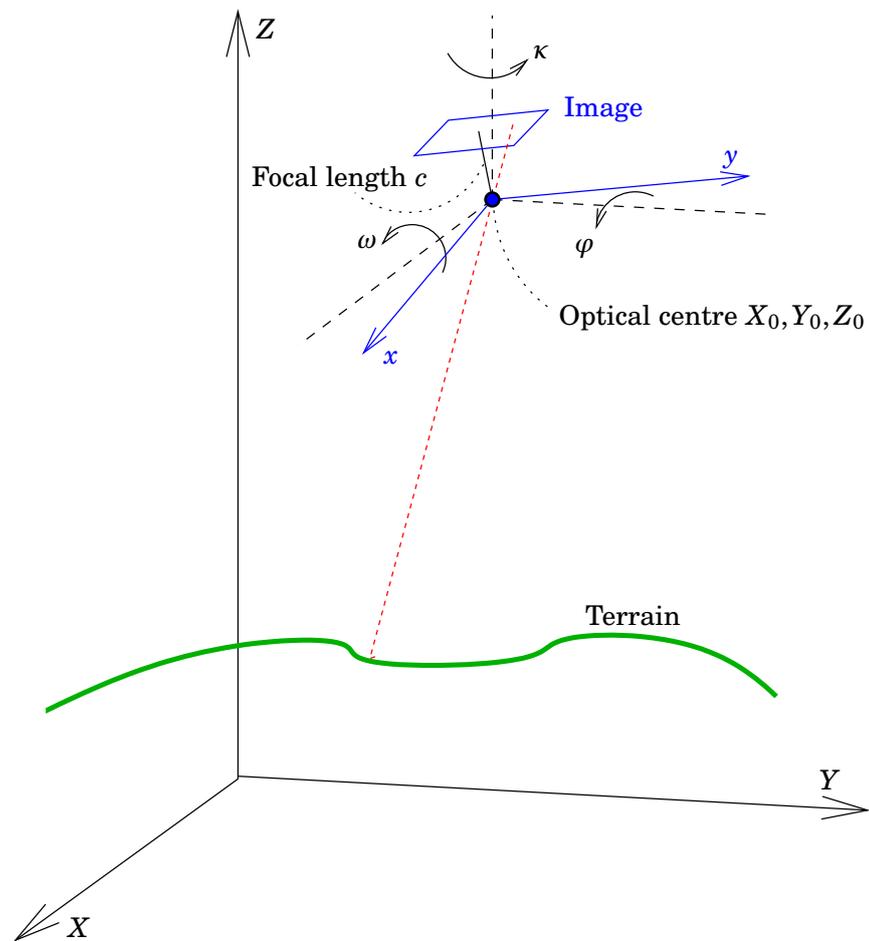
FIGURE 8.1. Exterior orientation elements. These together describe the relationship between the location of a terrain point in terrain co-ordinates, and the corresponding image point in camera co-ordinates.

a reasonable recovery of all the exterior orientation elements is at all possible.

The exterior orientation situation is depicted for two image acquisition moments in figure 8.2. Note that, as we have *six* unknown parameters to determine, the co-ordinates of three points should be sufficient, and four points is already overdetermined, allowing not only a solution, but a judgment on its quality as well, in the form of the *variance of unit weight* σ_0^2 coming out of the least-squares estimation computation.

8.2 Observation equations

The observation equations for exterior orientation are referred to as the *collinearity equations*.
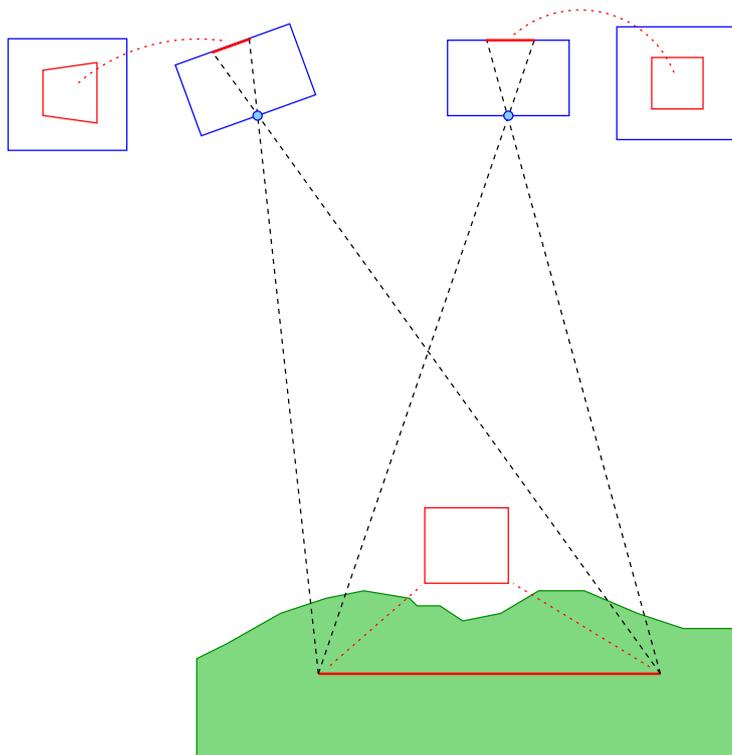
FIGURE 8.2. Exterior orientation of two images. A square in the terrain is
depicted as a square or as a trapezoid, depending on whether the
camera is looking down vertically (right) or obliquely (left).

We derive these equations by looking at a terrain point — $\mathbf{X}$ in terrain
co-ordinates — and the corresponding image point. They are connected
by a straight line through the camera optical centre, which, in terrain
co-ordinates, is given by the vector

$$\mathbf{X}_0 = \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}.$$

Similarly the image point is given in the camera co-ordinate frame:

$$\mathbf{x} = \begin{bmatrix} x \\ y \\ c \end{bmatrix},$$

where c is the calibrated focal length, and x, y the measured image co-
ordinates after reduction for camera calibration and interior orientation.

Now, in case the camera is looking straight down and the aircraft is
flying due East — with the camera operator sitting on starboard! — there
is the following relationship between these:

$$\mathbf{x} = -\frac{c}{Z_0 - Z} (\mathbf{X} - \mathbf{X}_0) = \frac{c}{Z - Z_0} (\mathbf{X} - \mathbf{X}_0),$$

where Z is the height of the terrain point above sea level, and Z_0 that of the exposure optical centre above sea level. Thus, $Z_0 - Z$ is the height of the camera *above the terrain*. The ratio $s = 1 : S \stackrel{\text{def}}{=} c/(Z_0 - Z)$ is an important number called the *scale factor* of the exposure. The inverse number $S = 1/s$, typically a large number, is called the *scale number*.

Of these three equations, only two are useful observation equations:

$$
\begin{aligned}
x &= c\frac{X - X_0}{Z - Z_0}, \\
y &= c\frac{Y - Y_0}{Z - Z_0}.
\end{aligned}
$$

In reality of course, the camera has an arbitrary, varying orientation relative to terrain co-ordinates, and we introduce *camera-rotated terrain co-ordinates* $\overline{\mathbf{X}}$ of the terrain point, as follows:[1]

$$
\overline{\mathbf{X}} = R\left(\mathbf{X} - \mathbf{X}_0\right),
$$

where R is the rotation matrix between terrain and camera co-ordinate axes, as follows:

$$
R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{e}_1^\mathsf{T} \\ \mathbf{e}_2^\mathsf{T} \\ \mathbf{e}_3^\mathsf{T} \end{bmatrix} = R(\omega, \varphi, \kappa).
$$

Here, we also expressed the rows of the matrix as column vectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$. As a rotation matrix is orthogonal (see appendix A.3), these vectors are unit vectors, and mutually ortogonal:

$$
\|\mathbf{e}_1\| = \|\mathbf{e}_2\| = \|\mathbf{e}_3\| = 1, \qquad \langle \mathbf{e}_1 \cdot \mathbf{e}_2 \rangle = \langle \mathbf{e}_1 \cdot \mathbf{e}_3 \rangle = \langle \mathbf{e}_2 \cdot \mathbf{e}_3 \rangle = 1.
$$

So, they form an orthonormal basis $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ of the space. They are in fact oriented along the axes of the (x, y, c) camera co-ordinate frame, as depicted in figure 8.1.

Now, for a realistic, $(\omega, \varphi, \kappa)$-rotated camera we obtain the *collinearity equations*:

$$
\begin{aligned}
x = c\frac{\overline{X}}{\overline{Z}} &= c\frac{\langle \mathbf{e}_1 \cdot (\mathbf{X} - \mathbf{X}_0) \rangle}{\langle \mathbf{e}_3 \cdot (\mathbf{X} - \mathbf{X}_0) \rangle} = \\
&= c\frac{r_{11}(\omega, \varphi, \kappa)(X - X_0) + r_{12}(\omega, \varphi, \kappa)(Y - Y_0) + r_{13}(\omega, \varphi, \kappa)(Z - Z_0)}{r_{31}(\omega, \varphi, \kappa)(X - X_0) + r_{32}(\omega, \varphi, \kappa)(Y - Y_0) + r_{33}(\omega, \varphi, \kappa)(Z - Z_0)},
\end{aligned}
$$

---

[1]The origin of these new co-ordinates is the camera optical centre, so formally one could say that $\overline{\mathbf{X}}_0 = 0$.

$$y = c\frac{\overline{Y}}{\overline{Z}} = c\frac{\langle \mathbf{e}_2 \cdot (\mathbf{X} - \mathbf{X}_0)\rangle}{\langle \mathbf{e}_3 \cdot (\mathbf{X} - \mathbf{X}_0)\rangle} =$$
$$= c\frac{r_{21}(\omega,\varphi,\kappa)(X - X_0) + r_{22}(\omega,\varphi,\kappa)(Y - Y_0) + r_{23}(\omega,\varphi,\kappa)(Z - Z_0)}{r_{31}(\omega,\varphi,\kappa)(X - X_0) + r_{32}(\omega,\varphi,\kappa)(Y - Y_0) + r_{33}(\omega,\varphi,\kappa)(Z - Z_0)}.$$

$$(8.1)$$

8.3 The rotation matrix

The rotation matrix R can be written in terms of the *Euler angles* (ω,φ,κ), which serve as the actual unknowns describing camera orientation. This is done as follows:

$$R = \begin{bmatrix} \cos\varphi\cos\kappa & \sin\omega\sin\varphi\cos\kappa + \cos\omega\sin\kappa & -\cos\omega\sin\varphi\cos\kappa + \sin\omega\sin\kappa \\ -\cos\varphi\sin\kappa & -\sin\omega\sin\varphi\sin\kappa + \cos\omega\cos\kappa & \cos\omega\sin\varphi\sin\kappa + \sin\omega\cos\kappa \\ \sin\varphi & -\sin\omega\cos\varphi & \cos\omega\cos\varphi \end{bmatrix}.$$

$$(8.2)$$

Substitution now yields the exact collinearity equations:

$$x = c\frac{f_x}{g}, \qquad\qquad y = c\frac{f_y}{g}, \qquad\qquad (8.3)$$

with the definitions[2]

$$f_x \stackrel{\text{def}}{=} \left\langle \overbrace{\begin{bmatrix} \cos\varphi\cos\kappa \\ \sin\omega\sin\varphi\cos\kappa + \cos\omega\sin\kappa \\ \sin\omega\sin\kappa - \cos\omega\sin\varphi\cos\kappa \end{bmatrix}}^{\mathbf{e}_1} \cdot \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \right\rangle =$$
$$= \cos\varphi\cos\kappa\,(X - X_0) + (\sin\omega\sin\varphi\cos\kappa + \cos\omega\sin\kappa)(Y - Y_0) +$$
$$+ (\sin\omega\sin\kappa - \cos\omega\sin\varphi\cos\kappa)(Z - Z_0),$$

$$f_y \stackrel{\text{def}}{=} \left\langle \overbrace{\begin{bmatrix} -\cos\varphi\sin\kappa \\ \cos\omega\cos\kappa - \sin\omega\sin\varphi\sin\kappa \\ \cos\omega\sin\varphi\sin\kappa + \sin\omega\cos\kappa \end{bmatrix}}^{\mathbf{e}_2} \cdot \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \right\rangle =$$
$$= -\cos\varphi\sin\kappa\,(X - X_0) + (\cos\omega\cos\kappa - \sin\omega\sin\varphi\sin\kappa)(Y - Y_0) +$$
$$+ (\cos\omega\sin\varphi\sin\kappa + \sin\omega\cos\kappa)(Z - Z_0),$$

$$g = \left\langle \overbrace{\begin{bmatrix} \sin\varphi \\ -\sin\omega\cos\varphi \\ \cos\omega\cos\varphi \end{bmatrix}}^{\mathbf{e}_3} \cdot \begin{bmatrix} X - X_0 \\ Y - Y_0 \\ Z - Z_0 \end{bmatrix} \right\rangle =$$
$$= \sin\varphi\,(X - X_0) - \sin\omega\cos\varphi\,(Y - Y_0) + \cos\omega\cos\varphi\,(Z - Z_0).$$

---

[2]The rows of the matrix R are here written as column vectors, just in order to save horizontal page space.

In the general case, this formulation has the problem of becoming unstable for certain combinations of angles. This is fortunately not the case for standard aerial vertical photogrammetry, where the angles φ and ω are close to zero. Then, we have

$$\cos\varphi \approx \cos\omega \approx 1, \qquad \sin\varphi \approx \varphi, \qquad \sin\omega \approx \omega, \qquad \sin\varphi\sin\omega \approx 0.$$

For the matrix above this means

$$R \approx \begin{bmatrix} \cos\kappa & \sin\kappa & \omega\sin\kappa - \varphi\cos\kappa \\ -\sin\kappa & \cos\kappa & \varphi\sin\kappa + \omega\cos\kappa \\ \varphi & -\omega & 1 \end{bmatrix}.$$

With this, the above equations 8.3 become

$$x \approx c\frac{\cos\kappa\cdot(X-X_0) + \sin\kappa\cdot(Y-Y_0) + (\omega\sin\kappa - \varphi\cos\kappa)(Z-Z_0)}{\varphi\cdot(X-X_0) - \omega\cdot(Y-Y_0) + (Z-Z_0)}, \quad (8.4)$$

$$y \approx c\frac{-\sin\kappa\cdot(X-X_0) + \cos\kappa\cdot(Y-Y_0) + (\varphi\sin\kappa + \omega\cos\kappa)(Z-Z_0)}{\varphi\cdot(X-X_0) - \omega\cdot(Y-Y_0) + (Z-Z_0)}. \quad (8.5)$$

This approximate simplification saves a lot of drudge when computing the partial derivatives for the design matrix A.

8.4   Linearisation

The expressions 8.4, 8.5 need to be *linearised* with respect to all six unknowns $X_0, Y_0, Z_0, \omega, \varphi, \kappa$. Doing this requires determining 2×6 partial derivatives $\partial x/\partial X_0 \ldots \partial y/\partial\kappa$. Deriving these partials, which form the design matrix A, is a lot of painstaking work but not really difficult.[3] The observation equations system (for a single point's two camera coordinates) then becomes

$$\begin{matrix} \Delta\underline{x} + \underline{v}_x \\ \Delta\underline{y} + \underline{v}_y \end{matrix} = \overbrace{\begin{bmatrix} \dfrac{\partial x}{\partial X_0} & \dfrac{\partial x}{\partial Y_0} & \dfrac{\partial x}{\partial Z_0} & \dfrac{\partial x}{\partial\omega} & \dfrac{\partial x}{\partial\varphi} & \dfrac{\partial x}{\partial\kappa} \\ \dfrac{\partial y}{\partial X_0} & \dfrac{\partial y}{\partial Y_0} & \dfrac{\partial y}{\partial Z_0} & \dfrac{\partial y}{\partial\omega} & \dfrac{\partial y}{\partial\varphi} & \dfrac{\partial y}{\partial\kappa} \end{bmatrix}}^{A} \begin{bmatrix} \Delta\widehat{X}_0 \\ \Delta\widehat{Y}_0 \\ \Delta\widehat{Z}_0 \\ \hline \Delta\widehat{\omega} \\ \Delta\widehat{\varphi} \\ \Delta\widehat{\kappa} \end{bmatrix}, \quad (8.6)$$

delta $\delta\Delta$ in which the Δ symbols refer to the differences of both observations x, y

---

[3]It can even be done numerically, avoiding the risk of error in an analytical derivation. This is a useful correctness check on your code.

and unknowns $X_0 \ldots \kappa$ from their assumed approximate values. The matrix of partial derivatives — to be evaluated at best available approximate values for the unknowns — is the *design matrix A*:

$$
A = \left[\begin{array}{ccc|ccc} \dfrac{\partial x}{\partial X_0} & \dfrac{\partial x}{\partial Y_0} & \dfrac{\partial x}{\partial Z_0} & \dfrac{\partial x}{\partial \omega} & \dfrac{\partial x}{\partial \varphi} & \dfrac{\partial x}{\partial \kappa} \\[2ex] \dfrac{\partial y}{\partial X_0} & \dfrac{\partial y}{\partial Y_0} & \dfrac{\partial y}{\partial Z_0} & \dfrac{\partial y}{\partial \omega} & \dfrac{\partial y}{\partial \varphi} & \dfrac{\partial y}{\partial \kappa} \end{array} \right]. \tag{8.7}
$$

In equation 8.6 we have marked the observations $\Delta\underline{x}, \Delta\underline{y}$ and their residuals $\underline{v}_x, \underline{v}_y$ as stochastic by adding an underline, and the unknowns $\Delta\widehat{X}_0 \ldots \Delta\widehat{\kappa}$ as estimators by adding a hat, as is formally correct in observation equations.

8.5 Computing the partials

Obtaining expressions for the partial derivatives making up the design matrix A, equation 8.7, is done by applying the *quotient rule* of differentiation to equations 8.4, 8.5: given two functions $f(p), g(p)$ of some parameter p, we have

$$
\frac{d}{dp}\left(\frac{f}{g}\right) = \frac{1}{g}\frac{df}{dp} - \frac{f}{g^2}\frac{dg}{dp}.
$$

We call, like above in equation 8.4,

$$
f_x \stackrel{\text{def}}{=} c\left(\cos\kappa \cdot (X - X_0) + \sin\kappa \cdot (Y - Y_0) + (\omega\sin\kappa - \varphi\cos\kappa)\right) \cdot (Z - Z_0),
$$

$$
g \stackrel{\text{def}}{=} \varphi \cdot (X - X_0) - \omega \cdot (Y - Y_0) + (Z - Z_0),
$$

so that $x = f_x / g$. Now we obtain, e.g., for $p \stackrel{\text{def}}{=} X_0$:

$$
\frac{\partial x}{\partial X_0} = -\frac{1}{g}c\cos\kappa + \frac{f_x}{g^2}\varphi,
$$

and for $p \stackrel{\text{def}}{=} \kappa$ (note that g does not contain κ):

$$
\frac{\partial x}{\partial \kappa} = -\frac{1}{g}c\left(\sin\kappa\,(X - X_0) + \cos\kappa\,(Y - Y_0) + (\omega\cos\kappa + \varphi\sin\kappa)(Z - Z_0)\right),
$$

and so forth. . .

8.6 Full observation equations

As we see in equation 8.6, a single ground control point photographed in the image produces two observation equations, with two observations x and y, but no less than six unknowns $X_0, Y_0, Z_0, \omega, \varphi, \kappa$. The design

matrix A containing the partial derivatives has size 2×6, two rows and six columns, equation 8.6, which we write like this:

$$\underline{\ell} + \underline{v} = A\widehat{x},$$

with

$$\underline{\ell} = \left[\begin{array}{c} \Delta \underline{x} \\ \Delta \underline{y} \end{array} \right], \quad \underline{v} = \left[\begin{array}{c} \underline{v}_x \\ \underline{v}_y \end{array} \right], \quad A = \left[\begin{array}{ccc|ccc} \frac{\partial x}{\partial X_0} & \frac{\partial x}{\partial Y_0} & \frac{\partial x}{\partial Z_0} & \frac{\partial x}{\partial \omega} & \frac{\partial x}{\partial \varphi} & \frac{\partial x}{\partial \kappa} \\ \frac{\partial y}{\partial X_0} & \frac{\partial y}{\partial Y_0} & \frac{\partial y}{\partial Z_0} & \frac{\partial y}{\partial \omega} & \frac{\partial y}{\partial \varphi} & \frac{\partial y}{\partial \kappa} \end{array} \right],$$

[4] and[4]

$$\widehat{x} = \left[\begin{array}{cccccc} \Delta \widehat{X}_0 & \Delta \widehat{Y}_0 & \Delta \widehat{Z}_0 & \Delta \widehat{\omega} & \Delta \widehat{\varphi} & \Delta \widehat{\kappa} \end{array} \right]^{\mathsf{T}}.$$

Now of course in reality we have more than one ground control point: if we have p control points, we will have an observation vector of length $n = 2p$, and the A matrix will be have two rows for each point, being $2p \times 6$ in size. If $2p \geq 6$, i.e., at least three control points, the problem will be well posed, and for $2p > 6$ even overdetermined.

As an example we give the observation matrix for the case of four ground control points:

$$\left[\begin{array}{c} \underline{\ell}_1 \\ \underline{\ell}_2 \\ \underline{\ell}_3 \\ \underline{\ell}_4 \end{array} \right] + \left[\begin{array}{c} \underline{v}_1 \\ \underline{v}_2 \\ \underline{v}_3 \\ \underline{v}_4 \end{array} \right] = \left[\begin{array}{c} A_1 \\ A_2 \\ A_3 \\ A_4 \end{array} \right] \widehat{x},$$

where

$$\underline{\ell}_i = \left[\begin{array}{c} \Delta \underline{x}_i \\ \Delta \underline{y}_i \end{array} \right], \quad \underline{v}_i = \left[\begin{array}{c} \underline{v}_{x,i} \\ \underline{v}_{y,i} \end{array} \right], \quad A_i = \left[\begin{array}{ccc|ccc} \frac{\partial x_i}{\partial X_0} & \frac{\partial x_i}{\partial Y_0} & \frac{\partial x_i}{\partial Z_0} & \frac{\partial x_i}{\partial \omega} & \frac{\partial x_i}{\partial \varphi} & \frac{\partial x_i}{\partial \kappa} \\ \frac{\partial y_i}{\partial X_0} & \frac{\partial y_i}{\partial Y_0} & \frac{\partial y_i}{\partial Z_0} & \frac{\partial y_i}{\partial \omega} & \frac{\partial y_i}{\partial \varphi} & \frac{\partial y_i}{\partial \kappa} \end{array} \right],$$

and $\widehat{x}$ as before, with i the ground control point number.

8.7   Initial approximate values for exterior orientation parameters

One might think that the above observation equations 8.4 and 8.5 could be further simplified, by the omission of ω and φ (being ≈ 0) to

$$x \approx c \frac{\cos \kappa \cdot (X - X_0) + \sin \kappa \cdot (Y - Y_0)}{(Z - Z_0)}, \tag{8.8}$$

$$y \approx c \frac{-\sin \kappa \cdot (X - X_0) + \cos \kappa \cdot (Y - Y_0)}{(Z - Z_0)}. \tag{8.9}$$

---

[4]Here, we write the column vector $\widehat{x}$ as a transposed row vector, in order to save vertical page space.

However, *this will not work* because then, in the linearised observation equations, all the partial derivatives in the columns belonging to φ and ω will vanish, and therefore it will not be possible to estimate values for $\Delta\widehat{\varphi}$ and $\Delta\widehat{\omega}$ — and *the least-squares solution will not converge* for these unknowns!

However, these equations are useful nevertheless: they allow us to derive a set of crude approximate values for the exterior orientation parameters, that can be used to start their iterative improvement. We write the simplified equations 8.8, 8.9 in matrix form:

$$
\begin{bmatrix} x \\ y \end{bmatrix} = \frac{c}{Z - Z_0} \begin{bmatrix} \cos\kappa & \sin\kappa \\ -\sin\kappa & \cos\kappa \end{bmatrix} \begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix}
$$

or, inverted, with the *scale number* $S \stackrel{\text{def}}{=} 1/s = (Z - Z_0)/c$:

$$
\begin{bmatrix} X - X_0 \\ Y - Y_0 \end{bmatrix} = S \begin{bmatrix} \cos\kappa & -\sin\kappa \\ \sin\kappa & \cos\kappa \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},
$$

or

$$
\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix} + \begin{bmatrix} S\cos\kappa & -S\sin\kappa \\ S\sin\kappa & S\cos\kappa \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},
$$

or, elaborately rearranging terms,[5]

$$
\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix} \begin{bmatrix} X_0 \\ S\cos\kappa \\ -S\sin\kappa \\ Y_0 \\ S\sin\kappa \\ S\cos\kappa \end{bmatrix}. \tag{8.10}
$$

We recognise equation 8.10 as a set of *observation equations*, with

$$
\ell = \begin{bmatrix} X \\ Y \end{bmatrix}
$$

the "observations": the horizontal terrain co-ordinates of a ground control point; the design matrix

$$
A = \begin{bmatrix} 1 & x & y & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & x & y \end{bmatrix};
$$

---

[5]An even simpler rearrangement would be

$$
\begin{bmatrix} X \\ Y \end{bmatrix} = \begin{bmatrix} 1 & 0 & x & -y \\ 0 & 1 & y & x \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \\ S\cos\kappa \\ S\sin\kappa \end{bmatrix}
$$

and the vector of unknowns

$$
\mathsf{x} = \left[\begin{array}{cccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{array} \right]^{\mathsf{T}} =
$$
$$
= \left[\begin{array}{cccccc} X_0 & S\cos\kappa & -S\sin\kappa & Y_0 & S\sin\kappa & S\cos\kappa \end{array} \right]^{\mathsf{T}}.
$$

We write these "observation equations" 8.10 for *all* ground control points $i = 1, \ldots, n$ within the image, and construct the ordinary least-squares solution $\mathsf{x} = (A^{\mathsf{T}}A)^{-1} A^{\mathsf{T}}\ell$. Then, the initial approximate values for the individual exterior orientation elements are extracted as follows:

$$
\begin{aligned}
X_0^{(0)} &= x_1, \\
Y_0^{(0)} &= x_4, \\
\kappa^{(0)} &= \mathrm{atan2}\left(-x_3, x_2\right) = \mathrm{atan2}\left(x_5, x_6\right), \\
Z_0^{(0)} &= \overline{Z} \pm c\sqrt{x_2^2 + x_3^2} = \overline{Z} \pm c\sqrt{x_5^2 + x_6^2},
\end{aligned}
\tag{8.11}
$$

and of course $\varphi^{(0)} = \omega^{(0)} = 0$.

In the $Z_0^{(0)}$ equation, the plus sign gives the physically realistic solu-
tion[6]. Here, $\overline{Z}$ is the average terrain height Z: we have made the further approximation that the terrain is relatively flat.

In the exterior orientation ("Spatial Resection") code of e-foto, part of the above expressions 8.11 are applied for setting the initial approximate values of the parameters.

---

with only four unknowns. Now

$$
\mathsf{x} = \left[\begin{array}{c} x_1 \\ x_2 \\ x_3 \\ x_4 \end{array} \right] = \left[\begin{array}{c} X_0 \\ Y_0 \\ S\cos\kappa \\ S\sin\kappa \end{array} \right]
$$

and

$$
\begin{aligned}
\kappa^{(0)} &= \mathrm{atan2}\left(x_4, x_3\right), \\
Z_0^{(0)} &= \overline{Z} + c\sqrt{x_3^2 + x_4^2}.
\end{aligned}
$$

This way of writing the problem is actually very similar to solving the parameters of a Helmert transformation in two dimensions, with $\left[\begin{array}{cc} X_0 & Y_0 \end{array} \right]^{\mathsf{T}}$ the translation vector, κ the rotation, and the flight height Z_0 as a "quasi-scaling".

[6]The fact that, mathematically, there is also a negative solution, explains why one sometimes sees near-convergence to an otherwise sensible-looking solution, with small residuals, but a negative flight height...

8.8   Exterior orientation using e-foto

Starting the e-foto software brings up the start screen, figure 2.6. The main menu, figure 2.7, offers the options Project, Execute and Help. Using the Exterior Orientation module requires that you load a project file created and saved by the Project Manager module, and modified by the Interior Orientation module. If you haven't done so yet, we refer you to the previous instructions on the Project Manager and Interior Orientation modules.

1. Select the Project menu item from the main menu.

2. Load the project definition file previously saved from the Interior Orientation module using one of the options "Load file", "Last project", or "Open file", depending on e-foto version. The file has an extension .epp. See figure 7.3.

If you select the project file, the screen shown in figure 6.1 will come up.

Now choose Execute ▷ Spatial Resection. You are ready to go!

The main screen that comes up is in figure 8.3. Note the toolbar, figure 8.4. In the point co-ordinate table, figure 8.5, ones sees the ground control points with originally only their terrain co-ordinates given.

8.9   Doing exterior orientation

Exterior orientation proceeds now as follows:

1. Enter the terrain point co-ordinates into a text file of suitable format, and load it into e-foto's Project Manager module.

2. Obtain a paper print of the photos or a map of the area on which the points are marked, so you can find them from the image.

3. For every point marker:

   (a) find it on the image on your screen using the information provided. It typically looks like a white square made of cardboard or painted on the roadtop, see the left picture in figure 8.7. Alternatively, the point is not clearly marked in the terrain but identified by a description, see the right picture in this figure.

   (b) Zoom if necessary using the Zoom toolbar button: click and drag the mouse to form a yellow outline that will be magnified

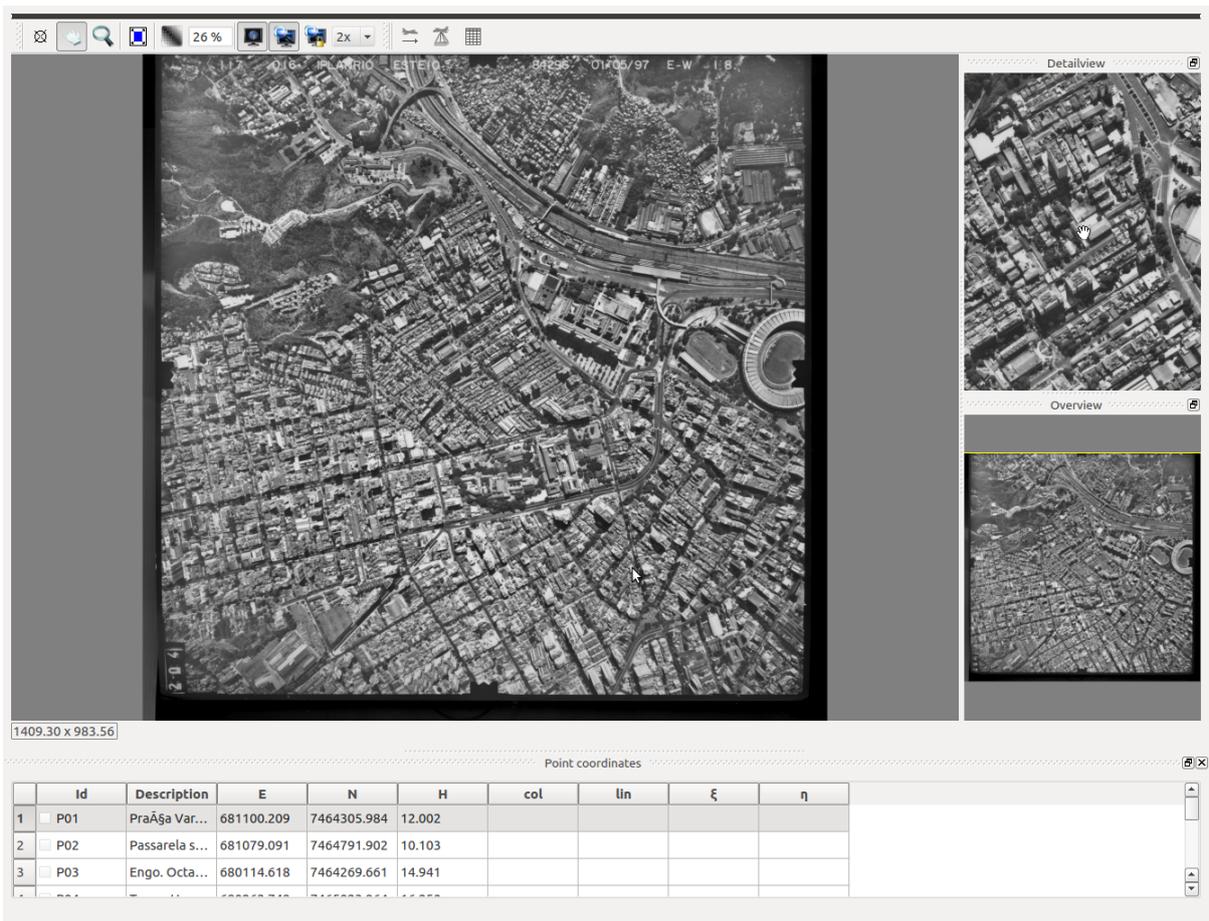   (c) activate point marking by clicking the "Measure mark" icon on the left of the toolbar

FIGURE 8.3. Exterior orientation main screen. Note the toolbar on top and the list of terrain points at the bottom.

(d) check once more that the point you are about to measure is the *active row* in the point co-ordinate table; click on the row number to activate the correct row if necessary

(e) place the crosshairs precisely over the point marker; if it is shown as big pixels, estimate the centre visually

(f) click.

(g) Like in interior orientation, also here you can *correct* a wrong

FIGURE 8.4. Exterior orientation toolbar.

	Id	Description	E	N	H	col	lin	ξ	η
1	■ P01	PraÃ§a Var...	681100.209	7464305.984	12.002				
2	□ P02	Passarela s...	681079.091	7464791.902	10.103				
3	□ P03	Engo. Octa...	680114.618	7464269.661	14.941				
4	□ P04	Torres Hom...	680262.749	7465023.964	16.250				
6	□ P08	Rio Maraca...	681875.866	7465066.854	5.980				

FIGURE 8.5. Point co-ordinates list.

measurement by activating the row for the point in the co-ordinate table, and re-measuring.

4. Now, you should do the exterior orientation computation. The icon for this will be greyed out; you should first tell the system the direction of flight associated with your images. Click the icon with a little aircraft and an arrow (Set flight), and follow instructions. You are presented with a little turnable dial (figure 8.6), the title of which refers to the "flight direction".

The angle that the dial sets is $\kappa^{(0)}$, the initial or approximate value of the κ angle of camera orientation around the Z axis.

This again is defined as the angle, counted positively (counterclockwise) from the terrain co-ordinate frame's X axis — which the e-foto software calls E, for easting — and the camera co-ordinate frame's x axis (or, as the e-foto software calls it, ξ). This latter axis points to the *right edge* of the image.
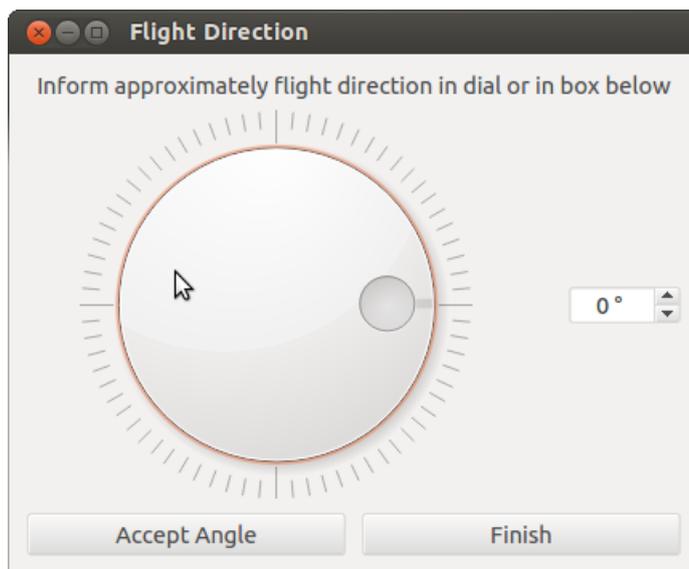
FIGURE 8.6. The "flight direction" — more accurately, "camera azimuth" — dial.

Now, whether the right edge of the image is on the side of the nose or the tail of the aircraft will depend on whether the camera operator sits on starboard or port with respect to the camera... as follows:

- If the operator sits on starboard, facing port, and the upper edge of the images is on port, κ will be the heading of the aircraft, counted counterclockwise *from the East*.

- If the operator sits on port, facing starboard, and the upper edge of the images is on starboard, then κ will be the aircraft heading reckoned counter-clockwise *from the West*.

... but of course you can always just experiment, until you find a $\kappa^{(0)}$ that works...

A better way of understanding κ is, that it is the angle over which the images are rotated, in the positive or counter-clockwise direction, from the "standard orientation" where the upper edge of the image is North, and the right edge East.[7] Perhaps "camera azimuth" would be a suitable name.

Note: if you are using the e-foto version from June 30, 2016, it doesn't actually have a flight direction dial. For this version, you have to *rotate the images before input* using a graphical editor. You have to do this in the Project Manager's Images tab, see section 6.5. The angle of rotation is the same $\kappa^{(0)}$ discussed above. This is necessary to achieve convergence of the exterior orientation iteration.

5. Now the Exterior orientation button is no longer greyed out. Click it, and check the results, figure 8.9. Again, the standard error of unit weight σ_0 tells you the general quality of the solution; it should be small. Check also the residuals — tab V —; they too should be small. And remember to check the unknowns — tab Xa in the report: these are the position co-ordinates X_0, Y_0, Z_0 of the camera optical centre, and the orientation angles ω, φ, κ of the camera. These values should all be *reasonable*. Reasonable values are, e.g., small values for two of the three angles (φ and ω), and a positive, realistic value for flight altitude Z_0.

The iteration should converge within five steps or so. If the iteration

[7]This means that e-foto's way of calling this dial "flight direction" is misleading: it may be *precisely opposite* to the direction of flight!

FIGURE 8.7. Detail with ground control point. Left, Helsinki, right, Rio de Janeiro.

doesn't converge in the ten steps allowed by default, there must be something wrong.

If you see huge, all-zero, or otherwise implausible values appear for any of the co-ordinates of the camera optical centre, this probably means that you have too few terrain points, or terrain points in unfavourable positions, or the flight direction dial setting wrong, or some other thing wrong: carefully check all the data you entered into the Project Manager module's many tabs.

6. If you are happy with the result, press Accept. Exit the Spatial Resection module (by clicking on the cross in the window's corner — or in later versions, on the green Done icon) and click Project ▷ Save file.

8.10   Completing exterior orientation

You have to exteriorly orient every image in your set. However, you may do it one image at a time, and save the accepted result to the project file.

It appears to be *difficult* in e-foto to re-do exterior orientation after the result has been accepted and saved. Therefore it might be a good idea to create a backup of your project definition file every time you complete one image.

Alternatively, the project definition file, with extension .epp, is in fact an XML file which is humanly readable. If you are very careful, you may
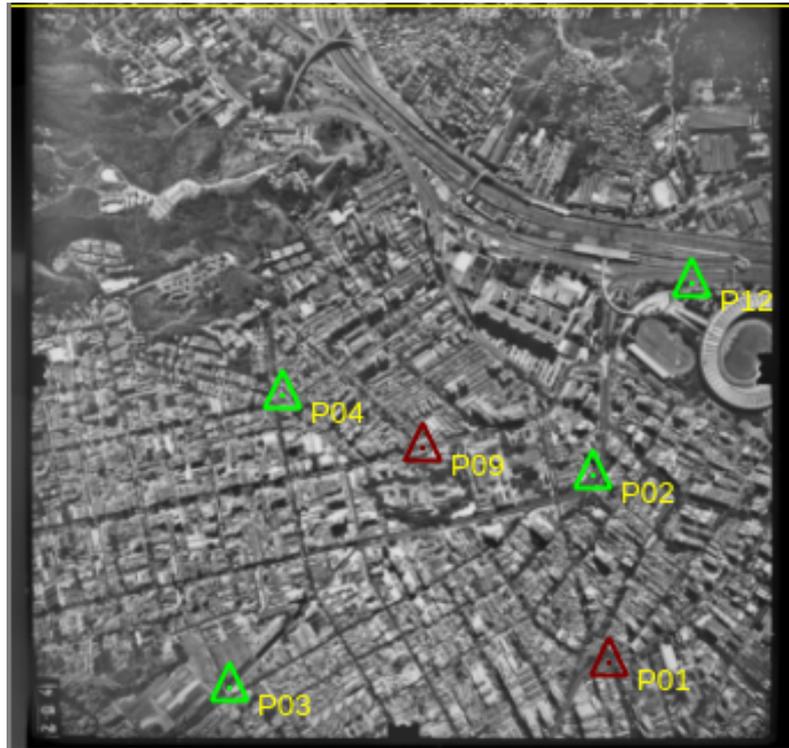
FIGURE 8.8. Overview picture of all terrain points in this image. Only the points marked green are activated and will be used for exterior orientation. The red points are deactivated. There is a tick box for activating or deactivating every point in the point co-ordinates list's leftmost column, see figure 8.5.
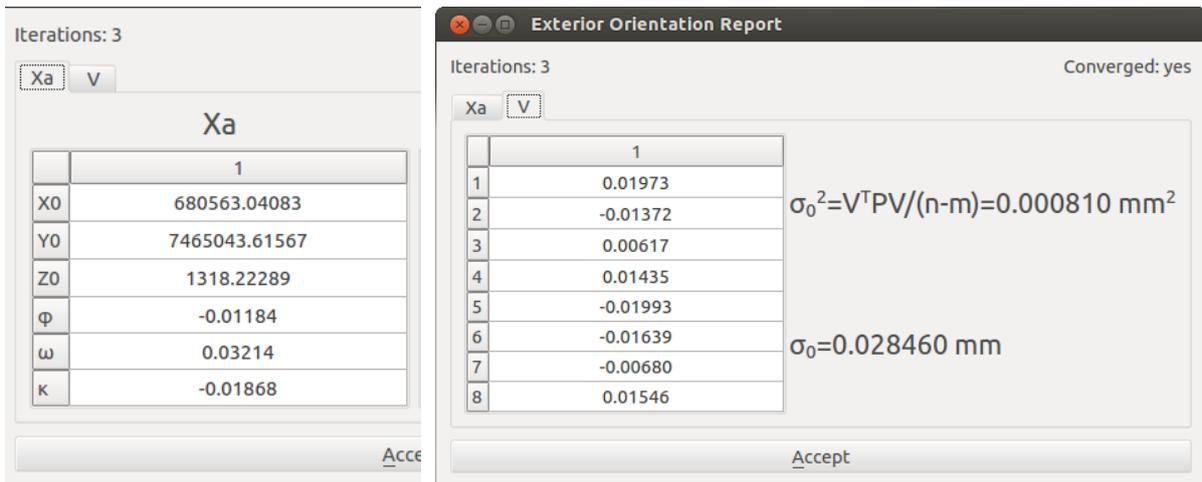

FIGURE 8.9. Exterior orientation adjustment report, displaying unknowns estimates (left) and residuals (right). If the number of iterations was insufficient for convergence, or you see large residuals, or implausible values in the Xa tab, do not accept yet.

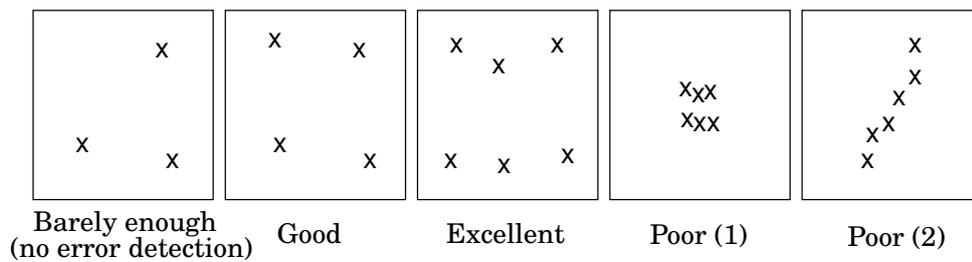|				
x x x	x x x x	x x x x x x	 xxx xxx	x x x x x x
Barely enough (no error detection)	Good	Excellent	Poor (1)	Poor (2)

FIGURE 8.10. Exterior orientation precision and ground control point geometry.

manually edit out the result of one image's Exterior Orientation. We will not give further instructions here, but what to do should be clear to anyone who can read XML files. Remember that a small error may lose you your work up to that point, so don't do this if you are unsure.

8.11   About geometric strength

Exterior orientation is done on a single image at a time, using ground control points (GCP) which appear on the image. The *precision* with which exterior orientation can be done and the exterior orientation parameters determined, depends on the geometry of these points within the area being imaged. This is depicted in figure 8.10. Note that three points is sufficient, but has no *overdetermination*, so a gross error in any of the control point co-ordinates will not be noticed: the solution is *not reliable*. This is generally not acceptable. Four points offer a minimum of overdetermination, provided no three points lie on a straight line.

On the right hand side of figure 8.10 are two examples of a *near-singular* or *ill-conditioned* geometry:

○ if all points are clustered together, a set of exterior orientation parameters will be obtained that represents only the immediate neighbourhood of this cluster well. At some distance, the quality of the solution becomes rapidly poorer.

○ If all points are on a straight line, then again close to this line the solution will be good, but away from the line, its quality gets poorer quickly.

8.12   Ground control points

Before an imaging flight, *ground control points* of known location must be marked or *signalised* in the terrain so they become visible in the images as marks that can be easily and precisely measured. Signal types used
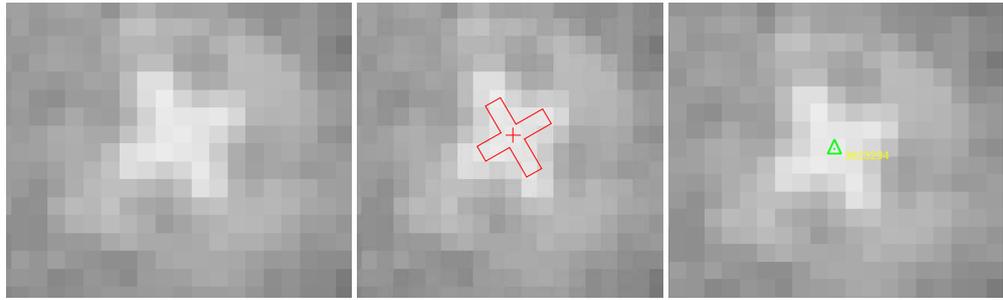
FIGURE 8.11. Ground control point: centring the marker.

are white discs or white crosses of sufficient size (depending on flying height or target resolution of the project). The discs or crosses can be painted on roadtop asphalt or manufactured from cardboard. Size will depend on the specified ground resolution.

It is important that the shape of the signal is symmetric, and that it is mounted symmetrically on the benchmark being signalised. Note that, with care, it is possible to measure the position of the mark in the image with sub-pixel accuracy.

Obtaining the co-ordinates of ground control points would nowadays always be done with GNSS. The proper technique to use to obtain precise positions this way is *static relative GNSS* using geodetic-quality (dual-frequency) receivers, with post-processing using geodetic GNSS processing software and precise satellite ephemeris.

The GNSS measurements should be connected to points known in a modern, nationally valid reference frame defined with the aid of space geodetic techniques and connected to the international reference frame. Any realisation of the ITRS will do; so will a national system like the Ethiopian Adindan datum, connected with ITRS through a conventionally fixed translation vector. The connection can be made

1. by including data collected by continuously operating reference stations (CORS) connected to the international network, or

2. by measuring on points the co-ordinates of which have already been determined, similarly by geodetic GNSS, in this system.

Co-ordinates thus obtained would be in some realisation of "WGS84" (or connected to it in some simple way) and be centimetre-level geocentric. They could also be readily converted to map projection co-ordinates, for example in the UTM projection.

In figure 8.11 is shown a magnification of the image around a control

point, that is marked on the ground with a white cross around the benchmark, a *signal*. Such signals are placed on control points in preparation for the flight. One sees that the pixel size is quite large compared to the size of the control point and its signal. Yet, it is important that the cursor is placed as precisely on the true point location as possible.

The figure shows how it is done. It requires some training and a little visual imagination. The middle sub-picture shows a replica in red of the white ground cross placed as well as possible to coincide with the pixellated cross in the image. In your mind's eye you should place this red cross on the image every time you mark a ground control point. Then, you choose the centre point of the red cross, and click to place the marker, see the rightmost sub-figure.

Note that for exterior orientation one must have a set of points with known three-dimensional co-ordinates in the terrain, all within the area appearing in this one image. On the other hand, photogrammetry by *aerotriangulation* is a way of determining three-dimensional point co-ordinates, but not from one image only; it requires *two or more* images to be used together. This brings us to the next, extensive subject, that of stereoscopic reconstitution of three-dimensional models from several overlapping images, or eventually a *block* of images covering the entire project area.

Aerotriangulation

9
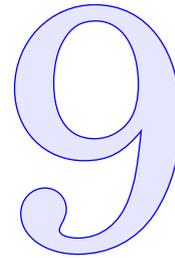
9.1 Description

Aerotriangulation (also known as phototriangulation) is the joint adjustment of a project set or *block* of aerial photographs, together with tie point information from the terrain, to obtain a three-dimensional model of the area mapped. It requires a number of preparatory steps, up to interior orientation for each image, the co-ordinates of points with known ground coordinates, so-called *ground control points* (GCP) visible in one or more of the images, as well as *tie points* which can be identified in overlapping images and serve to interconnect them.

Numerical aerotriangulation is typically performed by *bundle block adjustment*, a least-squares adjustment where the projective relationships between image and terrain co-ordinates serve as *observation equations*. An often used numerical technique, in view of the very large system of equations formed, is the iterative technique known as *conjugate gradients*, *steepest descent*, or an implementation known as the Levenberg-Marquardt algorithm (Wikipedia, Marquardt Algorithm). This algoritm is essentially a linearised, iterative solution algorithm stabilised by a technique similar to Tikhonov regularisation.

Contrary to exterior orientation, which is done one image at a time, aerotriangulation uses all images, and their ground control points, together. Therefore the situation may arise where some of the images cannot be externally oriented on their own, but, thanks to the addition of tie points, the block adjustment is successful. Much depends on the *geometry* of the tie points chosen, as well the overlap between the images in the block, both in the flight direction and perpendicular to it.

In figure 9.1 an aerial photography block is depicted where the overlap in the flight direction, between two images in the same strip, is approxi-
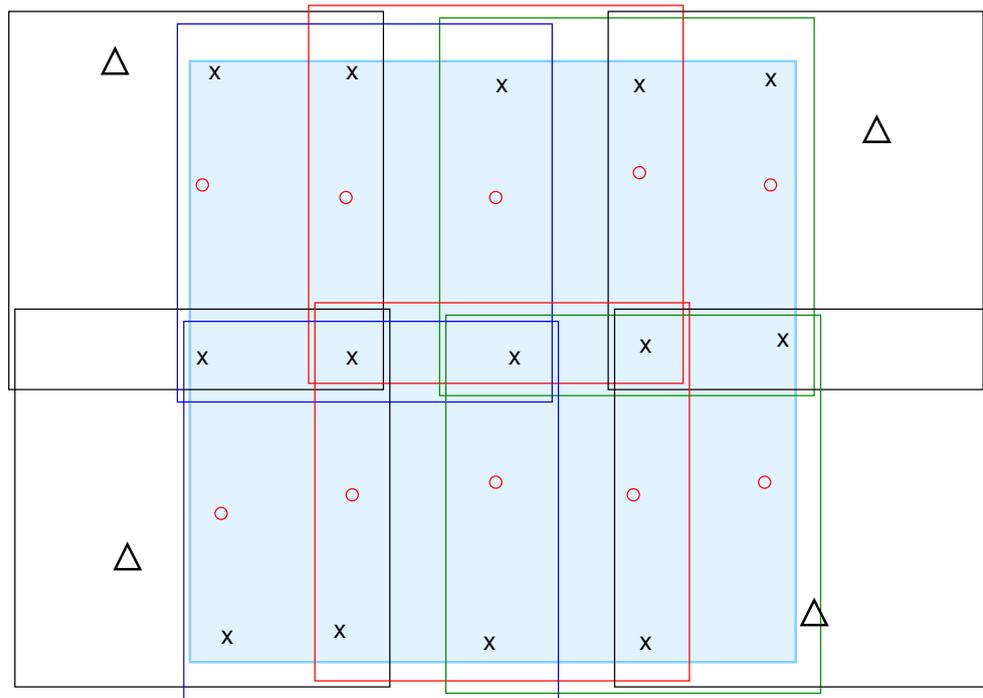
FIGURE 9.1. Tie-point geometry in aerotriangulation. The crosses form a tie point geometry that is sufficient; adding the circles will make it excellent. Also drawn are four ground control points (GCPs) as triangles.

mately 60 %, and in the perpendicular direction, between neighbouring strips, it is about 20 %. These values are fairly typical and give good results based on long experience.

Two tie point geometries are drawn:

1. a sufficient one in which there are four tie points between images in the same strip, and two between images in neighbouring strips; and

2. a great one where there are no less than *six* tie points between each pair of images in the flight direction.

Of course, existing ground control points can do double duty as tie points, and one should not choose tie points close to them. Good tie points are, e.g., the corners of building roofs which are well defined in both photographs. Again, *we do not need to know the ground co-ordinates of the tie points*. It suffices that they are identical in both (or all three, four) images in which they appear.

If the tie point geometry of a photogrammetric block is sufficient, or even good, this means that the *relative orientations* of the images within

the block — that is, the relative locations of their exposure centres, and the orientation angles of the cameras at exposure — are well constrained and can be well determined by the aerotriangulation.

However, *absolute orientation* of the block as a whole requires in addition *ground control points*. See again figure 9.1, where four proposed ground control points (GCPs) are drawn as triangles. Four points are sufficient, as they are for the exterior orientation of a single image (chapter 8), and provide some redundancy. Theoretically, the absolute minimum number of GCPs to constrain the six parameters of absolute orientation of the block is three points not on a straight line. The lack of redundancy makes this unacceptable, however.

Modern photogrammetric software like ERDAS Imagine can generate lots of tie points automatically using advanced image correlation techniques. Unfortunately e-foto does not have this capability.

9.2   Observation equations

The observation equations are similar to the ones we already derived for exterior orientation. However, note that now there is a much greater number of unknowns: *every camera location* on which an image was taken now contributes three unknowns of place (X_{0i}, Y_{0i}, Z_{0i}) and three unknowns of camera attitude $(\omega_i, \varphi_i, \kappa_i)$, where the index $i = 1, \ldots, p$ is the image number.

omega $\omega\Omega$
phi $\varphi\phi\Phi$
kappa $\kappa\mathrm{K}$

We can now write the observation equations as follows, giving the example of *two images A* and *B* containing *three ground control points*, one of which — point number 2 — appears in both images:

$$\overbrace{\begin{bmatrix} \underline{\ell}_{1A} \\ \underline{\ell}_{2A} \\ \underline{\ell}_{2B} \\ \underline{\ell}_{3B} \end{bmatrix}}^{\underline{\ell}}_{8\times 1} + \overbrace{\begin{bmatrix} \underline{v}_{1A} \\ \underline{v}_{2A} \\ \underline{v}_{2B} \\ \underline{v}_{3B} \end{bmatrix}}^{\underline{v}}_{8\times 1} = \overbrace{\begin{bmatrix} A_{1A} & 0 \\ A_{2A} & 0 \\ 0 & A_{2B} \\ 0 & A_{3B} \end{bmatrix}}^{A}_{8\times 12} \overbrace{\begin{bmatrix} \widehat{\mathrm{x}}_A \\ \widehat{\mathrm{x}}_B \end{bmatrix}}^{\widehat{\mathrm{x}}}_{12\times 1}. \tag{9.1}$$

Here, each observation $\underline{\ell}_{ji}$ stands for the *linearised* camera co-ordinates $\left(\Delta\underline{x}_{ji}, \Delta\underline{y}_{ji}\right)$, measured for ground control point j in image i. It thus consists of two real-number elements, and the total $\underline{\ell}$ vector on the left-hand side — as, similarly, the vector of residuals $\underline{v}$ — is eight real-number elements tall.

The partial design matrix or submatrix for every ground control point j and image i has now two rows and six columns, as follows:

$$A_{ji} = \left[\begin{array}{ccc|ccc} \dfrac{\partial x_{ji}}{\partial X_{0,i}} & \dfrac{\partial x_{ji}}{\partial Y_{0,i}} & \dfrac{\partial x_{ji}}{\partial Z_{0,i}} & \dfrac{\partial x_{ji}}{\partial \omega_i} & \dfrac{\partial x_{ji}}{\partial \varphi_i} & \dfrac{\partial x_{ji}}{\partial \kappa_i} \\[2ex] \dfrac{\partial y_{ji}}{\partial X_{0,i}} & \dfrac{\partial y_{ji}}{\partial Y_{0,i}} & \dfrac{\partial y_{ji}}{\partial Z_{0,i}} & \dfrac{\partial y_{ji}}{\partial \omega_i} & \dfrac{\partial y_{ji}}{\partial \varphi_i} & \dfrac{\partial y_{ji}}{\partial \kappa_i} \end{array} \right].$$

Here, the *unlinearised* camera co-ordinates — approximate, and only to be used for doing the partial derivations! — are given by the camera model equations

$$\begin{aligned}
x_{ji} &\approx c \frac{\cos\kappa_i\,(X_j - X_{0,i}) + \sin\kappa_i\,(Y_j - Y_{0,i}) + (\omega_i\sin\kappa_i - \varphi_i\cos\kappa_i)\,(Z_j - Z_{0,i})}{\varphi_i\,(X_j - X_{0,i}) - \omega_i\,(Y_j - Y_{0,i}) + (Z_j - Z_{0,i})}, \\
y_{ji} &\approx c \frac{-\sin\kappa_i\,(X_j - X_{0,i}) + \cos\kappa_i\,(Y_j - Y_{0,i}) + (\varphi_i\sin\kappa_i + \omega_i\cos\kappa_i)\,(Z_j - Z_{0,i})}{\varphi_i\,(X_j - X_{0,i}) - \omega_i\,(Y_j - Y_{0,i}) + (Z_j - Z_{0,i})},
\end{aligned} \tag{9.2}$$

according to equations 8.4, 8.5. As always, the partial derivatives must be evaluated for the *approximate values* used in the linearisation.

The full matrix A now consists of these submatrices of size 2×6. In our example equation 9.1 there are four such submatrices, as well as four submatrices of the same size 2×6 containing zeroes.

The vector of unknowns is, like before, containing *linearised* quantities:

$$\widehat{\mathsf{x}} = \left[\begin{array}{c} \widehat{\mathsf{x}}_A \\ \widehat{\mathsf{x}}_B \end{array} \right], \qquad \text{with } \widehat{\mathsf{x}}_A = \left[\begin{array}{c} \Delta\widehat{X}_{0A} \\ \Delta\widehat{Y}_{0A} \\ \Delta\widehat{Z}_{0A} \\ \hline \Delta\widehat{\omega}_A \\ \Delta\widehat{\varphi}_A \\ \Delta\widehat{\kappa}_A \end{array} \right], \qquad \widehat{\mathsf{x}}_B = \left[\begin{array}{c} \Delta\widehat{X}_{0B} \\ \Delta\widehat{Y}_{0B} \\ \Delta\widehat{Z}_{0B} \\ \hline \Delta\widehat{\omega}_B \\ \Delta\widehat{\varphi}_B \\ \Delta\widehat{\kappa}_B \end{array} \right].$$

Linearisation means that we choose *approximate values* for the unknowns, like in subsection 2.4.3:

$$\mathsf{x}^{(0)} = \left[\begin{array}{c} \mathsf{x}_A^{(0)} \\ \mathsf{x}_B^{(0)} \end{array} \right], \qquad \text{with } \mathsf{x}_A^{(0)} = \left[\begin{array}{c} (X_0)_A^{(0)} \\ (Y_0)_A^{(0)} \\ (Z_0)_A^{(0)} \\ \hline \omega_A^{(0)} \\ \varphi_A^{(0)} \\ \kappa_A^{(0)} \end{array} \right], \qquad \mathsf{x}_B^{(0)} = \left[\begin{array}{c} (X_0)_B^{(0)} \\ (Y_0)_B^{(0)} \\ (Z_0)_B^{(0)} \\ \hline \omega_B^{(0)} \\ \varphi_B^{(0)} \\ \kappa_B^{(0)} \end{array} \right].$$

Then, for example,

$$\begin{aligned}
\widehat{X}_{0A} &= (X_0)_A^{(0)} + \Delta\widehat{X}_{0A}, \\
\widehat{\varphi}_B &= \varphi_B^{(0)} + \Delta\widehat{\varphi}_B,
\end{aligned}$$

and so on. Now, the approximate values for the observations are obtained by propagating the approximate unknowns through the *non-linear, exact* camera model equations, based on the *exact* matrix R given in equation 8.2:

$$x_{ji}^{(0)} = \frac{(f_x)_{ji}^{(0)}}{g_{ji}^{(0)}}, \qquad\qquad y_{ji}^{(0)} = \frac{(f_y)_{ji}^{(0)}}{g_{ji}^{(0)}},$$

with

$$(f_x)_{ji}^{(0)} =$$

$$= \left\langle \begin{bmatrix} \cos\varphi_i^{(0)}\cos\kappa_i^{(0)} \\ \sin\omega_i^{(0)}\sin\varphi_i^{(0)}\cos\kappa_i^{(0)} + \cos\omega_i^{(0)}\sin\kappa_i^{(0)} \\ \sin\omega_i^{(0)}\sin\kappa_i^{(0)} - \cos\omega_i^{(0)}\sin\varphi_i^{(0)}\cos\kappa_i^{(0)} \end{bmatrix} \cdot \begin{bmatrix} X_j - X_{0,i}^{(0)} \\ Y_j - Y_{0,i}^{(0)} \\ Z_j - Z_{0,i}^{(0)} \end{bmatrix} \right\rangle =$$

$$= \cos\varphi_i^{(0)}\cos\kappa_i^{(0)}\left(X_j - X_{0,i}^{(0)}\right) +$$

$$+ \left(\sin\omega_i^{(0)}\sin\varphi_i^{(0)}\cos\kappa_i^{(0)} + \cos\omega_i^{(0)}\sin\kappa_i^{(0)}\right)\left(Y_j - Y_{0,i}^{(0)}\right) +$$

$$+ \left(\sin\omega_i^{(0)}\sin\kappa_i^{(0)} - \cos\omega_i^{(0)}\sin\varphi_i^{(0)}\cos\kappa_i^{(0)}\right)\left(Z_j - Z_{0,i}^{(0)}\right),$$

$$(f_y)_{ji}^{(0)} =$$

$$= \left\langle \begin{bmatrix} -\cos\varphi_i^{(0)}\sin\kappa_i^{(0)} \\ \cos\omega_i^{(0)}\cos\kappa_i^{(0)} - \sin\omega_i^{(0)}\sin\varphi_i^{(0)}\sin\kappa_i^{(0)} \\ \cos\omega_i^{(0)}\sin\varphi_i^{(0)}\sin\kappa_i^{(0)} + \sin\omega_i^{(0)}\cos\kappa_i^{(0)} \end{bmatrix} \cdot \begin{bmatrix} X_j - X_{0,i}^{(0)} \\ Y_j - Y_{0,i}^{(0)} \\ Z_j - Z_{0,i}^{(0)} \end{bmatrix} \right\rangle =$$

$$= -\cos\varphi_i^{(0)}\sin\kappa_i^{(0)}\left(X_j - X_{0,i}^{(0)}\right) +$$

$$+ \left(\cos\omega_i^{(0)}\cos\kappa_i^{(0)} - \sin\omega_i^{(0)}\sin\varphi_i^{(0)}\sin\kappa_i^{(0)}\right)\left(Y_j - Y_{0,i}^{(0)}\right) +$$

$$+ \left(\cos\omega_i^{(0)}\sin\varphi_i^{(0)}\sin\kappa_i^{(0)} + \sin\omega_i^{(0)}\cos\kappa_i^{(0)}\right)\left(Z_j - Z_{0,i}^{(0)}\right),$$

$$g_{ji}^{(0)} = \left\langle \begin{bmatrix} \sin\varphi_i^{(0)} \\ -\sin\omega_i^{(0)}\cos\varphi_i^{(0)} \\ \cos\omega_i^{(0)}\cos\varphi_i^{(0)} \end{bmatrix} \cdot \begin{bmatrix} X_j - X_{0i}^{(0)} \\ Y_j - Y_{0i}^{(0)} \\ Z_j - Z_{0i}^{(0)} \end{bmatrix} \right\rangle =$$

$$= \sin\varphi_i^{(0)}\left(X_j - X_{0i}^{(0)}\right) -$$

$$- \sin\omega_i^{(0)}\cos\varphi_i^{(0)}\left(Y_j - Y_{0i}^{(0)}\right) +$$

$$+ \cos\omega_i^{(0)}\cos\varphi_i^{(0)}\left(Z_j - Z_{0i}^{(0)}\right).$$

Then, the *linearised* observations ("delta-observations") are

$$\underline{\ell}_{ji} \stackrel{\text{def}}{=} \begin{bmatrix} \Delta\underline{x}_{ji} \\ \Delta\underline{y}_{ji} \end{bmatrix} = \begin{bmatrix} \underline{x}_{ji} - x_{ji}^{(0)} \\ \underline{y}_{ji} - y_{ji}^{(0)} \end{bmatrix},$$

the input for the adjustment computation.

9.3   Tie points

9.3.1   Measurement

Tie points may be any points that show in two or more images and are well defined. Often, roof corners of buildings make good tie points.

   If you are manually measuring tie points, follow these simple rules:

1. Measure a set of tie points that — together with the ground control points — are well spread out over the area to be mapped. It is useless to create tie points that are too close together or too close to a ground control point.

2. Only measure real points, uniquely defined in three dimensions. A "point" that should *not* be used as a tie point is shown in figure 12.5, an intersection of road lines on two different levels. Also, don't measure points that might move between exposures: cars, animals, shadows, trees waving in the wind.

9.3.2   Theory

If you introduce tie points, this will lead to the further introduction of three co-ordinate unknowns for every tie point. These will appear in the vector $\widehat{x}$ in equation 9.1, and corresponding elements will appear in the design matrix A. Note that these elements, for tie point k, are *now* obtained by partial differentiation with respect to the three tie-point co-ordinates (X_k, Y_k, Z_k), where k appears in the place of j in equations 9.2:

$$\underline{\ell}_{ki} + \underline{v}_{ki} = A_{ki}\widehat{x}_k,$$

with

$$\underline{\ell}_{ki} = \left[\begin{array}{c} \Delta \underline{x}_{ki} \\ \Delta \underline{y}_{ki} \end{array} \right],$$

the camera co-ordinates for tie point k in image i,

$$\widehat{x}_k = \left[\begin{array}{c} \widehat{X}_k \\ \widehat{Y}_k \\ \widehat{Z}_k \end{array} \right]$$

the (unknown) terrain co-ordinates of the tie point, and

$$A_{ki} = \left[\begin{array}{ccc} \dfrac{\partial x_{ki}}{\partial X_k} & \dfrac{\partial x_{ki}}{\partial Y_k} & \dfrac{\partial x_{ki}}{\partial Z_k} \\ \dfrac{\partial y_{ki}}{\partial X_k} & \dfrac{\partial y_{ki}}{\partial Y_k} & \dfrac{\partial y_{ki}}{\partial Z_k} \end{array} \right].$$

Left Image Points — 1997_016_300dpi.bmp

Id	Col
P01	2196.000
P02	2141.000
P09	1512.000
P04	1001.000
P03	805.000
P12	2499.000
P05	1142.000

All Registered Points — All Points

Id	Type	E
P01	Control	681100.2
P02	Control	681079.0
P03	Control	680114.6
P04	Control	680262.7
P08	Control	681875.8
P09	Control	680626.2
P10	Control	682031.4

Right Image Points — 1997_017_300dpi.bmp

Id	Col
P13	2678.000
P12	1523.000
P11	2643.000
P10	2420.000
P01	1206.000
P02	1156.000
P08	2238.000

Images Measurements — Point P01

Image File	Column	Line
1997_018_300dpi.bmp	167.000	2529.0
1997_017_300dpi.bmp	1206.000	2415.0
1997_016_300dpi.bmp	2196.000	2445.0

FIGURE 9.2. Phototriangulation main window.

For this, we need to partially differentiate

$$
x_{ki} = c \cdot
$$
$$
\cdot \frac{\cos\kappa_i\left(X_k - X_{0i}\right) + \sin\kappa_i\left(Y_k - Y_{0i}\right) + \left(\omega_i\sin\kappa_i - \varphi_i\cos\kappa_i\right)\left(Z_k - Z_{0i}\right)}{\varphi_i\cdot\left(X_k - X_{0i}\right) - \omega_i\cdot\left(Y_k - Y_{0i}\right) + \left(Z_k - Z_{0i}\right)},
$$
$$
y_{ki} = c \cdot
$$
$$
\cdot \frac{-\sin\kappa_i\left(X_k - X_{0i}\right) + \cos\kappa_i\left(Y_k - Y_{0i}\right) + \left(\varphi_i\sin\kappa_i + \omega_i\cos\kappa_i\right)\left(Z_k - Z_{0i}\right)}{\varphi_i\cdot\left(X_k - X_{0i}\right) - \omega_i\cdot\left(Y_k - Y_{0i}\right) + \left(Z_k - Z_{0i}\right)}.
$$

Every tie point thus produces *two rows* in the system of observation equations *for every image* in which it is visible, and *three elements* in the vector of unknowns irrespective of the number of images in which it is visible. This means that the design matrix A also gains two rows per image, and three columns.
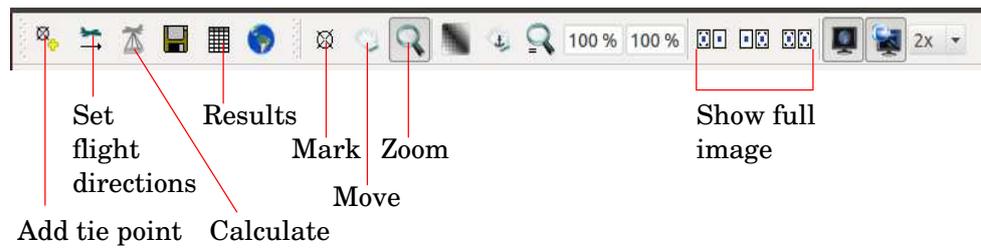
FIGURE 9.3. Aerotriangulation tool bar.

9.4   Aerotriangulation in e-foto

We must construct a common three-dimensional model for several images, in either a relative or an absolute co-ordinate frame. The model for doing this contains as unknown parameters the six location and orientation parameters $X_0, Y_0, Z_0, \omega, \varphi, \kappa$ of the camera in the terrain co-ordinate frame for every image used.

See figure 9.2 for the main window. Constructing the model requires

1. a sufficient number of known points or *ground control points*, the terrain co-ordinates of which were loaded from file in the Project Manager's Points tab, and for which camera co-ordinates may have been measured and saved in the Exterior Orientation module.

2. However, these points may or may not suffice for connecting each image strongly enough with the other images to allow a strong determination of the model parameters. Therefore, additionally we may need to determine *photogrammetric tie points*: points that can be seen in two or more images and thus "tie them together". These points should be evenly distributed throughout the overlap area of the pictures. These tie points will be numbered successively after the ground control points: PF24, PF25, etc. Unlike control points, tie points have no known co-ordinates; however, in the course of the aerotriangulation, terrain co-ordinates will be determined (estimated) for them, and these will be saved into the Project Manager module's Points list.

In order to do phototriangulation, one brings up the main screen by choosing Phototriangulation from the Execute main menu. There are many panels in the main window; only two images are displayed at a time, the Left Image and the Right Image, which can be selected separately.

The *user interface* is very similar to that of Exterior Orientation: also here, for some versions of e-foto, one first has to specify a "flight direction"
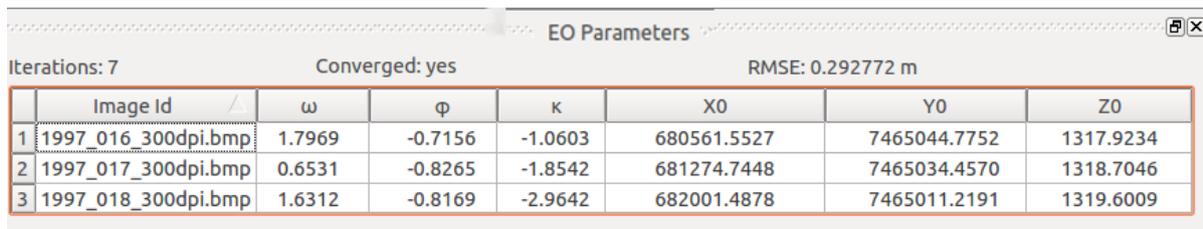
	Image Id	ω	φ	κ	X0	Y0	Z0
1	1997_016_300dpi.bmp	1.7969	-0.7156	-1.0603	680561.5527	7465044.7752	1317.9234
2	1997_017_300dpi.bmp	0.6531	-0.8265	-1.8542	681274.7448	7465034.4570	1318.7046
3	1997_018_300dpi.bmp	1.6312	-0.8169	-2.9642	682001.4878	7465011.2191	1319.6009

EO Parameters — Iterations: 7 — Converged: yes — RMSE: 0.292772 m

FIGURE 9.4. Exterior orientation elements obtained.

using a dial-like interface (figure 8.6), invoked by clicking on the aircraft-with-arrow toolbar icon. Use here the same directions you used for the individual photographs in Exterior Orientation.[1] Note that, as a block of images used for phototriangulation may consist of multiple strips in which the aircraft flew in opposite directions, you may have to enter different values for different images. Setting these directions correctly will make the phototrangulation adjustment computation converge rapidly. See figure 9.4, showing the exterior orientation elements computed after only eight iteration steps.

After setting the flight directions, the computation icon — showing an aircraft in the act of photographing the terrain — is no longer greyed out. Click on it.

The module offers a window (figure 9.5) where you can choose which images to take along in the computation, and which points, both ground control and tie points. You can move any or all of these between the left window — which lists what is available — and the right window — listing what is actually used — before proceeding with the calculation.

You will have to choose the number of iterations allowed, as well as the precision criteria applied to judge if the computation has converged, separately for camera location co-ordinates and for camera orientation angles. These usually do not have to be changed; only the number of iterations may be increased if convergence is slow.

It may happen that the module complains that the geometry is not strong enough, and that *tie points* need to be added; the toolbar offers buttons for doing this.

Technical remark: e-foto's Phototriangulation module will offer you the option of using a *local topocentric reference* for doing its computations in. While this seems like a good idea, this feature is — in

---

[1]And consider that, also here, the angle to be entered is κ_0, which may be either the flight direction or the direction opposite to it. . .
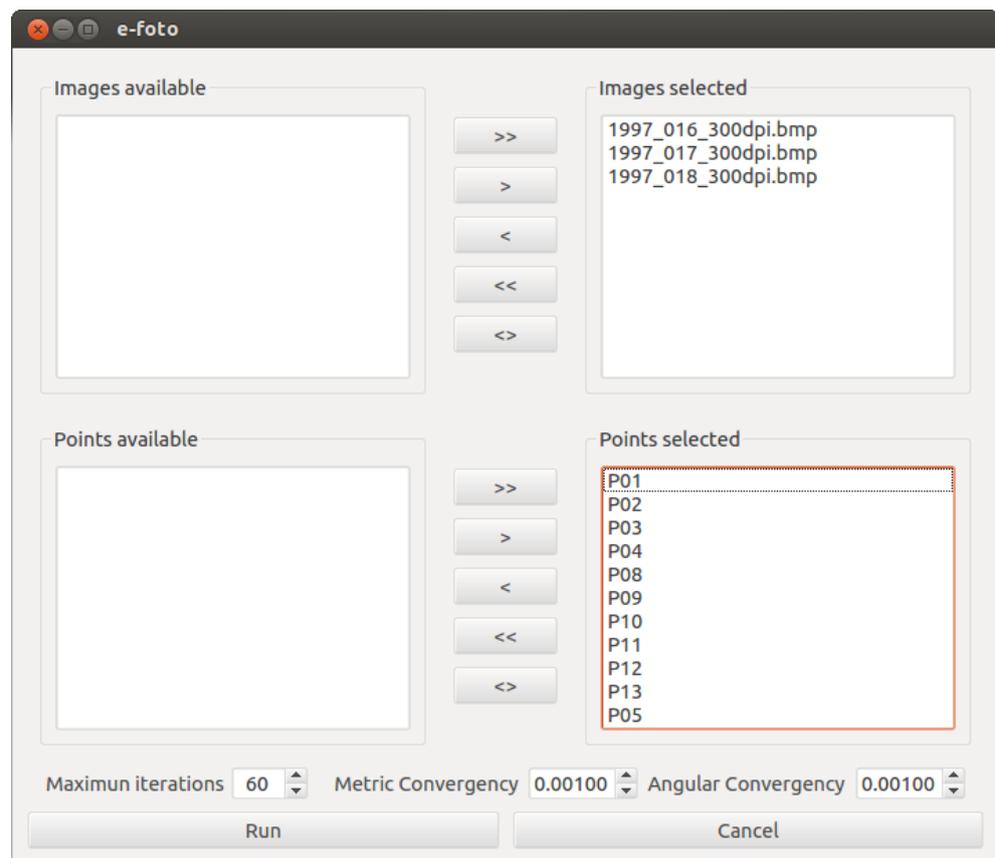
FIGURE 9.5. Inclusion / exclusion of images and points.

some e-foto versions — *buggy* and you should not use it: choose the "No" option. If accidentally you choose the "Yes" option, you will obtain nonsensical values (very large, or all zero values, huge RMSE, no convergence) for the exterior orientation elements. And somehow, e-foto will remember this nonsense: you have to *exit the programme completely* and start it again to be able to get reasonable results.

9.5   Debugging an aerotriangulation adjustment

The end result of a successful phototriangulation is high quality exterior orientation elements for all images in the block. One should especially verify that

1. the *height* of the aircraft Z_0 is realistic, and roughly the same for all images; in this example Brazilian case (figure 9.4) it is about 1318 m

2. the *camera orientation angles* ω, φ, κ (given here in degrees) are
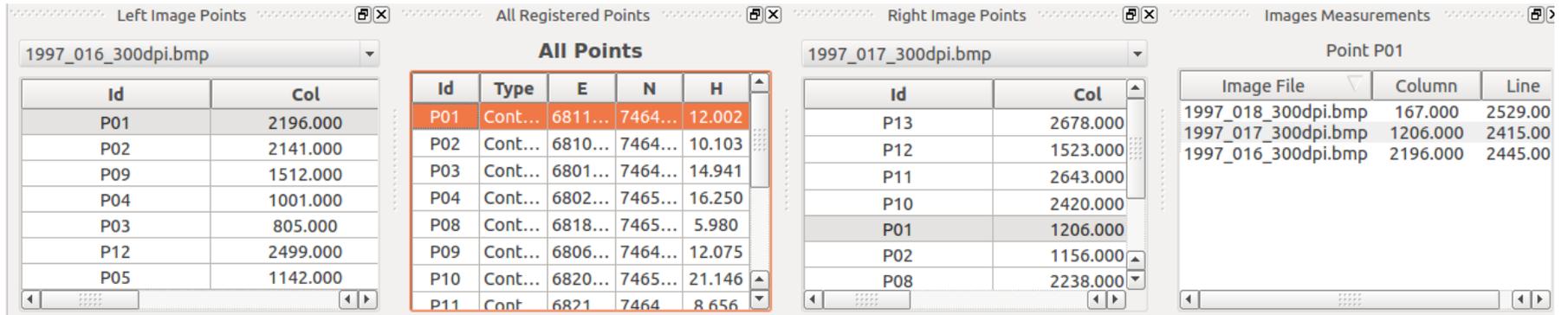
FIGURE 9.6. Aerotriangulation point list. Points in two images are listed, with in the middle the list of all points (control points and tie points).

realistic: φ and ω no more than a few degrees, and κ roughly the same for all images *in the same strip*. For adjacent strips, one expects the κ values to be opposite, i.e., $\kappa' = \kappa \pm 180°$

3. the computation has *converged*, and

4. the root-mean-square error (RMSE) is acceptable (in the example it is ± 0.29 m, which is a little large). Do give allowance though for a large block to produce a larger RMSE than the uncertainties of individual ground control points in individual images.

If you are performing aerotriangulation on a large block with many images, it is not likely that you will achieve a correct result on first try. There will be *errors* in the input data:

1. The ground control point co-ordinates in the terrain reference system of some of your ground control points may be wrong

2. The measurement of the image co-ordinates for these ground control points may be wrong

3. Wrongly measured tie points

4. Something wrong in the e-foto set-up.

The important thing now is: *don't panic*! Finding and eliminating the error requires a systematic approach. First of all, you should realise that the aerotriangulation computation does not need to be done on the whole set of images (which may be very large); you can at first do the computation on sub-sets of images, or even on individual photographs. The latter case corresponds to exterior orientation (chapter 8) of that photograph. It requires, for a robust result, the presence of at least four ground control points, of which no three are positioned on one line.

Generally the requirements for processing sub-sets of photographs in a way that will detect data errors are:

1. There should be four (or more) ground control points within the area covered by the images together. These points need *not* be in the overlap areas of two or more images. These should be spread reasonably around the area, with no three points on one line.

2. There should be four or more common points in the overlap area of each pair of images. These may be ground control points or tie points. If this condition is not met, you should be measuring more tie points in your image pairs.

It is clear from the above, that for successful debugging, it is good to

have *an abundance* of ground control points! So, include all points you can find that have their terrain co-ordinates measured, and if you are yourself responsible for measuring them, make a generous measurement plan. The time spent on ground measurements will pay itself back when processing the imagery.

1. Firstly, find the *individual images* that contain four or more ground control points. Perform exterior orientation (chapter 8). At least some of these images should give a correct result, as judged by the root-mean-square error (RMSE). If not, i.e., if *all* images give poor results, you must suspect that there is something wrong with the general set-up of e-foto (chapter 6); go through everything carefully again.

2. For those images that do not produce a good result, go through each of the ground control points (GCP), checking that

   (a) the given terrain co-ordinates in the GCP table are correct, and

   (b) the image co-ordinates measured in exterior orientation are correct, and expecially that the points were all *identified* correctly in the image.

3. If that doesn't help, you should *remove* each of the ground control points in turn from the computation, to see if this gives a better result. If you get a good result by removing a GCP, chances are that it is precisely that point which is in error. This technique is called "cross-validation". Of course you can only do this if you have a sufficient number of GCPs appearing on the image, i.e., five at least.[2]

4. Next, for those images for which the above treatment is inapplicable, find *pairs of images* that contain four or more control points, and a sufficient number of common points — measure additional tie points if you need to. Perform the *aerotriangulation computation* on those pairs. Again, your aim should be to get a correct result, i.e., a result with a small RMSE. If not, go through the individual ground points again, like described above, items 1-3.

5. Next, consider *triplets* or even larger sub-sets of images. Apply the

---

[2]Because, if you only have four GCPs, removing one of them, *any* one, will leave you with three points only, which will always allow exterior orientation to be performed without producing any contradictions, even if errors are present. . .

same treatment.

Finally, when all images have produced acceptable results, either individually, in pairs, in triplets, or in larger sub-sets, you can attempt to perform aerotriangulation on the whole block. The above describes a debugging technique not unlike that used for finding errors in computing code. Good luck!

Stereo restitution

10

Traditionally, aerial photographs have been acquired and processed in *pairs*, allowing the formation of scenes that can be viewed in stereo by the human eye. In the traditional photogrammetric approach, a pair of photographs is placed inside a stereo restitution instrument that allows them to be precisely oriented relative to each other, so that a stereoscopic image would appear to the human eye. Then, the human operator would be able by further mechanical manipulation to move a marker or cursor within both pictures, effectively moving it three-dimensionally within the viewed scene, and allowing it to be placed on chosen terrain details. The mechanics of the stereo restitution instrument would then allow three-dimensional co-ordinates to be recovered for plotting or map-making.

Today, this whole process is done digitally, using photographs in digital form. The formation of the 3-D scene is done computationally and presented on a computer screen. Stereo *vision* however, the sensation of depth and three-dimensionality, continues to require the human eye-brain combination, although also there, computing technology is making progress...

10.1 Relative orientation

For stereo restitution to work, one does not need to solve for the location and orientation of the two camera locations used in any absolute sense. One only needs to do a so-called *relative orientation*. This process, involving a sufficient number of common points in the terrain seen in both photographs, establishes *the vector connecting the two camera locations in space*, and the camera orientations at both ends, but in a co-ordinate frame that is *still arbitrary* and not connected to the terrain.

Note that the common points used in relative orientation do not need to

have known co-ordinates. It suffices that they are visible in both images, for example corners of roofs are well defined and typically used.

10.2 Absolute orientation

In absolute orientation we use the absolute terrain co-ordinates (X, Y, Z) of known points to obtain absolute camera locations and orientations in the same, 3-D, terrain co-ordinate frame. This is not necessary for successful stereo vision. It is however necessary for the successful measurement of 3-D co-ordinates in a stereo pair, obtaining co-ordinates in a known system.

Absolute orientation is very similar to exterior orientation for a single camera: but here, *seven orientation elements* are to be determined, three translations (shifts of the co-ordinate origin), three rotations, and *the length of the baseline* (distance between the two camera locations) in the model. And, contrary to exterior orientation, here *the complete model* with its two camera positions and the baseline between them, is translated and rotated in space. But just like in exterior orientation, also here, ground control points with known co-ordinates are used in the computation. But, as we have here three-dimensional model co-ordinates for these ground control points, the number of condition equations is actually $3p$ for p points. This means that using three points will give an overdetermination of 9/7, making the solution robust.[1]

10.3 Feature extraction

10.3.1 Parallax elimination

In order to successfully extract features from a pair of images using *stereo vision*, they need to be properly placed on the screen with respect to each other. First of all they should be oriented correctly: the vector between the two exposure centres of the image pair should have the same direction as the vector connecting the pupils of your eyes.

Then, especially the *parallax in the y direction* ("up-down") needs to be removed by *shifting* one of the images. See figure 10.1.

---

[1]One should still require these three points to not be on one line. More than three points would even be better, and the wider they are distributed throughout the model area, the better.
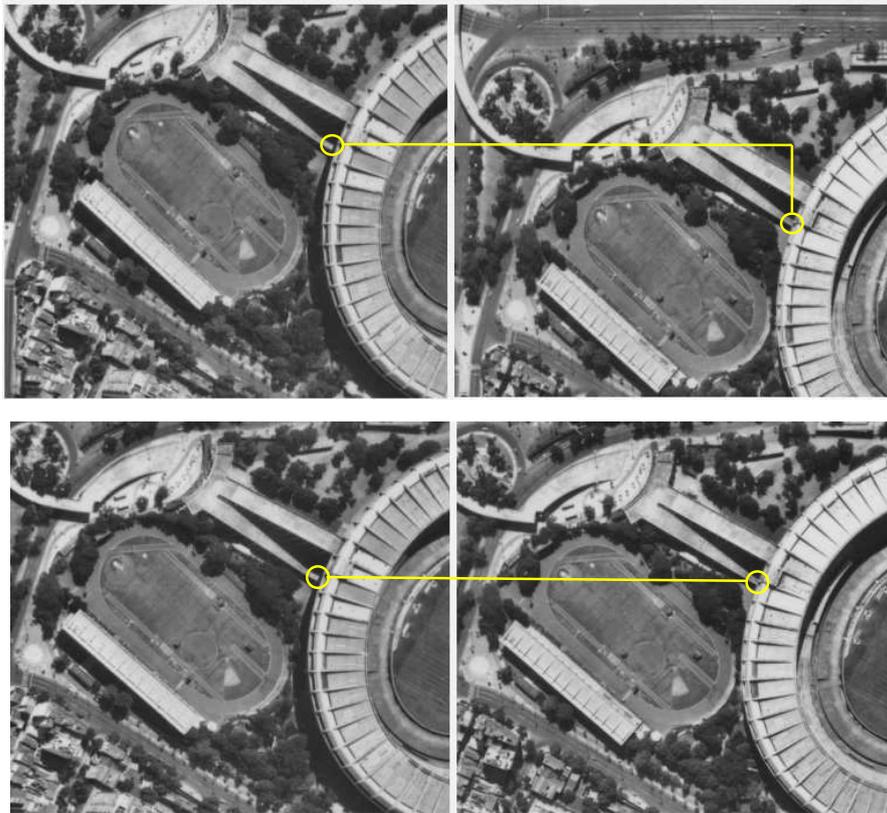
FIGURE 10.1. Eliminating the *y*-direction parallax. In the top figure, there is a large *y* parallax, and stereo vision is not possible. In the bottom figure, the right side image has been moved in the *y* direction to eliminate the parallax.

This makes the baseline between the two camera positions in the stereo model parallel with the line between your left and right eye pupils. Then, if everything else is in order, you will suddenly see the stereo image appear.

We can also describe this more formally: the *epipolar lines* in both images must be brought into superimposition and parallel to the eye base, see figure 10.2.

Also the *x*-direction parallax needs to be eliminated, but this is less critical as the human eyes are more flexible in converging. See figure 10.3.

After eliminating the parallax in both directions, you can not only view the image pair in stereo, but more importantly, you can view the *cursor* in stereo. This is called the *floating mark*. While, on a personal computer work station, the cursor can be moved left and right using the mouse, the vertical motion within the model is typically tied to the *mouse wheel*.[2] The [2]
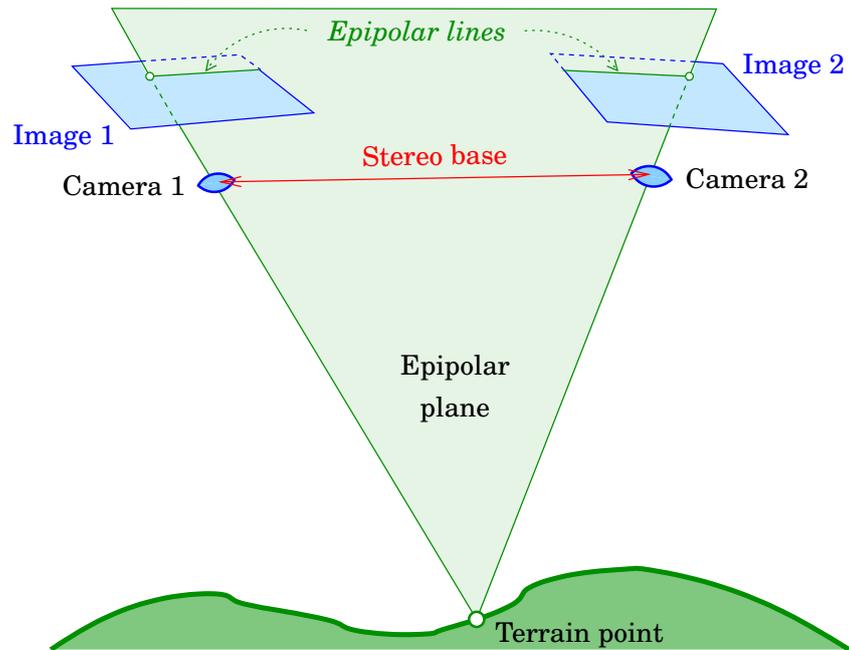
FIGURE 10.2. Epipolar plane and lines in a pair of aerial photographs. The epipolar plane contains the ground point, the two optical centres and the two image points of the ground point.

motion is implemented by shifting the cursor due right ($+x$ direction) in the right image, and by the same amount due left ($-x$ direction) in the left image. The stereo impression to the human eye is, that the mark moves vertically down, towards, and ultimately through, the terrain surface.
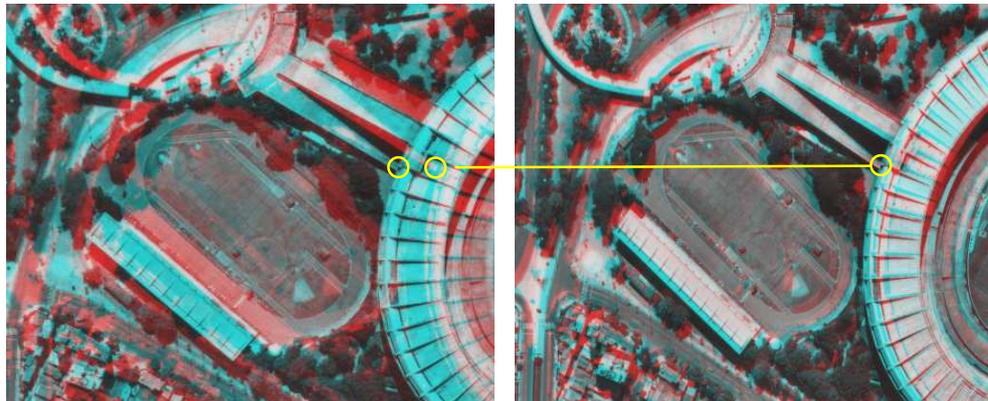
FIGURE 10.3. Eliminating the x-direction parallax. Left, before elimination; right, after elimination.

[2]In the old optomechanical photogrammetric workstations the two horizontal motions were controlled by the left and right handwheels, wheras the vertical motion was controlled by a footwheel. See figure 2.3 on page 11.

It is important that you become comfortable with the floating cursor, and move it up and down to land softly on the terrain, while it is also horizontally positioned on a chosen terrain feature. As you move around the image, you will notice that the parallax elimination may have to be repeated.

10.3.2   Features of interest

Now that we are set up to perform three-dimensional measurements in our pair of images, we can use this to *digitise* visible features for mapping. This is done by identifying the feature, recording its type and sub-type according to a *catalogue* of features to be mapped, and measuring the point or points making up the feature or its outline.

Typically features are categorised as *points*, *lines*, and *polygons* or areal features. Points are features without any interesting extent, like street lights. Lines typically are roads, canals, railroads, i.e., objects only having an interesting linear extent connecting two locations. Polygons again are all things with areal extent: buildings, parks, fields, ... the closed polygon being their outline.

Of course this classification may be dependent upon map scale: on a small-scale map an entire city may become a point feature.

Examples of areal feature properties:

- Land use and cover
- Vegetation or crop type
- Bedrock or soil type and texture, soil depth
- Landform type, slopes
- Drainage conditions, flood hazard
- etcetera.

10.4   Using the Stereoplotter module in e-foto

Note: before you can use the Stereoplotter module, you *must* have determined exterior orientation elements for *all* images of the project, either by using the Exterior Orientation module, or by using the Phototriangulation module.

The Stereoplotter module lets you extract sets of three-dimensional point co-ordinates for features visible in an image pair. Features may be points, lines, or polygons (i.e., surface areas, like buildings, land parcels etc.).
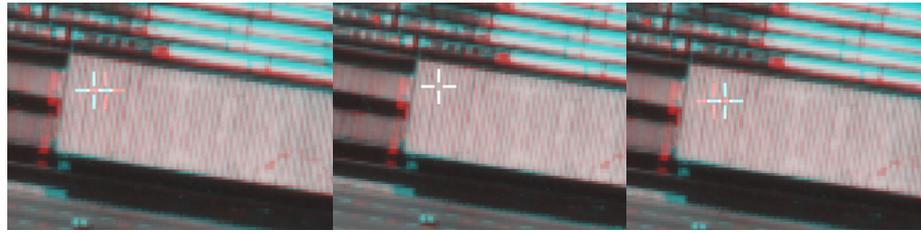
FIGURE 10.4. The floating mark above, at, and below the object surface.

In order to register a feature point, be it a point feature or a single point of a line or polygon feature, one must register camera co-ordinates in two images. From these, three-dimensional terrain co-ordinates may be computed if the camera's location and orientation parameters are known, for example by determination in the Exterior Orientation (Spatial Resection) module, or in the Phototriangulation module.

Like in expensive commercial stereoplotters, also here one may move a crosshair cursor in three dimensions through the terrain space, and place it on a feature *in three dimensions directly*. On old optomechanical stereo plotters, there were two handwheels for the horizontal X and Y motion, and a footwheel for Z. On our personal computer it is simpler: the mouse motion does X and Y and the mouse wheel does Z. Three-dimensional navigation requires however *stereo vision*, which again requires that the two images are placed on the screen approximately correctly with respect to each other.

Important: in the Stereoplotter module one needs to be able to move the left and the right image relative to each other, to remove any parallax that would interfere with stereo vision. This is done by holding down the mouse left button (Button 1) and at the same time either the SHIFT key or the CTRL key on the keyboard.

First of all, click the Move button in the toolbar (the little hand; *it will only work with this cursor*). After that

SHIFT-Mouse-1: drag the left image only

CTRL-Mouse-1: drag the right image only.

Before starting the work, you should do some basic settings, see figure 10.6:

Pair: the image pair you are working on

Stereo mode: choose here Anaglyph. These are the inexpensive red and cyan coloured "anaglyphic glasses" we use for stereo viewing (the
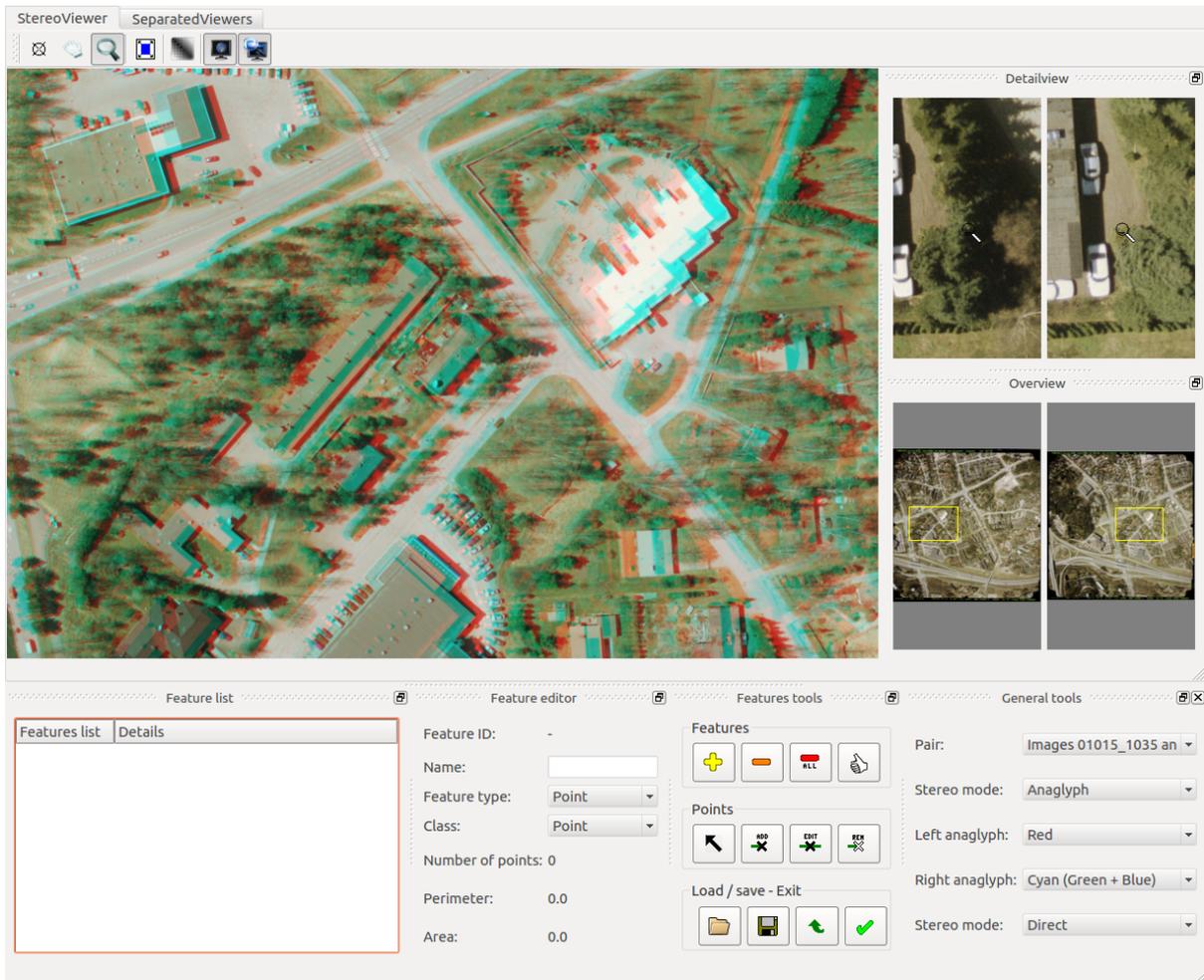
FIGURE 10.5. Stereoplotter main window.

alternative offered is polarised glasses; choose this only if your work station emits polarised images and you have suitable glasses for viewing).
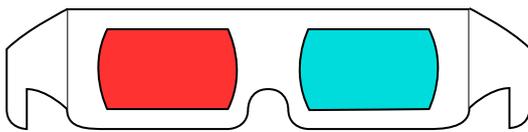
Do not touch the other settings. Direct ▷ Reverse may be useful if the images were loaded the wrong way round (or you folded your cardboard glasses the wrong way).

In the stereoplotter toolbar (figure 10.7) we have

- Zoom, which lets you place a little yellow rectangle around the area you want to zoom into: *mouse down* for the top left corner, *drag*, and *mouse up* again for the bottom right corner (using mouse button 1)

- Fit view, which will give you back a view on the full photographs
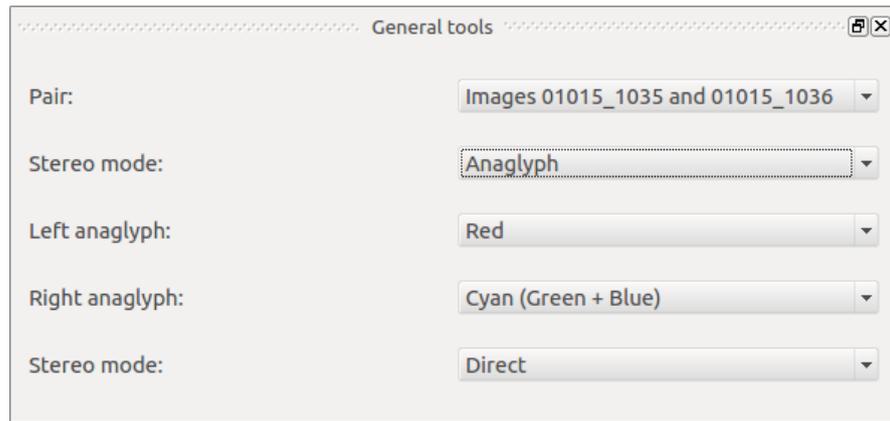
FIGURE 10.6. General tools.

- Move, which lets you move the images in two directions by clicking and dragging
- Mark lets you set a marker in either image. For accuracy, make sure that the image is enlarged enough (by using Zoom).

10.5 Editing features

Then, we have the machinery for entering features, a set of three panes depicted in figure 10.8. What you should do first, when entering a feature, is decide *what type* it is: *point*, linear (*line*), or covering an area (*polygon*). If it is a line or a polygon, you will be offered a list of alternative descriptors: if it is a line, is it a paved street, a railway, a river? Or if it is a polygon, is it a building, a square, a park, or a lagoon? And finally, you may give the feature a *name*. This is done in the middle pane; the data entered appears in the left pane.

The right pane is a block of buttons explained in Tableau 10.1.
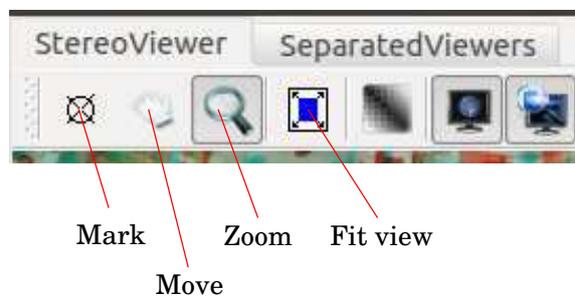
FIGURE 10.7. The stereo plotter toolbar.

TABLEAU 10.1. Editing panel for feature entry.

1. The standard workflow:

✚	Add new feature	Add new features such as point, line or polygon
⊕ ADD	Insert point mode	*Toggle*. If in the down position, indicates that new points will be added to the current feature
✔	Exit Stereoplotter	Exit the software

2. Correcting and modifying your work:

↖	Select feature / point	Select a feature / point to be edited ("current feature")
REM	Remove point	Remove the current point of the current feature. The feature and point to be removed must be selected in the *leftmost pane*.
EDIT	Edit point mode	*Toggle*. If in the down position, indicates that the current points will be modified
━	Remove feature	Remove the current feature
👆	End feature	Stop editing the current feature
━ ALL	Remove all features	Delete all features (empties the feature data base!)

3. Loading and saving work:

📂	Load features	Load previously measured features from file
💾	Save features	Save current work, i.e., all features currently in memory
↰	Export features as text	Export features in text format

10.6 Inserting a point

In e-foto, inserting a point in the corresponding locations in two images is tricky. You should proceed in the following fashion.

1. First select the feature that you want to register, and enlarge it sufficiently by zooming in (Zoom button on toolbar)
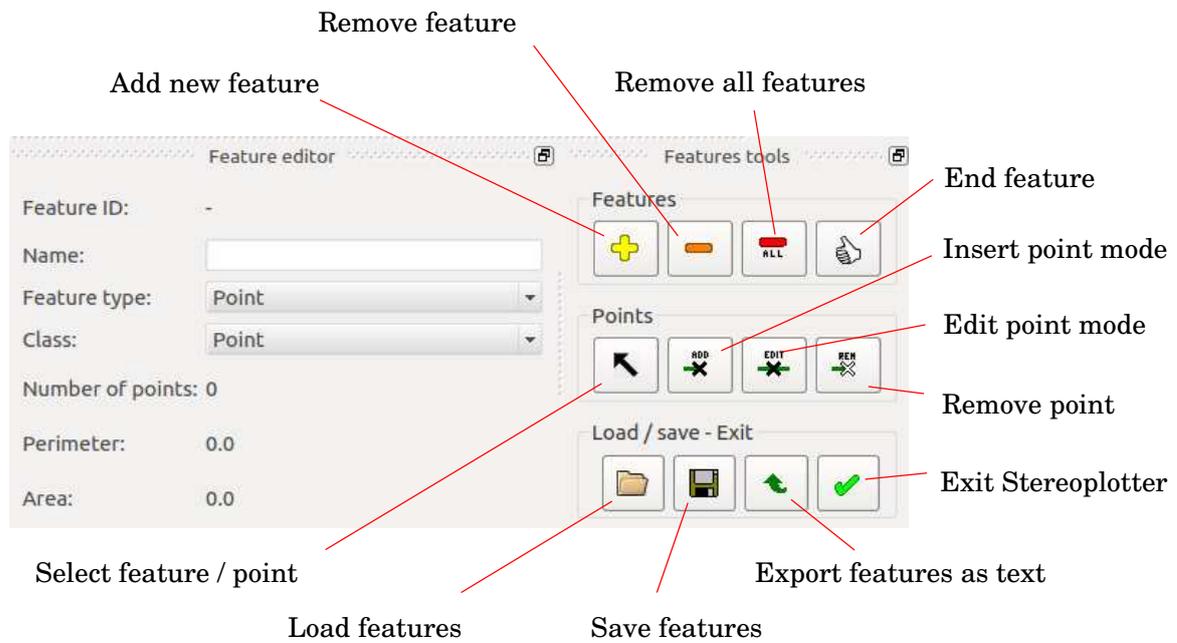
FIGURE 10.8. Panels used in entering feature data.

2. Check that the two images in the anaglyph superposition are placed correctly with respect to each other. Remove any parallax in the *y* direction completely, and in the *x* direction enough, so that you can comfortably see the stereo effect.

3. Select the feature's type and sub-type, and give it a name (if you don't, e-foto will just number your features of each type sequentially). It is a good idea to register all features of the same type together, so you have to enter this data only once. It is entered into the middle pane.

4. Activate the Mark button in the toolbar (the crosshair icon) and place the crosshair cursor on the first point of the feature — *in three dimensions*. You should see the crosshair "hanging" over the terrain, in mid-air. Move it sideways by moving the mouse, and *vertically by rolling your mouse's wheel*.

5. Now, click to register the point.

6. Note that the three-dimensional terrain co-ordinates ("object co-ordinates") (X, Y, Z) are displayed immediately under the anaglyph superposition.

After this, you can proceed to measure all the points of this feature, assuming that they are all on the same height. For the roof of a building this will be the case. For other objects this may *not be assumed*: for

example, a street will generally vary in height. Then, you have to use the mouse wheel to place the floating mark on every feature point before measuring it.

10.7   Working example

As an example we use the first pair of images from Rio de Janeiro. We select the railway station at Maracanã, and first of all, using the Move cursor (the little hand) and the CTRL key, we eliminate the y direction parallax and reduce the x parallax so that comfortable stereo viewing becomes possible:
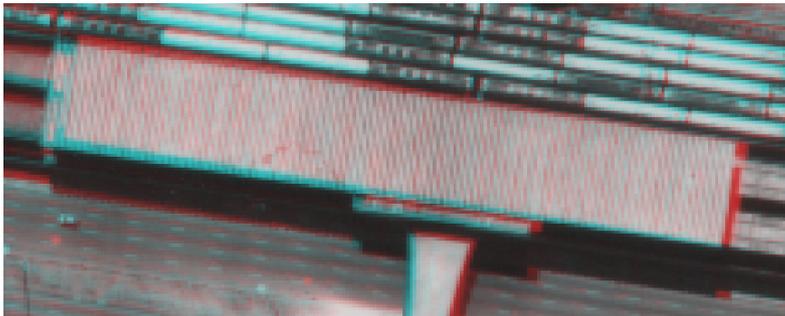
Next, we click on Add new feature and Insert point mode. Also, on the tool bar the crosshair ("Mark") button should be selected.

Position the crosshairs on the first corner of the station building using the mouse, move the mark vertically *using the mouse wheel* until it is at the same level as the station roof, and *click*.
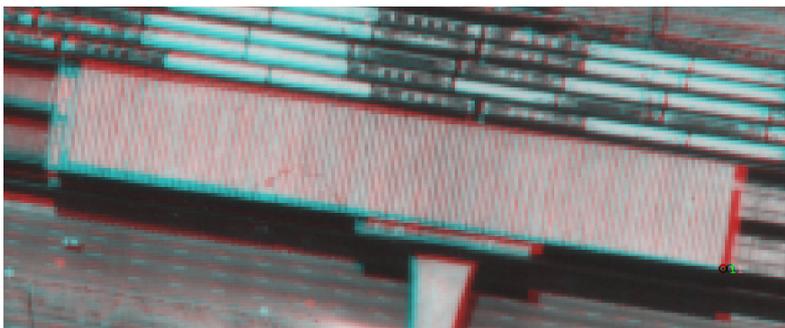
Repeat the procedure by moving the crosshair cursor to the second corner, adjusting height with the mouse wheel as necessary, and *click*.
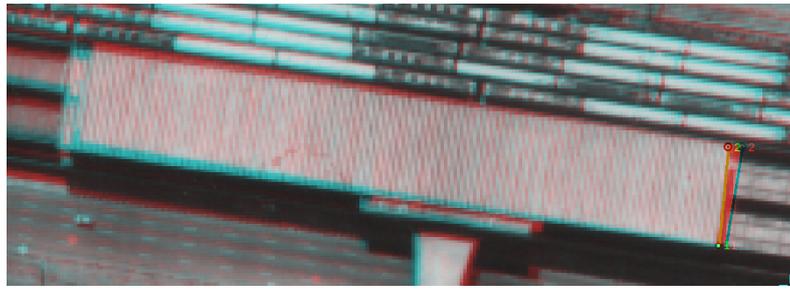
Same for the third point...
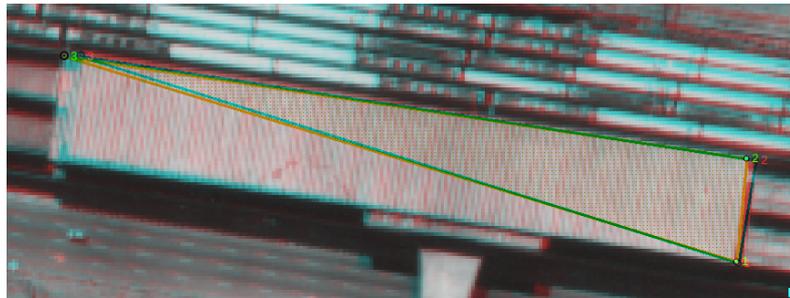
...and the fourth.
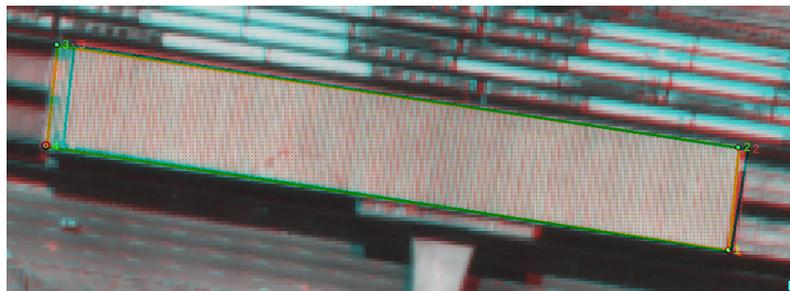
Terminate the feature by clicking the End feature button:

Now you should have the four point co-ordinate sets in the feature list:

FIGURE 10.9. A bigger sample of stereo plotter production showing several urban features in the Rio area.

Note that these are three-dimensional *terrain co-ordinates* in the reference frame defined by the ground control points used in Exterior Orientation or Phototriangulation.[3]

You may choose to save this data to a disk file for future reference using the Save features button. Alternatively, Export features as text creates a text file that can be loaded and edited in an ordinary text editor.

In order to *speed up the motion of the crosshair cursor*, which may be a bit slow on small machines, you may want to disable the Detailview pane in the user interface. Additionally you can remove the Overview pane and make the main window smaller.

Technical remark: many people find working with red-cyan anaglyphic glasses cumbersome, unpleasant, or imprecise. Some people do not even see stereo with them. Then, if you don't have access to a work station with special hardware (for example using polarisation), it

---

[3]You may verify this by measuring a ground control point as a point feature.

is a poorly guarded secret that with a little training, one can place the floating mark three-dimensionally even *without any glasses at all*.

One uses, instead of anaglyphic glasses, the "Detailview" window on the right hand side of the screen. It enlarges the same detail seen in the left and right images, and, after precisely eliminating the y parallax (by SHIFT-Mouse-1 or CTRL-Mouse-1 with the hand cursor), one uses the mouse wheel to position the twin cursors on the same point in both images. Consider, though, that the refresh of the Detailview window is a bit slow, so give it time.

The method can be speeded up by using the circumstance that the points of a rooftop outline are usually all on the same level. This means that one can use the hand cursor to travel from point to point without using the mouse wheel. This is how figure 10.9 was produced.

Measuring in aerial images

11

11.1   Image pyramids

It is typical in photogrammetric software that uses statistical analysis techniques on its image data, to construct an *image pyramid* first. This means that, in addition to the original image, an image is constructed at half the resolution, occupying one quarter of the disk space; then, one at one-quarter of the resolution; etc. See figure 11.1. One has to agree on a technique to derive the pixel values for every layer of the pyramid from the layer directly below it: simplest is just averaging 2×2 pixels. More sophisticated methods exist too.
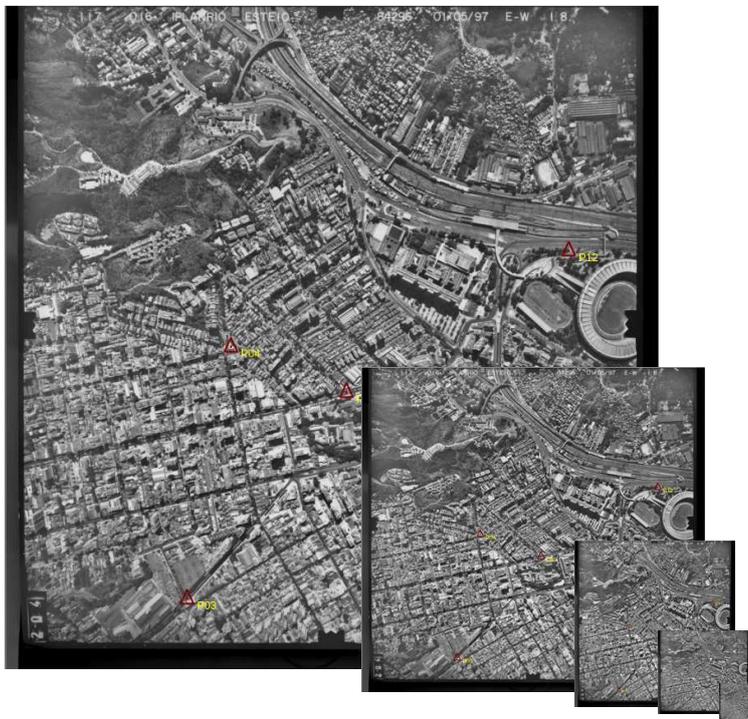
FIGURE 11.1. An image pyramid.

Note that e-foto does not use image pyramids. The widely used ERDAS LPS software however does.

11.2 Automatic feature matching by correlation

The statistical techniques that are typically used have to do with *automatic feature matching*. For example the automatic generation of tie points in image pairs for aerotriangulation is done this way: the *correlation* between two patches of the image is computed as[1]

$$\mathrm{Corr}\{g_1,g_2\}(\Delta x,\Delta y) = \frac{\mathrm{Cov}\{g_1,g_2\}}{\sqrt{\mathrm{Var}\{g_1\}\,\mathrm{Var}\{g_2\}}}, \tag{11.1}$$

with

$$\mathrm{Cov}\{g_1,g_2\} \stackrel{\mathrm{def}}{=} \int_{-d}^{d}\int_{-d}^{d} \left(g_1(x,y)-\overline{g}_1\right)\left(g_2(x+\Delta x,y+\Delta y)-\overline{g}_2\right)dx\,dy,$$

$$\mathrm{Var}\{g_1\} \stackrel{\mathrm{def}}{=} \int_{-d}^{d}\int_{-d}^{d} \left(g_1(x,y)-\overline{g}_1\right)^2 dx\,dy,$$

$$\mathrm{Var}\{g_2\} \stackrel{\mathrm{def}}{=} \int_{-d}^{d}\int_{-d}^{d} \left(g_2(x+\Delta x,y+\Delta y)-\overline{g}_2\right)^2 dx\,dy,$$

in which $g_1(x,y)$ and $g_2(x,y)$ are the two images, $\overline{g}_1$ and $\overline{g}_2$ their *averages* over the whole patch, and $\begin{bmatrix} \Delta x & \Delta y \end{bmatrix}^{\mathsf{T}}$ is the shift vector between the two images needed to maximise this correlation value. The value is always in the interval $\begin{bmatrix} -1,1 \end{bmatrix}$. What is typically seen is that, for most values of the shift vector, the correlation is fairly close to zero, but for one value of the shift vector, it becomes close to one. This means we have achieved a *match*.

The patch over which the correlation is computed is only $2d \times 2d$ pixels in size, but still, for a typical aerial image pair, this is a quite massive operation, due to the large number of alternative shift vectors $\begin{bmatrix} \Delta x & \Delta y \end{bmatrix}^{\mathsf{T}}$ that need to be inspected. One could limit the work by keeping d small, but then you will see many "false positives", high correlation values that nevertheless do not mean a match has been found — and consider the case where the image contains repeating features: this would easily fool the algorithm. But then, it would fool the human eye too.

---

[1]Folks who have studied statistics will notice the similarity with the statistical definition of correlation: here, the place of the *expectancy operator* is taken by the double integral over the patch area. The overbar notation used for average is also just shorthand for that.

The pyramid approach helps here: perform the correlation calculation first on the smallest image at the top of the pyramid, to find an approximate vector $\begin{bmatrix} \Delta x_1 & \Delta y_1 \end{bmatrix}^\mathsf{T}$. Then, use this to *limit the search* in the next lowest layer of the pyramid to the small square

$$
\begin{bmatrix} \Delta x_2 \\ \Delta y_2 \end{bmatrix} = \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \end{bmatrix} + \begin{bmatrix} i \\ j \end{bmatrix}, \qquad i, j = -1, \ldots, 1.
$$

In this way, the computational work is limited, even for a generous patch semi-size d, and of order $9\log 2d$ rather than $(2d)^4$.

11.3   Least-squares matching

The correlation method can be refined by a method called least-squares matching, where we allow, besides differences in *brightness* and *contrast* between the patches (perhaps due to changed lighting conditions between exposures), also a geometric distortion in the form of an *affine transformation*, which can account for the differences in deformations in the images caused by terrain patch slopes.

The free parameters describing all these differences are optimised in the least-squares sense. The model connecting the pixel brightness patterns to be correlated g_1 and g_2 in patches of the two images is not, as in section 11.2, $g_1(x,y)$ and $g_2(x + \Delta x, y + \Delta y)$, but the more complicated (Silveira et al., 2011)

$$
g_1(x, y) \qquad \text{and} \qquad \alpha g_2(ax + by + \Delta x, cx + dy + \Delta y) + \beta. \qquad (11.2)
$$

Here $(\Delta x, \Delta y)$ are the co-ordinate offsets between the centres of the patches, α describes the difference in contrast, and β the difference in brightness. The parameters a, b, c, and d together describe the geometrical affine transformation.

alpha αA
beta βB

Once homologuous point pairs have been identified, one can search for nearby points by recursively applying the same procedure, stepping in four different directions. The range of $(\Delta x, \Delta y)$ values to look at will be limited if the terrain is smooth. In this way, the whole area can be matched.

11.4   Scale Invariant Feature Transform (SIFT)

The SIFT method, Scale Invariant Feature Transform, was originally designed by Lowe (1999). The method is too complex to explain here in
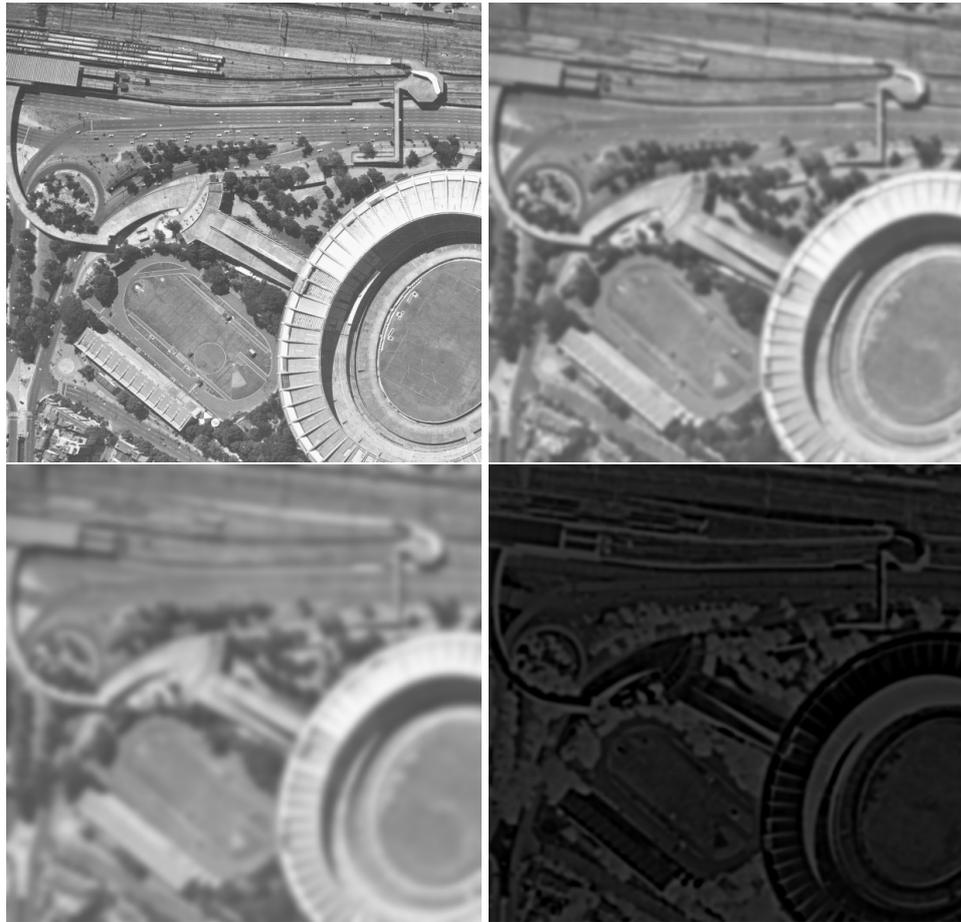
FIGURE 11.2. Difference of Gaussians. Original image, Gaussian blur width 8, Gaussian blur width 16, and their difference, presenting detail only between these two resolution levels.

Adding together the difference images (bottom right) thus obtained for all widths $k = 1$, 2, 4, …, 2^n will give us back the original image.

detail, but involves the building of a *pyramid* of images, produced from the original image by applying a *Gaussian blur* of widths in increasing powers of 2, and, subtracting in every step, the similarly constructed Gaussian blur image for the one lower power of 2 ("difference of Gaussians", see figure 11.2). This neatly separates the details on the various spatial scales, defined by a binary "scale number" k, into layers of a pyramid.

In this pyramid, there are three co-ordinates: the x and y pixel co-ordinates of the original image, and the "blurring scale" k. A point is declared a *point of interest* if this "difference-of-Gaussians" value assumes a local minimum in this three-dimensional space.

An alternative variant of SIFT, using wavelets instead of differences of

Gaussians, is also popular. Wavelets are well suited for this hierarchical decomposition by scale in powers of 2.

11.5 Automatic feature recognition

Another application of pyramids is *automatic feature recognition*. This is more challenging and requires a library of known and recognizable features. This is used, for example, in automatic construction of digital elevation models (DEMs) from image pairs, by recognising edge-like features — terrain ridges — which must at least be included in any DEM.

Digital Elevation Models

12

12.1 Infrastructure applications

Height information, of good quality, geographical distribution and spatial resolution, is a resource that is essential for the development of a country, more precisely, the construction of its infrastructure. The reason for this is that fluids — for example water — flow under the influence of gravity, and the direction in which they flow, and the energy that is released when they flow, depend on the *height differences* in the terrain. This applies to drinking or irrigation water, naturally flowing water from rainfall, in rivers and in the soil, and to sewage. It even applies to traffic, requiring roads and railroads to be built with only moderate changes in height along their trajectories.

There are a number of techniques for establishing a height system, or *vertical datum*, in a country. More traditional techniques include precise levelling, which however is laborious. More rapid, modern techniques use satellite positioning; however, in order to produce the proper kind of heights related to the gravity field, that is, *orthometric* or *normal heights* "above sea level", one needs a physical model of the figure of sea level, that is a *geoid model*. We shall discuss this problem later on.

The detailed description or modelling of heights in a smaller area often takes the form of digital elevation models, or DEMs. These DEMs are needed in connection with large infrastructure projects: roads, railroads, bridges, reservoir dams, irrigation works, mobile telephony link towers, etc. etc.

DEMs are typically *raster files* describing a larger or smaller area. An alternative format is *triangulation*, for example Delaunay[1] triangulation: points are given connected by edges that form triangles approximating

---

[1]Boris Delaunay (1890–1980) was a Russian mathematician, mostly remembered for
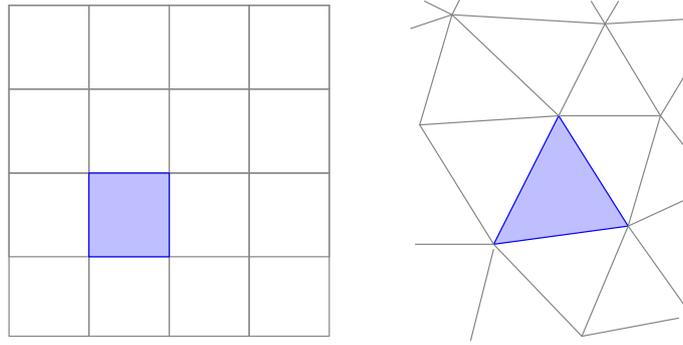
FIGURE 12.1. Digital elevation models as rasters (left) or triangulation models.

the shape of the terrain. This is often called a TIN (triangulated irregular network) model. What is nice about triangulation is that it can describe optimally terrains containing both smooth and rugged areas, as the inter-point spacing can be varied from place to place.

12.2   Presentation of map materials

DEMs can also be used to enhance the presentation of map materials. An obvious way is to add *height contours* to a map: this allows the user to read off the height of any point, as well as judge the convenience of routes, etc. Also, in physical geography maps, it is common to add *height colours* to maps between the contours, making the heights of the terrain more readily visible to the untrained eye.

A third technique often used in visualisation is *relief shading*, using an artificial light source, created by software, that shines on the terrain
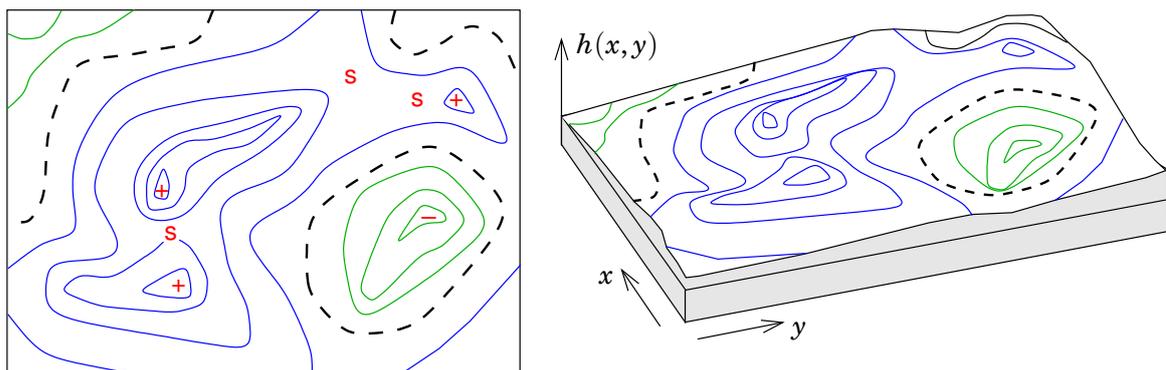
FIGURE 12.2. Height contours (left) and perspective view.

inventing a technique for optimally connecting a network of points on a surface into a triangulation of the surface. He was also an accomplished mountaineer.
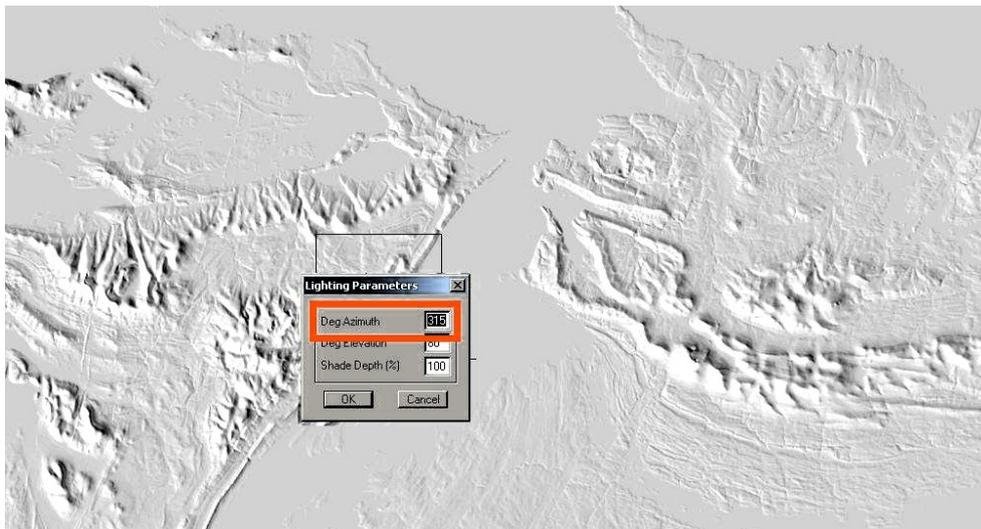
FIGURE 12.3. Example of a shaded relief representation of a digital elevation model. The light source is here placed in the North-West at a 60° elevation angle, as is conventional in cartography. © Eric Gaba @ Wikimedia Commons, (Gaba, 2007), detail.

from some oblique direction, making the relief visible — see, e.g., (Relief Shading). The merit of this approach is, that is visualises not just the heights but also the *slopes* of the terrain.

Of course the most obvious technique for representing terrain heights is as a *perspective image*, showing it as a bird's eye view, again from some oblique direction.

Both the shading technique and the perspective technique exploit deep properties of the human vision system.

12.3 Extraction from imagery

Extracting a digital elevation model from a set of exteriorly oriented aerial images can be done manually or more or less automatically. The manual process is laborious and no longer used. It was based on following, in a stereo restitution instrument, the height contours of the landscape using the floating mark at a constant height. The contours were registered digitally, for example by printing out a paper tape or punched cards. Together, the contours formed an elevation model of the terrain.

Today's DEMs are almost always in *raster format,* like the aerial images from which they are derived. Constructing the raster requires going through a pair of images and finding corresponding locations, so-called *homologuous points*. This is a challenging task to do automatically or

semi-automatically. One proceeds in the following *steps*:

1. One starts by providing a number of *seed points*, points for which three-dimensional terrain co-ordinates (X, Y, Z) are known. These could be, e.g., the ground control points of the aerial photography block.

2. The software then undertakes a *search process* outward from the seed points, seeking for neighbouring little patches of terrain visible in both images. Correspondence between patches can be established for example by *correlating* the patches. Then the terrain height is found for which this correlation is maximal. The new point added to the data base is then the centre of this patch. This can be done with good computational efficiency because the height is expected not to change much from that of the previous point. Nevertheless the number of numerical operations involved is huge, due to the large number of pixels in each image.

3. In order to cut down on the calculation effort, one may restrict the calculation to a sparser grid if it is known that the terrain is reasonably smooth.

4. After a grid of points with known heights has been formed throughout the area, an *interpolation* is undertaken creating a smooth model surface with a height value for every pixel in the DEM.

12.4 Automatic feature matching

In the literature are a number of approaches for establishing the correspondence of points or *patches* in overlapping image pairs, recently an object of scientific interest due to the development of *machine vision*. The following techniques can be distinguished:

1. SIFT, or Scale Invariant Feature Transform

2. Methods based on correlating patches

3. Methods based on topological relations.

Of these, the method based on correlating patches, which was discussed in section 11.2, can be refined into least-squares matching, as described in section 11.3. Also SIFT is discussed there, section 11.4.
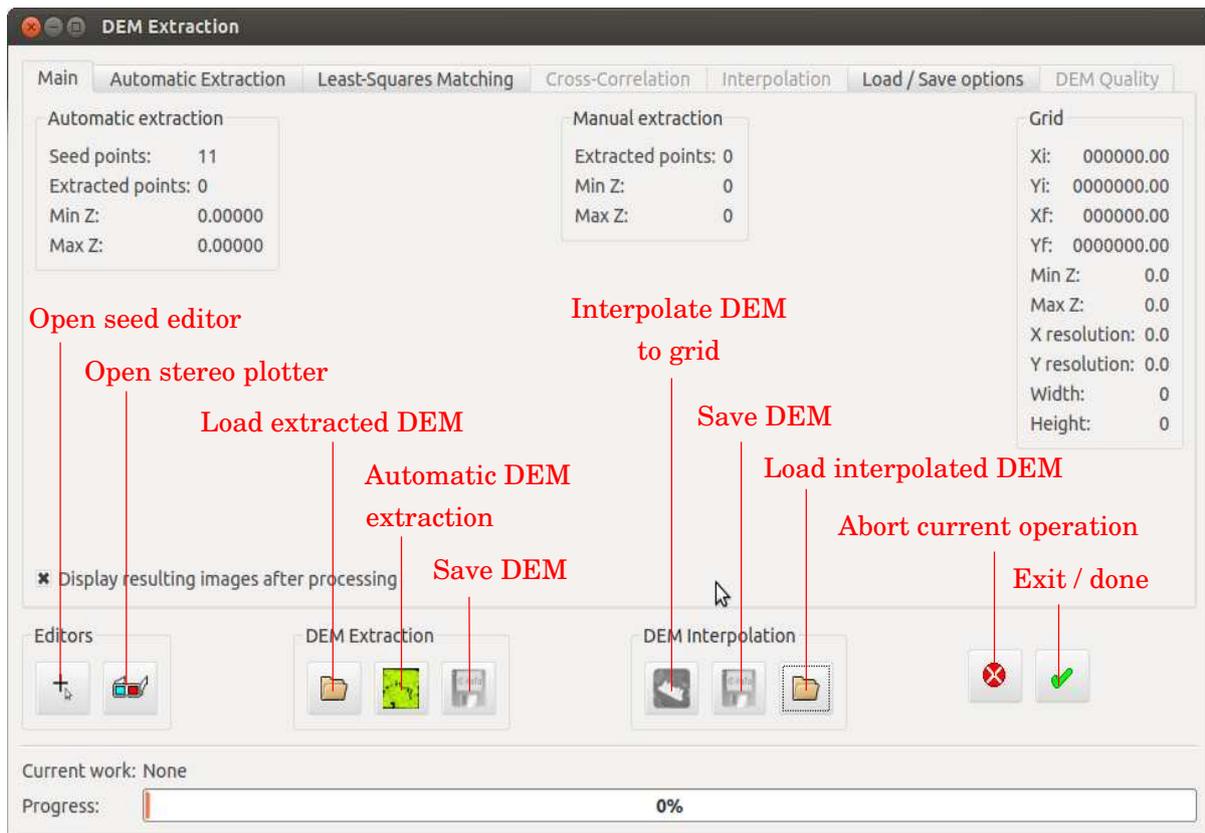
FIGURE 12.4. Main menu of DEM Extraction module.

12.5 Digital elevation models (DEM) with e-foto

The purpose of the DEM Extraction module is to construct a digital terrain model (DTM) out of a set of aerial photographs *that have been exteriorly oriented* — either individually in the Exterior Orientation module, or together in Phototrangulation. The DEM is obtained in the terrain co-ordinate system (X, Y, Z) in which the exterior orientation was obtained, that is the reference system of the given ground control points. It is thus a requirement that the images are interiorly and exteriorly oriented, for example by the Phototriangulation module.

The way the DEM is being constructed is by starting from a set of given *seed points*, see Silveira (2011). The technique used is *differential* in nature.

The DEM generation consists of two parts:

1. Obtaining a set of homologuous points for each stereo pair in the project, and their terrain co-ordinates (X, Y, Z) — "DEM Extraction"

2. Interpolate these points to a regular grid — "DEM Interpolation".
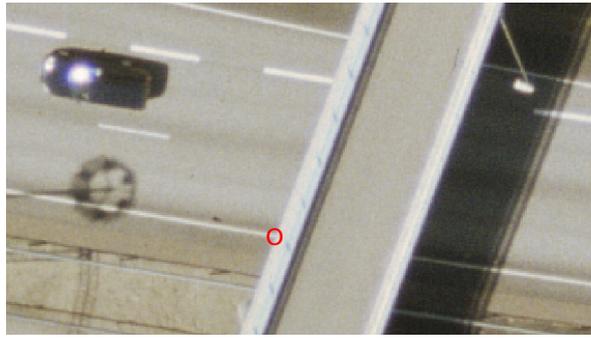
FIGURE 12.5. Quasi-point at a non-level road intersection. This is not a unique three-dimensional point.

12.6   Doing the DEM generation

12.6.1   Seed points

The "seed editor" that comes up when clicking on the button is shown in figure 12.6. E-foto generates a small number of "seed points" automatically, actually the ground control points. The user interface should be rather familiar by now. The idea is to place a number of *seeds*, homologuous points in each stereo pair, from which the software moves gradually outward, in a search process of areal expansion finding further homologuous point pairs with their three-dimensional co-ordinates to be used for building the DEM. If the seed points generated automatically are not sufficient, one can manually add some more.

Good seed points meet the following requirements:

1. they are spread evenly throughout the area of overlap of the image pair. Place new seed points especially there, where there are large *empty areas* between the already existing seed points

2. they are *sharply defined* in the terrain. E.g., the corners of buildings or other structures, *but at ground level*, or corners of white road markings like pedestrian crossings.

3. Make sure the *surrounding terrain is smooth*, so the areal expansion process will work. Don't use points on the roofs of buildings;[2] they won't allow areal expansion into the surrounding terrain.

4. Don't measure "quasi-points", like in figure 12.5, which are not really unique three-dimensional points, as may occur at non-level

---

[2]One can however load the information from feature extraction (the Stereoplotter module) into DEM Extraction.
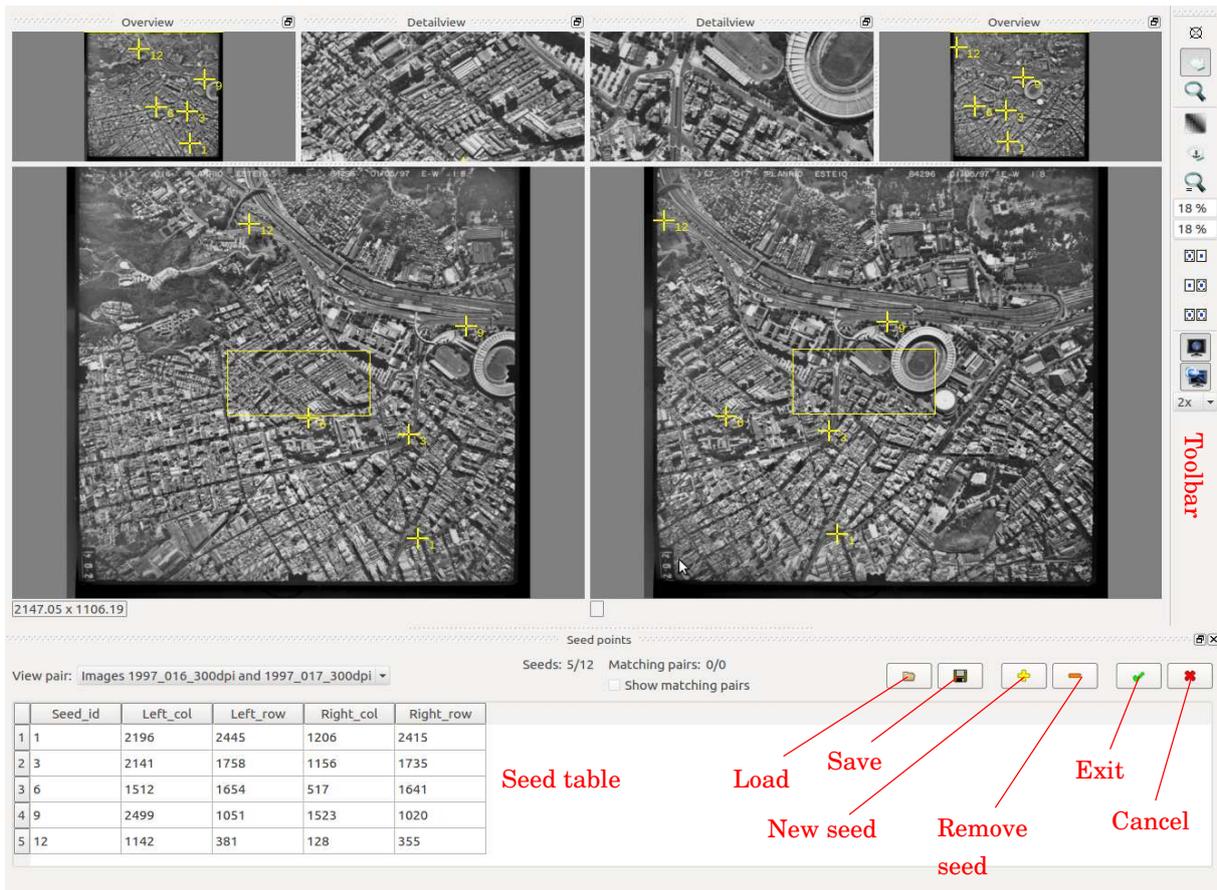
FIGURE 12.6. Seed editor main window. The seed points are in yellow, and are listed in the table bottom left. Seed points can be added using the buttons bottom right.

road crossings. Also, don't measure anything that *moves between exposures*, like cars, animals, trees in the wind, or even the shadows of lamp-posts!

An important feature not shown in the figure is the tick box in the middle marked "Show matching pairs". If you tick this box, the result of the previous DEM extraction / areal expansion run — i.e., the homologuous points already found — will be displayed as red crosses; see figure 12.7. This allows you to judge in which areas further seed points are needed to make a more complete coverage possible.

12.6.2 Areal expansion ("DEM Extraction")

Areal expansion of the set of homologuous points proceeds from existing points, by testing the four neighbouring points in the North, South, East and West directions at a distance given by the "Region Growing step", see figure 12.8. It compares the pattern found in a patch of the

	Seed_id	Left_col	Left_row	Right_col	Right_row
6	6	1325.37	10291.2	5435.49	10312.9
7	7	1218.66	1999.04	5462.77	1999.05
8	8	3804.35	7130.66	7992.41	7107.06
9	9	988.91	10966.6	5097.79	10997.4
10	10	6917.54	3491.04	11195.4	3458.79
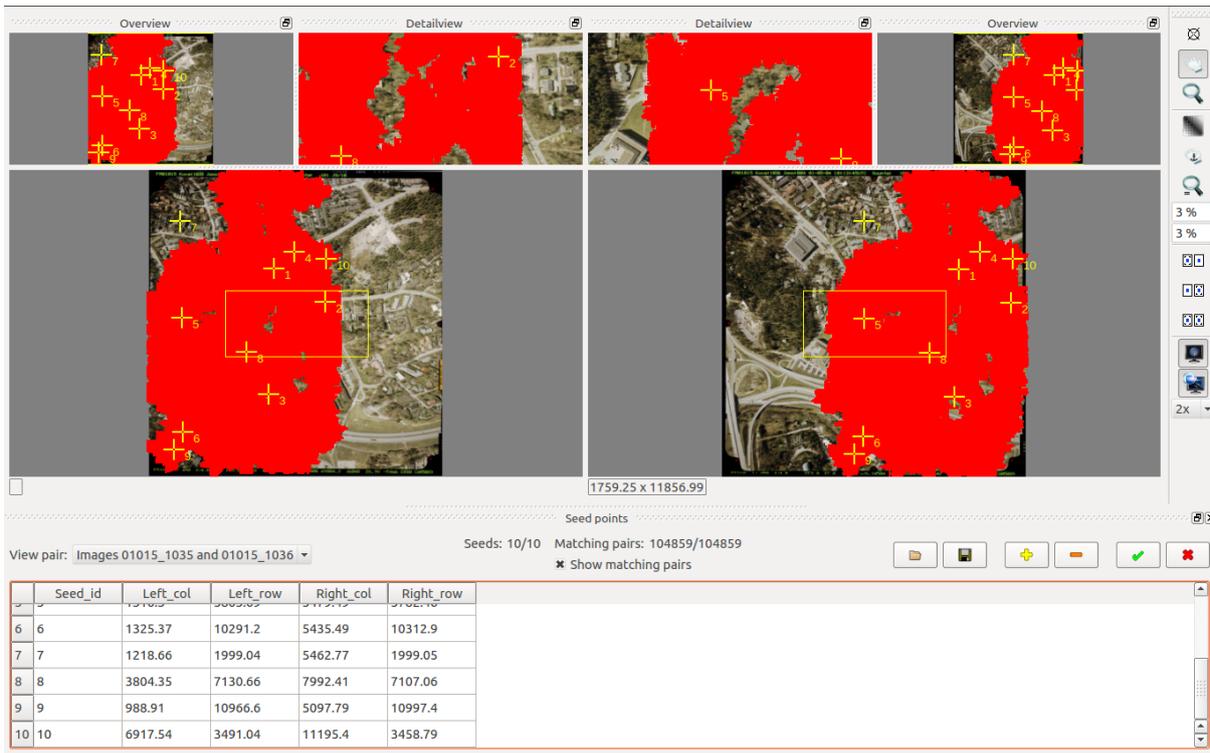
FIGURE 12.7. Seed editor with "Show matching pairs" activated, so the red areas show the coverage already obtained using Areal Expansion from the given set of seed points. Several seed points have been added to improve coverage.

image (a *template*) around the point with the corresponding pattern in a patch of the other image, a process called *cross-correlation*, equation 11.1. A variant of the technique, equation 11.2, allows for arbitrary geometric deformations between the two patches in the form of an affine transformation: this method is known as *least-squares matching*. In both techniques, the free parameters are estimated by the least-squares technique, minimising the sum of squares of differences and at the same time maximising the *Pearson*[3] *correlation* between the patches.

This process of gradual areal expansion can of course only proceed in areas that are visible from both images of a pair. It is clear that in our test area in the city of Rio de Janeiro (and in urban areas in general) this is a problem: high buildings frequently occlude the ground in one of the images, and these ground points cannot be reached by the algorithm. Adding more seeds helps a little, but is not a complete solution to this. Also discontinuous jumps in terrain height will block progression of the

---

[3]Karl Pearson (1857–1936) was an influential English statistician.
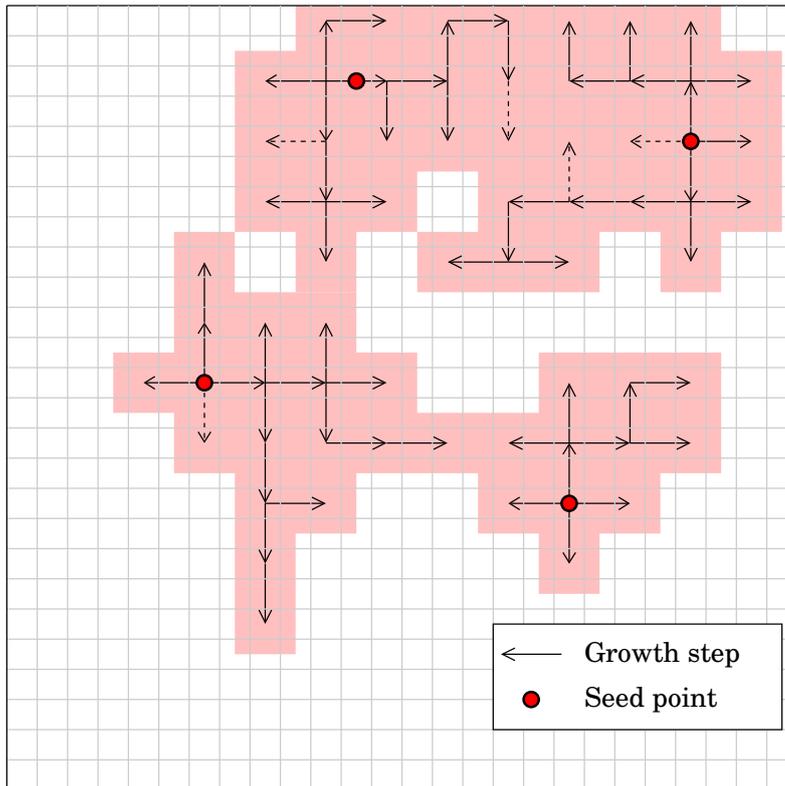
FIGURE 12.8. The areal expansion process. In this image, the growing step is two pixels, so a sparse grid of determined heights is constructed that still needs to be interpolated, see section 12.7. The pink background shows obtained coverage, like the red in figure 12.7.

areal expansion.

The further menus of DEM Extraction are depicted below. Each contains default values for the parameters with which the algorithm is known to function generally well based on experience. Of course more adventurous users can experiment with these values.

12.6.3 Extraction sub-menus

The Automatic Extraction sub-menu is shown in figure 12.9.

In this, Region Growing step is the number of pixels at a time the areal expansion proceeds in the NSEW directions. Thus, a *sparse* but regular grid of points is constructed that will need *interpolation* — the second step, discussed below — for full coverage by the DEM generated. This value may be increased to as much as 10 in smooth terrain, to speed up the process.

Also for smooth terrain, one may *downsample* the images to speed
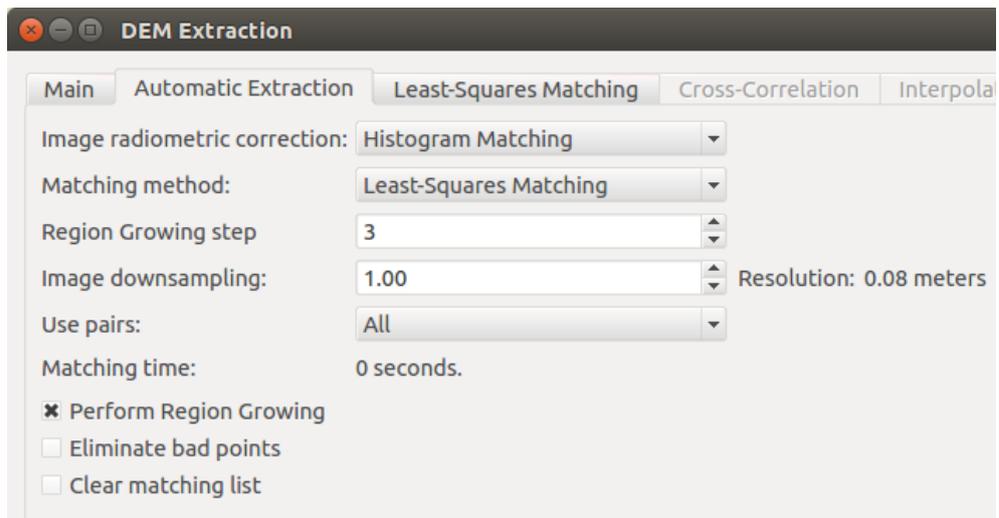
FIGURE 12.9. The Automatic Extraction sub-menu.

up the process. However, this will also reduce the resolution of the orthophoto image produced, so should be used with care.

There are two different approaches to automatic extraction offered: *least-squares matching*, and the simpler *cross-correlation*. Each has its own sub-menu, Least-Squares Matching in figure 12.10 and Cross-Correlation in figure 12.11.

The result obtained, after a prolonged run, looks like figure 12.12, a graph showing in yellow the homologuous points that were found by the process, and in black, locations not reached.

The statistics for matching accuracy (Pearson correlation) are given as a histogram, figure 12.13.

As you see, coverage is far from complete. Some of the large, black areas may be filled in by adding "seed points" in the seed point editor; you should place these seed points in the middle of fairly level areas. However, some areas, like high-rise, densely built urban areas, are just impossible to process. Here, the option of importing digitised features from the feature editor is useful.

12.7   Doing DEM Interpolation

DEM Extraction will produce a point set throughout the model that will still be sparse: if you magnify out a small area, you will see something looking like a sieve, with lots of little black holes. Therefore after DEM Extraction, one should run DEM Interpolation, which fills in the missing pixels in-between the pixels for which extraction has produced heights.
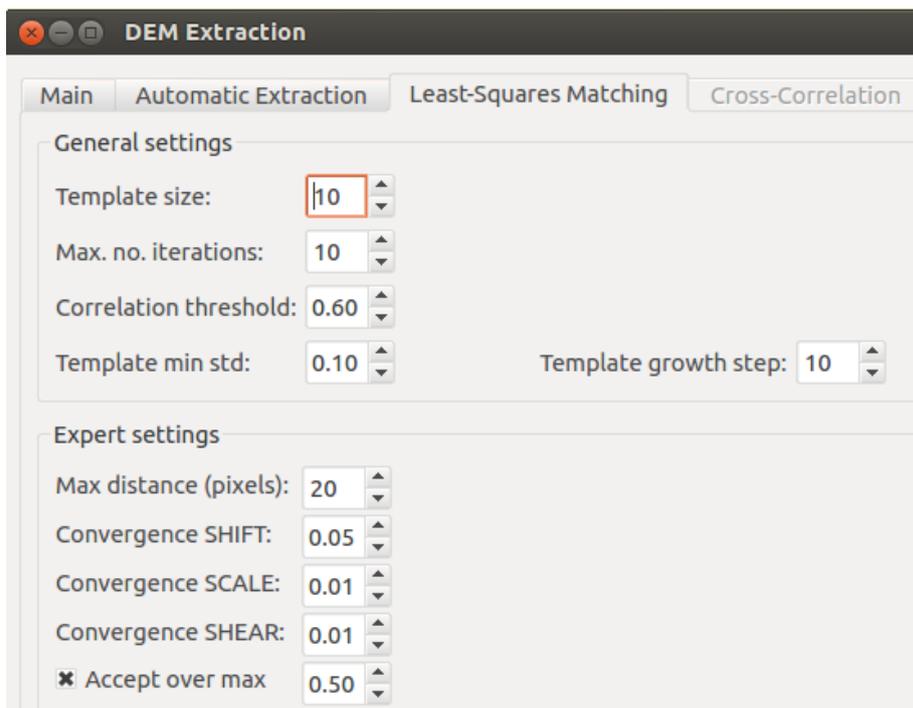
FIGURE 12.10. Least-Squares Matching sub-menu. The expert-setting items "Convergence SHIFT, SCALE, SHEAR" are related to the affine transformation introduced between the patches as discussed in section 12.4.

The result will be a grid file of point heights very similar to the image files used. This DEM file can be saved and used in many applications,
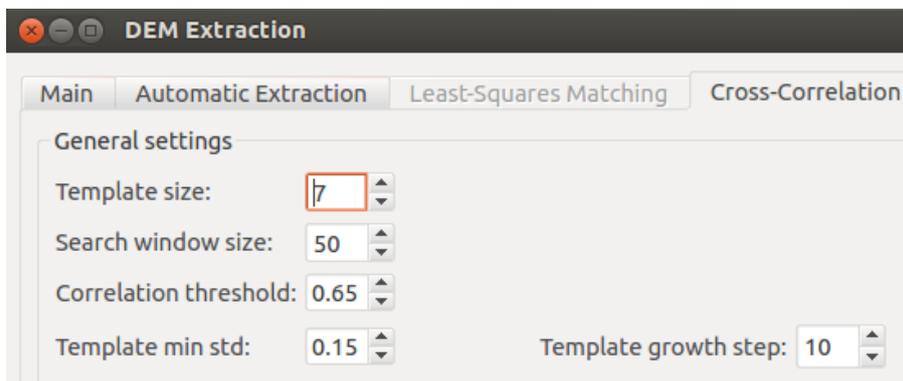
FIGURE 12.11. Cross-Correlation sub-menu. As in the Least-Squares Matching sub-menu, the user may enter the minimum Pearson correlation value required for a match, the minimum amount of greytone variability ("std", standard deviation) required to make the patch or "template" useable for correlation, and the way its size may be grown if there is too little such variability.
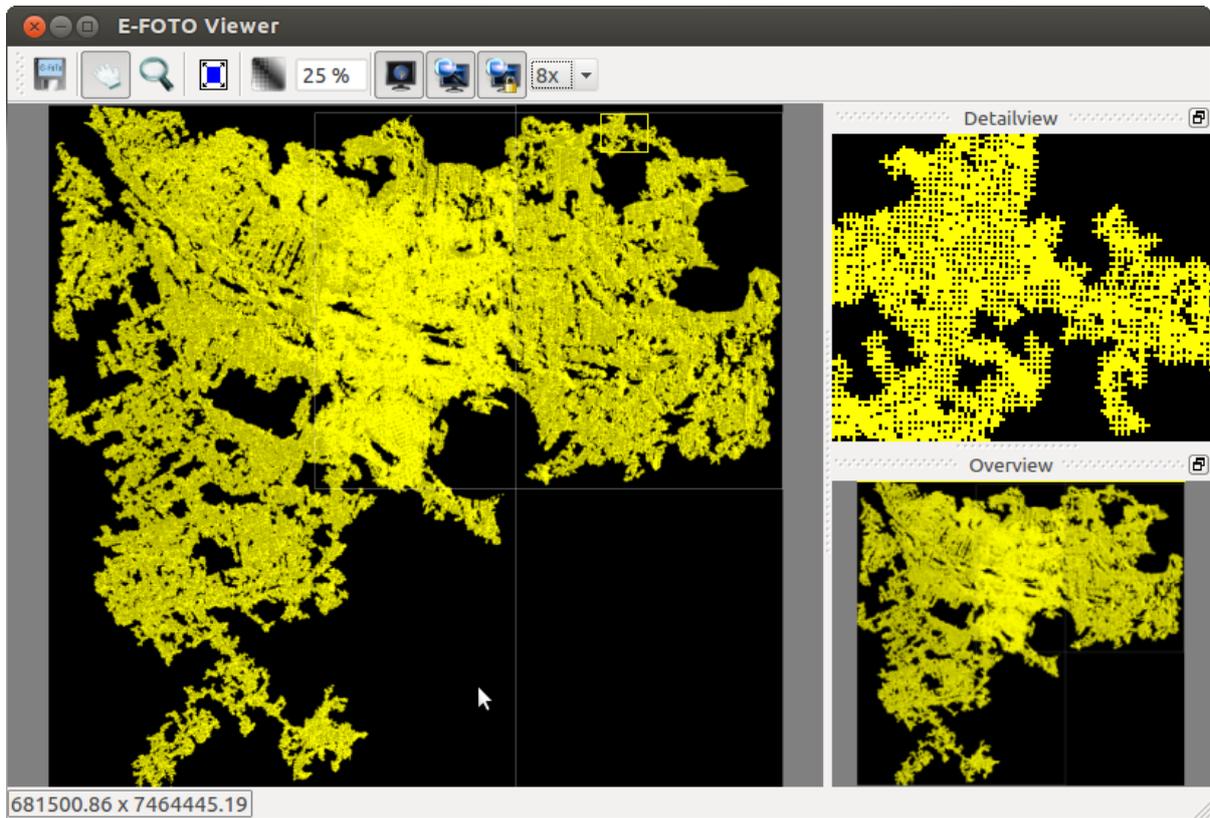
FIGURE 12.12. Result of areal expansion (Region Growing), for the Rio de Janeiro area.

including the production of orthophoto maps from the images.

The interpolation result may look like figure 12.14.

12.8   Manual DEM entry

It is clear that for urban areas — like all three test image sets — the automatic generation of a digital elevation model will not work very well.
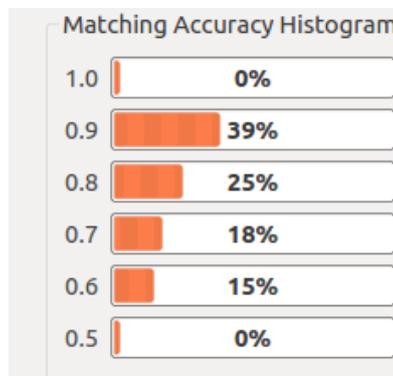
FIGURE 12.13. Matching Accuracy Histogram.
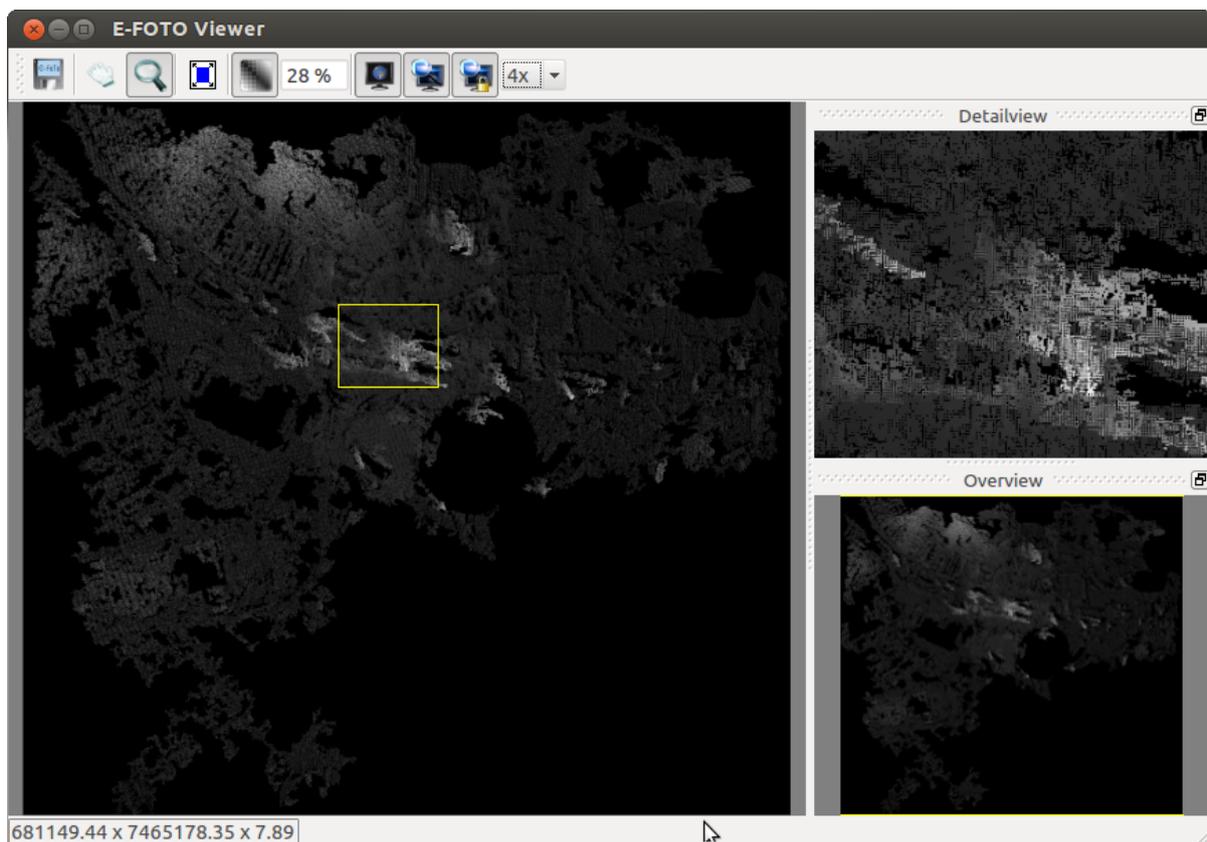
FIGURE 12.14. DEM interpolation result by e-foto. Coverage is still incomplete.

It is best suited for areas without buildings and smooth terrain, like open fields, savannah or desert. Forest might work, but the outcome could be that the DEM contains the forest canopy rather than the ground.

One available option is to *import* a DEM that has been produced by other means, e.g., airborne laser scanning or synthetic aperture radar (SAR) from aircraft or satellite.

For urban areas, the obvious alternative is to import a *feature file* that has been produced by the Stereoplotter module, containing the three-dimensional descriptors of point, line and polygon features like buildings, roads, parks etc. The DEM Extraction module can then *interpolate* an elevation model from this, good enough for use in orthophoto production.

See figure 12.15.
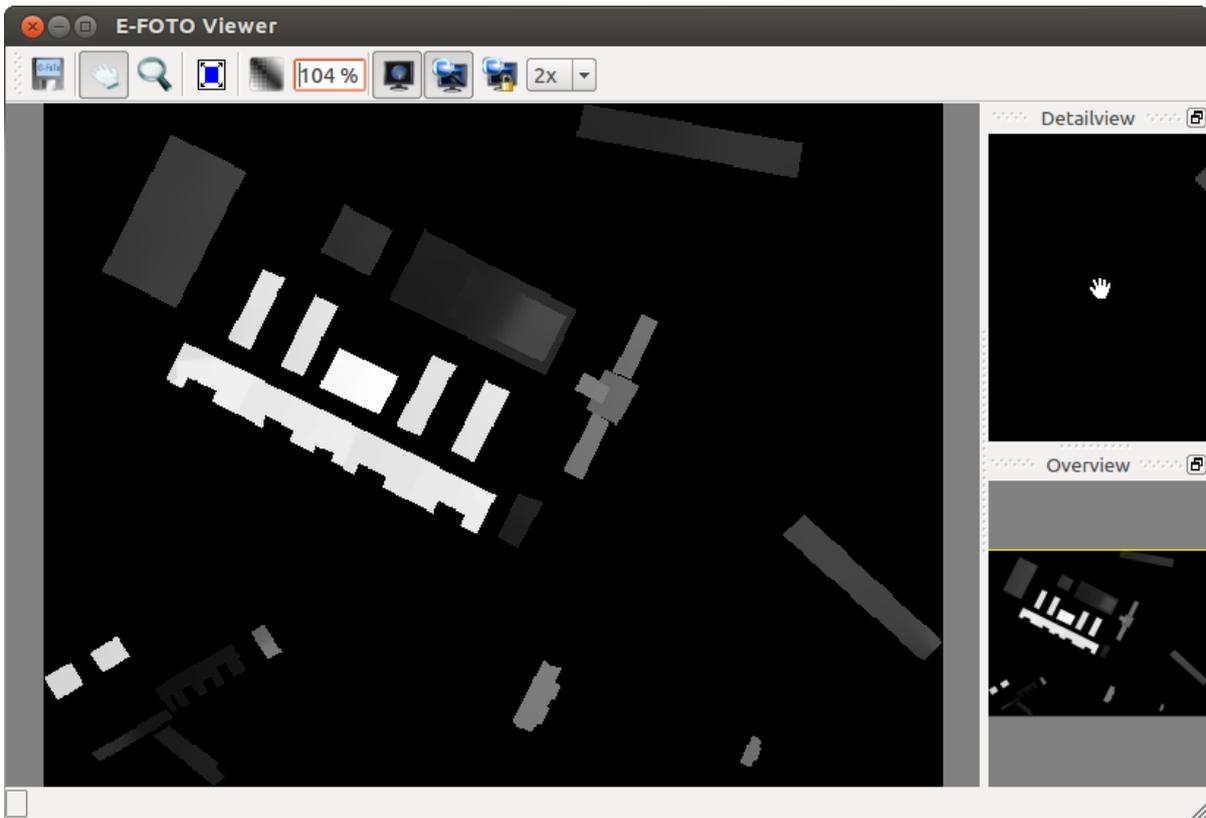
FIGURE 12.15. Manually generated DEM based on urban features.

12.9   Cleaning up the digital terrain model

12.9.1   Why cleaning up?

For an urban area, a digital elevation model (DEM) will contain many buildings, which is not what we usually call "terrain". So, if we are interested in generating an DTM, a "digital terrain model", that represents terrain features as commonly understood — like for generating terrain height contours to be included on a topographic map — then we must remove these buildings by editing the DTM.

However, if the *only* purpose of the DTM is to be used for ortho-rectification of the imagery, than this is unnecessary, and in fact not even desirable. If the DTM contains the buildings (meaning that it is a DEM), then in ortho-rectification, the roof edge of every building will be shifted to the correct (X, Y) location, which is precisely what we want. Every building will be represented by its roof outline on the orthophoto map produced.

The ERDAS Imagine software contains a facility for the manual removal of buildings, basically by drawing a polygon around the building at terrain
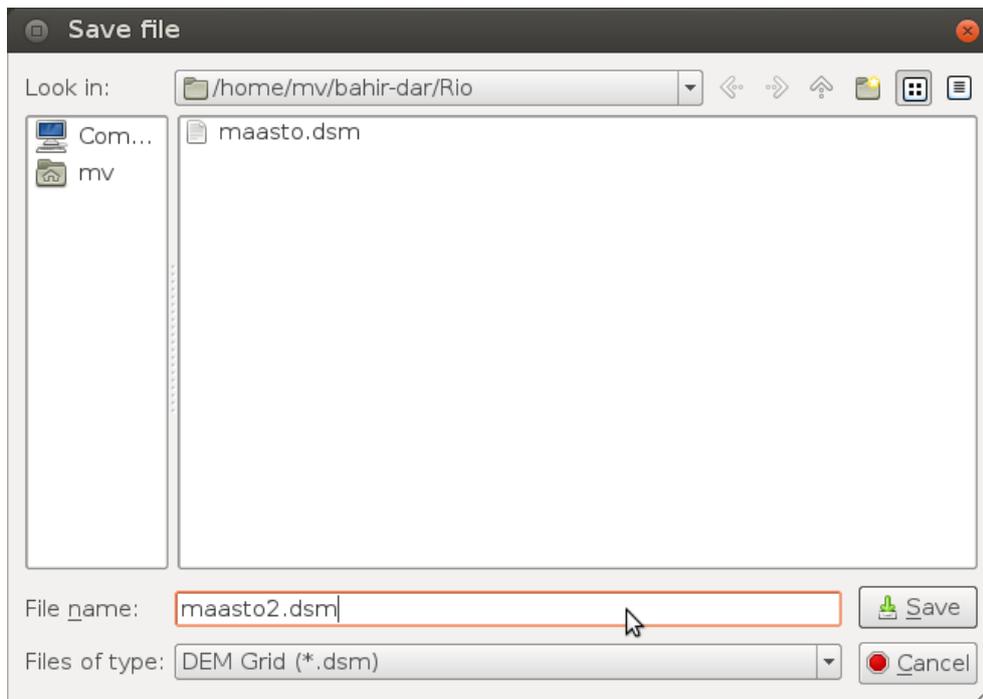
FIGURE 12.16. Saving the digital surface model file.

level, and then filling the polygon with values on that level.

e-foto does not offer any such facility, but, on the other hand, the e-foto DTM generation leaves the building areas uncovered anyway. These uncovered areas may now be interpolated by a *repeat application* of the DTM interpolator, using a sufficiently large interpolation radius. For this, you have to re-load the already interpolated DTM file again into the interpolator.

12.10 Saving the digital surface model

After completing the interpolation, you will want to *save* the digital surface model obtained into a file. This file will have the extension .dsm; you should write the full filename including extension before saving it, see figure 12.16. This file will be needed in the following operation, *ortho-rectification*.

12.11 Other DEM construction technologies

12.11.1 Radar

Interferometric Synthetic-Aperture Radar (InSAR) from satellites is a powerful and precise technique to build DEMs. It is based on sending a

radar pulse to Earth from a satellite on two different overpasses. The two different locations of the satellite antenna in space allow the construction, by *interferometry*, of a model of the terrain heights underneath. Alternatively and for higher resolution on the ground, two separate antennas on an aircraft are used in a single overflight.

12.11.2   Airborne laser scanning (LiDAR)

In this technique the terrain under the aircraft is scanned, typically whiskbroom wise, by a range-measuring laser beam. For airborne laser scanning, the preferred technique is a pulsed laser beam. The angle under which the beam goes out is steered by a rapidly rotating mirror and is a linear function of time for every sweep.

This technique allows, among many other things, the construction of *geocentric* terrain height models of high precision (several cm, depending on the nature of the terrain) and resolution (sub-m^2). Key to the success of airborne laser scanning is the positioning by GNSS of the scanning aircraft placing the height models obtained in a geocentric reference frame. Compared to using photogrammetry the workflow is largely automatic, explaining in part its current popularity. Precise calibration, and especially connecting the laser scanner's origin with the electromagnetic centre of the GNSS antenna, is however demanding.

Modern instruments allow the recording and use of multiple reflections, for example from the forest canopy, mid-level foliage, understorey, and forest floor. Then one can choose to construct either a digital elevation model (including the canopy) or a digital terrain model (only including the ground). In urban areas, for constructing a *terrain* elevation model, buildings will have to be removed, either manually or automatically.

Orthophoto mapping

13

13.1 Rationale

Before a set of aerial photographs, obtained as part of an aerial mapping project, can be used for actually drawing a map, it is necessary to extract the information in them in the form of three-dimensional co-ordinates. This is typically done by going through the sequence of interior and exterior orientation and formation of a stereo model, using given points in the terrain of which the true geocentric co-ordinates are known. After this, three-dimensional co-ordinates of arbitrary terrain points can be recovered from the photographs.

Digital elevation models or DEMs are a prerequisite for orthophoto mapping.

If we know the height of a point in the photograph above the reference surface (e.g., sea level), we may use this information to derive the true horizontal position (X, Y) from the position in the photograph, and from our knowledge of the absolute position (X_0, Y_0, Z_0) and absolute orientation of the camera optical centre. See figure 13.1.

Another way to look at this is that we want to map the landscape as it would look from infinitely far away, looking straight down from the zenith. This is known in cartography as an *orthographic projection* (greek: *ortho* = straight).

The procedure for doing so may be automated, allowing the automatic drawing, by optical projection, of the aerial photograph onto a light sensitive emulsion with every pixel being in the correct (X, Y) position. This method is known as *orthophoto mapping*.

According to the World Bank, orthophoto mapping is the preferred method for cadastal mappings (Timo Linkola, personal communication; see also for example Hailu and Harris (2014)).
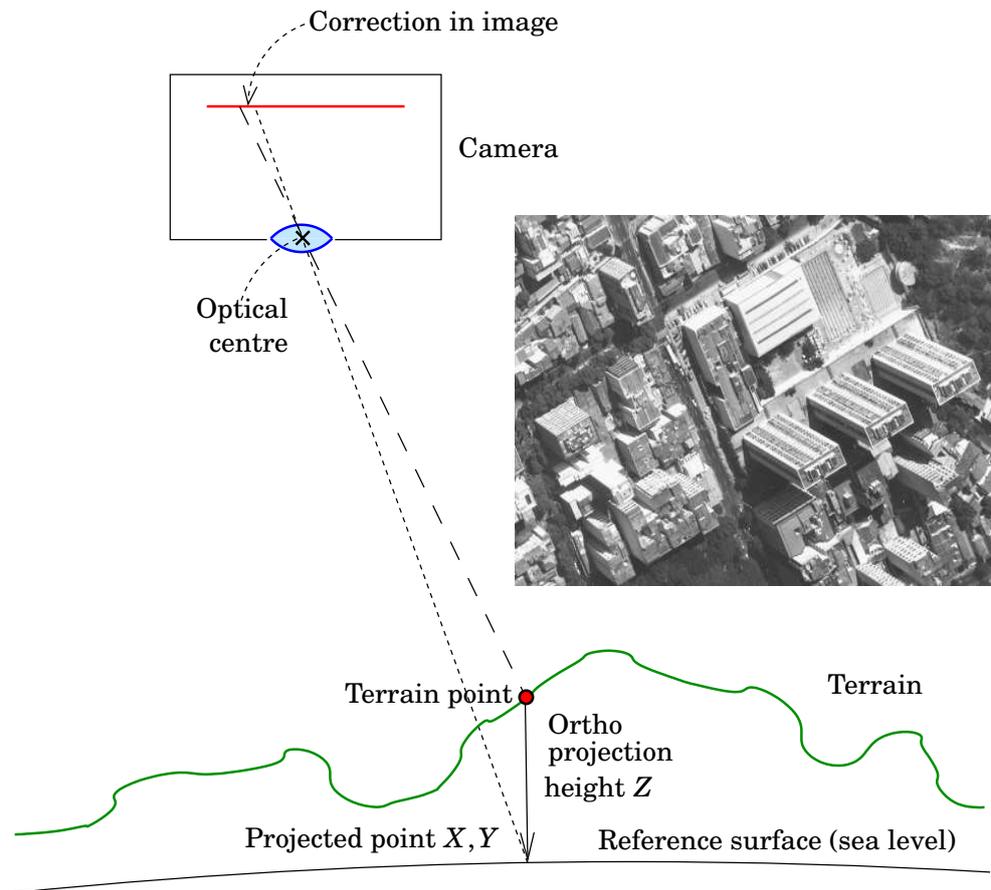
FIGURE 13.1. Orthophoto mapping. Notice how buildings are "falling over" outside the image centre (photo detail from Rio de Janeiro). The correction in the image, computed using a digital elevation model of the terrain, is applied in an orthophoto mapping instrument.

13.2 Advantages and drawbacks

The great *advantage* of the method of orthophoto mapping is, that it is rapid, requiring little additional manual work after the acquisition of the imagery and its stereo restitution. However, a digital elevation or surface model (digital terrain model) or DEM, $Z(X,Y)$ of the area to be mapped must be available. Traditionally such a model was constructed from the stereo restitution itself, which is able to provide three-dimensional co-ordinates for every terrain point captured on two images. Nowadays often pre-existing DEMs are used, typically acquired using the technique of airborne laser scanning — also called LiDAR, Light Detection And Ranging, by analogy with radar.

A *disadvantage* of orthophoto mapping is, that the plotted information is purely visual, coming directly from the aerial photograph without any
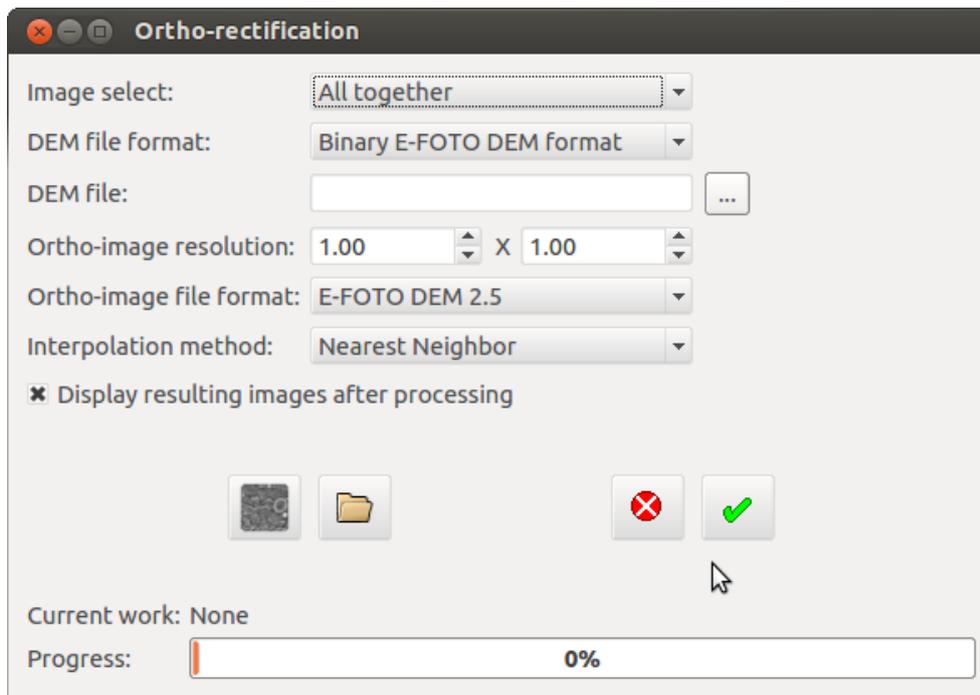
FIGURE 13.2. Ortho-rectification in e-foto.

interpretation. The user is expected to do the interpretation. In practice, interpreted information is added to the map from independent sources: place names, the road network, elevation contours, etc. Such a map may be rapidly produced yet be almost as useable as a traditionally produced map.

Orthophoto maps are extensively used in order to provide a first, rough cut at mapping an area on which little information pre-exists, and where time is of the essence. An example of this is a military situation where no good maps of enemy territory are available. Another use case is large, thinly populated areas which have not been previously mapped at large scale, and where the collection of detailed information by traditional means would be prohibitively expensive against its expected utility.

13.3 Ortho-rectification with e-foto

In this step, we eliminate from the photographic images the effect of "falling over" away from the image's centre due to the height of objects in the terrain. I.e., we re-draw the images with the content of every pixel shifted, in space, vertically downward to sea level, in the image, toward the nadir, the point straight under the aircraft, by an amount depending on the pixel's height.

E-foto will do the ortho-rectification computations for every image in turn, which will take a little time. Then, it presents its results in the image viewer.

Applications

14

14.1 Cadastral system, zoning, and aerial mapping

In a modern, high-investment economy, real estate property and real rights or encumbrances on it — such as mortgage — are essential financial instruments. They need to be reliably recorded, including property boundaries. For this, there is the *cadastral system*. The precision with which boundaries need to be recorded, depends on the value of the land: in areas of low land value, such as deserts, the precision requirement is low, whereas in areas of high value, such as urban areas and city centres, the precision requirement will be high. This is referred to as various *measurement classes*.

As one example we give here the classification presented in the Finnish regulatory document *"Drafting a Local Zoning Map"*, also known as JHS 185. See table 14.1.

It is possible to use aerial mapping techniques for taking inventory of real-estate boundaries, if these boundaries are already marked in the terrain by physical features. Such features may be hedges, stone fences, streams, or whatever physical distinguishing marks that are somewhat permanent and visible from the air. This is the so-called "General Boundaries" system which is in use in many Anglo-Saxon countries. It works well provided the precision required is not very great, and if the system is supported by law and agreement by land owners to place or grow these boundary markers between their properties and accept their legitimacy. Advantages are speed and economy.

Where the land is more valuable and mapping precision requirements are higher, greater precision and definiteness of boundary markers are required, and proper surveys have to be carried out. This can be done with traditional geodetic survey techniques, but also from the air. However, the

TABLE 14.1. Measurement classes, from the Finnish publication JUHTA (JHS 185). Note that lake or sea shore (measurement class 3) is especially valued in Finland for its recreational use.

Survey areas are divided into *three measurement classes*. The measurement class defines the accuracy of measurement and graphical rendering.

Measurement class 1: Urban areas where land is very valuable and where there is a valid local zoning plan with a binding parcel division, or a construction ban awaiting the drafting of such a plan.

In surveys intended to be incorporated into the municipal GIS for use in technical planning requiring great accuracy, a higher accuracy level may be used (*measurement class 1e*).

Map scale $1 : 500$ or $1 : 1000$. Parcel boundary co-ordinate accuracy (mean error or standard deviation) ± 0.085 m.

Measurement class 2: Urban areas for which the local zoning plan to be drafted does not require binding parcel division.

Map scale $1 : 1000$ or $1 : 2000$. Parcel boundary co-ordinate accuracy ± 0.14 m.

Measurement class 3: Areas zoned as lake or sea shore, lake or sea shore areas, and other areas where the land is clearly more valuable than agricultural or forest land, e.g., so-called dispersed settlements (Wikipedia, Dispersed Settlement).

Map scale is usually $1 : 2000$. Parcel boundary co-ordinate accuracy ± 0.21 m.

A scale of $1 : 4000$ or $1 : 5000$ may be accepted if a zoning plan can be drafted with such a map without essentially compromising the requirements to be placed on it.

Measurement class 4: All other (unzoned) areas.

A digital map has no scale. The accuracy of the collected data corresponds to that on a map of a certain scale. One should not print out graphical products from a digital data base at a scale greater than the accuracy of the data allows.

In choosing the measurement method and the scale of the map, one must take into account surveys carried out earlier in the area or its surroundings, and the extent and character of the area.

work of marking (signalising) the boundaries so they are visible from the air can be substantial. It may be speedier and cheaper to directly measure the boundaries using for example an automated tacheometer and prism staff, or GPS in real-time kinematic (RTK) mode. The latter requires a second GPS receiver or base station on a known location nearby, adding to cost. A careful comparison of logistics and costs, both of hardware and of labour, should be made before the choice of method.

14.2   Land-use inventory and aerial mapping

Here the techniques discussed in the previous chapter find application.

- Mapping current land use patterns

- Assessing demand on irrigation water, transport capacity, electric grid power, . . .

- Agricultural production estimates for domestic use and export, preparation for shortages due to drought

- Urbanisation, also uncontrolled, and related infrastructure demand.

14.3   Maps in support of land-use planning

- Large infrastructures: roads, railroads, bridges, dams, canals, aqueducts, airports

- Electricity, high-voltage lines, consumption / load planning. Hydro, solar

- Soil quality and use.

14.4   Maps in natural resource management

- Geophysical surveys: gravity, magnetism, terrain, ores and mineral resources, deep aquifers, . . .

- Vegetation and forest inventory

- Natural habitat protection

- etcetera.

14.5   Water resources, hydro power, height systems

In mountainous countries the building of reservoir dams both for regulating river water and its availability for irrigation, and for the generation of electrical power, is attractive on a large scale. However, the engineering task requires a proper mapping of the area affected especially regarding heights. This concerns not only the area of the dam itself, but especially the area behind the dam where water will accumulate over a long area of river valley, and the area downstream serviced by the irrigation water. Deficiencies in this knowledge may lead to unpleasant surprises.

There needs to be a good vertical reference system. This can be established by precise levelling, which is however laborious and challenging

especially in mountainous terrain. Satellite positioning (GNSS) is an alternative method for establishing a vertical reference, but requires the availability of a geoid model. Either method will establish a sparse network of points or lines of known height throughout the area. Areally dense knowledge of the terrain height in the area, in the form of digital elevation models, can then be acquired using aerial survey. Such knowledge is needed in order to predict what areas will be covered by water when the reservoir fills up, in support of evacuation, resettlement, and compensation. It also allows estimation of the total water volume, energy stored, and areas reachable by irrigation water.

More precise geodetic survey techniques are required to monitor the structural integrity of the dam, and its deformations and those of the surrounding Earth's crust. Filling up reservoirs typically causes small deformations and even microseismicity, which need to be monitored for safety reasons. Aerial survey techniques may be useful for this, though their precision is limited. Airborne laser scanning may be an attractive option. In all cases, a precise geodetic base network tied to the non-deforming surroundings is a requirement.

As a matter of general interest, the amount of water stored behind reservoir dams worldwide amounts to some 30 millimetres sea-level equivalent (Chao et al., 2008)!

Datums in geodesy and photogrammetry

15

15.1   Geodetic reference systems

In order to describe, in geodesy and photogrammetry, the locations of points in numbers, we need *co-ordinate reference systems*. A reference system describes the nature of the co-ordinates that give the location of a point in the system, consisting of

1. The nature of the co-ordinates used: are they rectangular co-ordinates, spherical co-ordinates, ellipsoidal co-ordinates, or something else?

2. the way the origin and the orientation of the co-ordinate axes are chosen: is the origin in or near the geocentre (the centre of mass of the Earth), or is it on or near the Earth's surface? Is one of the co-ordinate system's axes oriented along the Earth's rotation axis, or along the local plumbline? And does another axis point to a location on the Earth's surface, or to a location on the celestial sphere? Or to the local North, or along some instrumentally defined zero direction?

Every answer to these questions defines a *co-ordinate reference system*. Note that we haven't yet talked about geodetic networks and observations at all, which we must do if we are to achieve our goal of quantifying point locations.

15.2   Geodetic reference frames and datums

Geodetic measurements, in the form of geodetic measurement networks, allow us to to tie a co-ordinate reference system to the physical Earth, producing a *co-ordinate reference frame*. This is referred to as *realising* the reference system.

However, when actually trying to compute a geodetic network, it turns out that there usually are things that need to be fixed in order to arrive at a unique solution.

For example when adjusting — that is, computing — a levelling network, one must conventionally fix the height of one point in order to find unique height values for all the points in the network. This is an example of a *datum defect*: a number that must be provided before a unique solution becomes possible. Typically, the point will be at sea level, close to a *tide gauge* monitoring local sea-level variations. The height value chosen for the point will make mean sea level at this tide gauge equal to zero. This conventional choice defines a geodetic *datum*.

For two- or three-dimensional co-ordinate reference systems, the datum fixing issue becomes more complicated, as we shall see.

15.3 Geocentricity

In geodesy and photogrammetry we express co-ordinates, of ground control points, of the photographing aircraft, or of the terrain being mapped, in some co-ordinate reference frame. This frame will be uniquely tied to the Earth by network measurements, and by fixing the geodetic *datum*.

Traditionally, datums were always local, or at best national or regional. This was because measurement technologies available were terrestrial in nature. Using theodolites and spirit levels, one could only measure within small areas on the Earth's surface. Thus, these techniques could not provide an overview over large areas, or of the Earth as a whole.

This changed with the introduction of satellite positioning technologies, in other words GNSS, Global Navigation Satellite Systems. The first modern such system was the American GPS, the Global Positioning System. These technologies allow the creation of *geocentric* datums, tied to the Earth's centre of mass as their origin, and to the Earth's rotation axis for their orientation.

Such global datums are based on so-called terrestial reference systems (TRS), like the International Terrestial Reference System (ITRS) of the International Association of Geodesy (IAG). This system has been the basis for a series of global datums or *realisations* called International Terrestial Reference Frames (ITRF).

15.4 Connecting local and global reference frames

The problem arises how to connect these global, geocentric reference frames or datums — which are typically highly accurate — with the traditional, non-geocentric, often national, datums. This connection must be made in situations where, for example, the imaging aircraft has been positioned in flight using GNSS technology, but the co-ordinates of ground control points visible in the imagery are provided in a traditional datum. In the days before satellite positioning you wouldn't have to worry about this. Nowadays, however, GNSS on board the aircraft produces *geocentric locations* $\left(\overline{X}_0, \overline{Y}_0, \overline{Z}_0\right)$ of the aircraft for the moment each image was exposed. These locations must be *transformed before use* to map positions (X, Y) and heights above sea level Z *in the same reference frame* as used for the ground control points. And this is also the reference frame — for both map co-ordinates and heights above sea level — that the photogrammetric mapping mission will use for its end product.

There are *two issues* here:

1. in which co-ordinate reference frame the ground control is given, and

2. in which co-ordinate reference frame do we wish the final result of the mapping mission to be presented.

Relatively simple techniques based on local rectangular co-ordinates are still commonly used when all point co-ordinates are given in the same datum. Typically this will be a local and non-geocentric datum. Then, many approximations are valid due to the limited size of the study area, without loss of precision.

15.5 Topocentric co-ordinates

15.5.1 Definition

Topocentric[1] co-ordinates are a special case of local co-ordinates, where the origin of the co-ordinate frame is some given point on the Earth's surface, like the location of a geodetic instrument or measurement station.

Topocentric co-ordinates are sometimes called "local astronomical".

There are as many local systems as there are points of origin, or instrument locations, that can be chosen on the Earth's surface: in fact,

---

[1]From the Greek *topos*, meaning place.
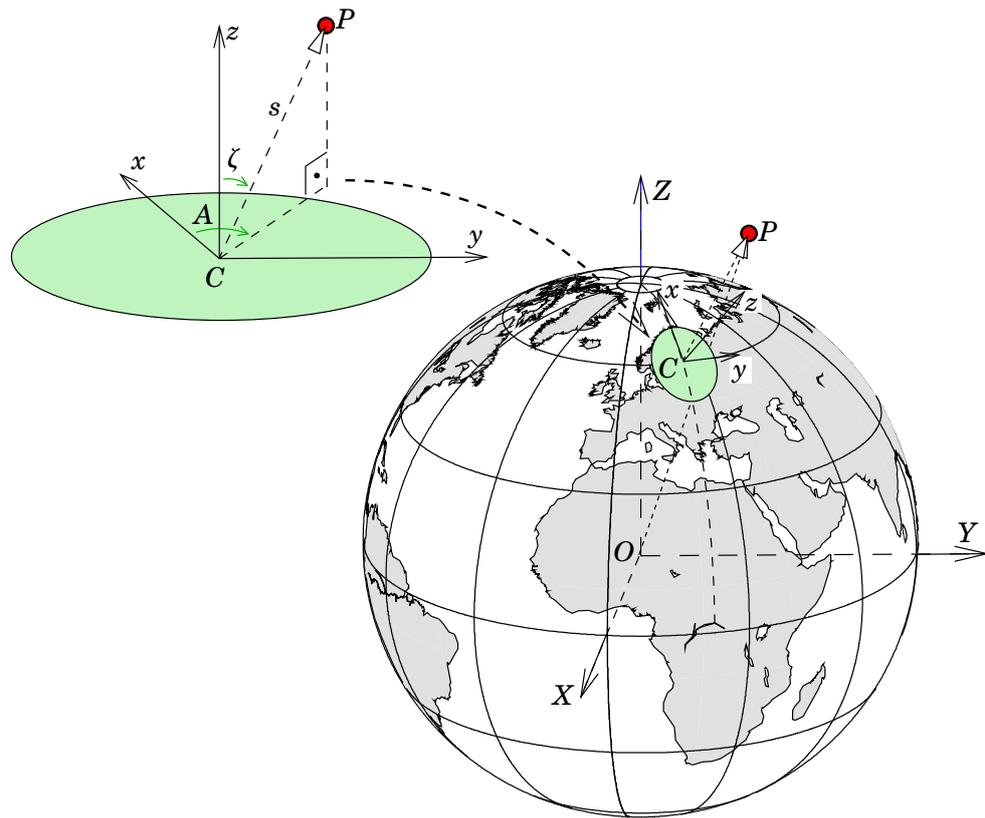
FIGURE 15.1. Topocentric or local astronomical co-ordinates (x, y, z), and how
they connect to geocentric co-ordinates (X, Y, Z).

infinitely many.

15.5.2 Rectangular and spherical

In the rectangular version of topocentric co-ordinates, the system's axes
are as follows:

1. the z axis points to the local zenith, straight up.

2. The x axis points to the local north.

3. The y axis is perpendicular to both others and points east.

This is sometimes referred to as a NEU ("North-East-Up") system. In
figure 15.1 the situation of a local topocentric system in the global context
is depicted.

The spherical co-ordinates of the same topocentric system are:

○ the azimuth, or compass direction angle, A. In geodesy, reckoned
from north $(A = 0°)$ through east $(A = 90°)$.

zeta ζZ   ○ The zenith angle ζ, reckoned from the zenith downward; alterna-
eta ηH   tively the elevation angle $\eta = 100\,\mathrm{gon} - \zeta = 90° - \zeta = \frac{\pi}{2} - \zeta$, reckoned

from the horizon upward.

  ○ The distance s.

The transformation between a point's topocentric spherical co-ordinates and its topocentric rectangular co-ordinates is

$$
\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} s \cos A \sin \zeta \\ s \sin A \sin \zeta \\ s \cos \zeta \end{bmatrix}.
$$

The inverse transformation is

$$
\begin{aligned}
s &= \sqrt{x^2 + y^2 + z^2}, \\
\zeta &= \arctan \frac{\sqrt{x^2 + y^2}}{z}, \\
A &= 2 \arctan \frac{y}{x + \sqrt{x^2 + y^2}}.
\end{aligned}
$$

The last formula is known as the *half-angle formula* and avoids the problem of finding the correct quadrant[2] for the azimuth angle A. The result is in the interval $\left(-180°, 180°\right]$, and negative values may be incremented by 360° to make them positive.

15.5.3 Transforming from geocentric to topocentric and back

Often we will want to connect directions and orientations in a local topocentric system and a global geocentric system, for example when using GNSS positioning locally in a differential (relative) mode.

Let (X, Y, Z) be a geocentric co-ordinate system and (x, y, z) a topocentric *instrument co-ordinate system*. In an instrument co-ordinate system the x axis points to the zero direction of the *instrument* instead of north. As examples, in the case of a theodolite, this would be the zero direction on the horizontal circle, or in the case of an aerial camera, the left-to-right direction in the image in normal orientation.

Now we can symbolically write, with the notations $\mathbf{X} = \begin{bmatrix} X & Y & Z \end{bmatrix}^\mathsf{T}$ and $\mathbf{x} = \begin{bmatrix} x & y & z \end{bmatrix}^\mathsf{T}$, the *same* vector in the geocentric and topocentric systems, respectively:

$$
\mathbf{x} = R_1 \overline{R}_2 R_3 \left(\mathbf{X} - \mathbf{X}_C \right),
$$

---

[2] In computer code, one may use the function $\mathrm{atan2}(y, x)$ for the same purpose.
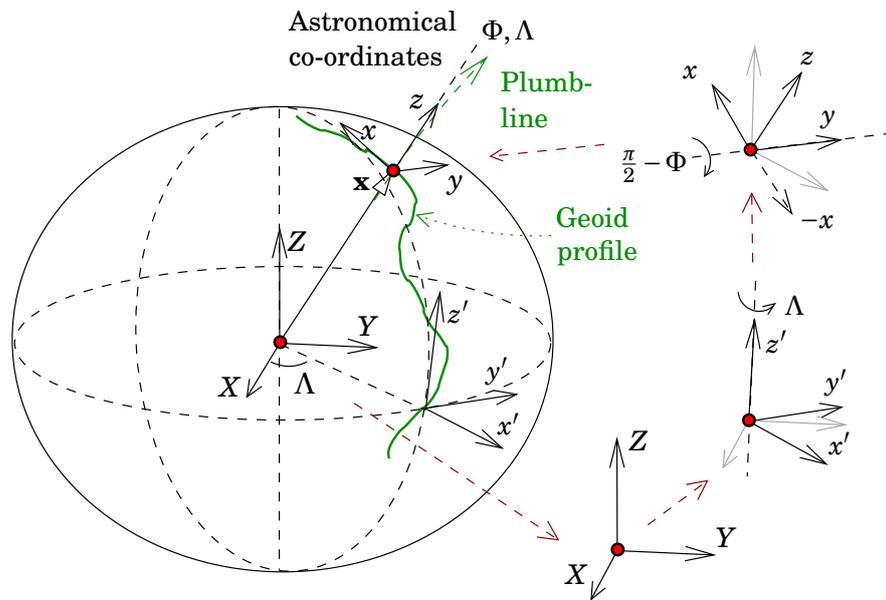
FIGURE 15.2. From the geocentric to the topocentric system. The matrix R_1 mentioned in the text is left out here.

where the rotation matrices $R_3, \overline{R}_2, R_1$ act in succession to transform $\mathbf{X}$ into $\mathbf{x}$. See figure 15.2. $\mathbf{X}_C$ denotes the co-ordinates of the local origin C, expressed in the geocentric system.

The inverse transformation chain of this is

$$\mathbf{X} = \mathbf{X}_C + R_3^\mathsf{T} \overline{R}_2^\mathsf{T} R_1^\mathsf{T} \mathbf{x},$$

as can be easily derived by multiplying the first equation *from the left* by the matrix $R_1^\mathsf{T} = R_1^{-1}$, then by the matrix $\overline{R}_2^T$, and then by the matrix R_3^T, and finally by moving $\mathbf{X}_C$ to the other side.

R_3 rotates the co-ordinate frame around the z axis from the Greenwich meridian to the local meridian of the observation site, rotation angle Λ, astronomical longitude):

lambda $\lambda \Lambda$

$$R_3 = \begin{bmatrix} \cos\Lambda & +\sin\Lambda & 0 \\ -\sin\Lambda & \cos\Lambda & 0 \\ 0 & 0 & 1 \end{bmatrix}. \qquad (15.1)$$

Seen from the direction of the z axis we see (figure 15.3), that

$$\begin{aligned} x' &= x\cos\Lambda + y\sin\Lambda, \\ y' &= -x\sin\Lambda + y\cos\Lambda. \end{aligned}$$

(The correct algebraic signs should always be established by the aid of a sketch! Also, the directional conventions of different countries may differ.)
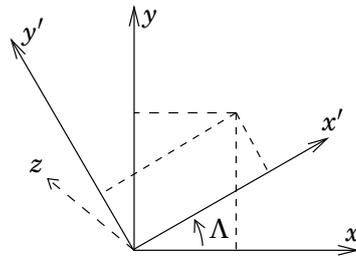
FIGURE 15.3. Rotation by an angle Λ around the z axis.

$\overline{R}_2$ turns the co-ordinate frame around the y axis, in such a way that the z axis points to the North celestial pole instead of to the zenith. The rotation angle needed for this is $90° - \Phi$, where Φ denotes astronomical latitude. Furthermore, the new x axis needs to be mirrored to point to the North instead of the South. This gives us a "rotation-plus-mirroring matrix"

phi $\varphi\phi\Phi$

$$\overline{R}_2 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \sin\Phi & 0 & -\cos\Phi \\ 0 & 1 & 0 \\ +\cos\Phi & 0 & \sin\Phi \end{bmatrix} =$$

$$= \begin{bmatrix} -\sin\Phi & 0 & \cos\Phi \\ 0 & 1 & 0 \\ +\cos\Phi & 0 & \sin\Phi \end{bmatrix}. \tag{15.2}$$

R_1 rotates the co-ordinate frame around the new z axis or vertical axis by the amount A_0, after which the x axis points to the azimuth of the instrumental zero, for example the zero point of the instrument's horizontal circle:

$$R_1 = \begin{bmatrix} \cos A_0 & +\sin A_0 & 0 \\ -\sin A_0 & \cos A_0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

15.6 Traditional geodetic datums on the reference ellipsoid

The classical approach to geodetic datums is based on the use, and local fit, of a *reference ellipsoid*. Based on local terrestial measurements only, one cannot make this reference ellipsoid genuinely or precisely geocentric. However, by using astronomical geodetic measurements — of astronomical latitude and longitude of network points and azimuth of network sides — it is possible to give the datum a proper *orientation* with respect to the
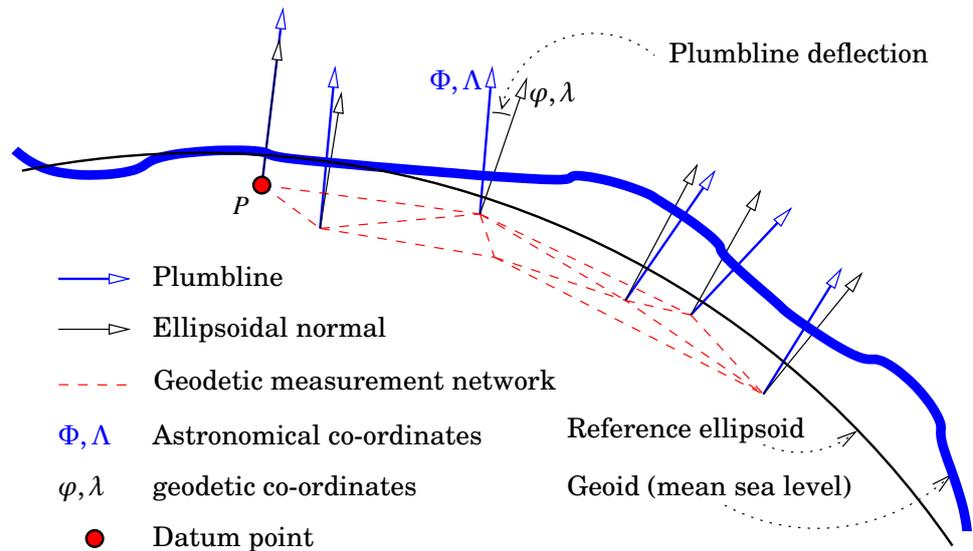
FIGURE 15.4. Geodetic and astronomical co-ordinates.

Earth as a whole. For example the North direction within the datum will be the true North astronomically.

The ellipsoid as a reference surface for geodetic measurements will, thus constructed, not be uniquely defined. Not even the measurements of astronomical latitude and longitude will help: these determine the direction of the *plumbline* or vertical, which, due to Earth curvature, will change from place to place, already over a distance of a hundred metres. These changes are partly irregular, due to the gravity field of the Earth being itself irregular because of the irregular distribution of masses inside the Earth.

One can however construct a reference ellipsoid which *locally* fits as well as possible to the figure of the Earth. This means that in every location within the area of study, the *ellipsoidal normal* is as close as possible to the plumbline. In other words, the *plumbline deflections* are as small as possible.

There are two widely used techniques for achieving this:

1. A traditional geodetic datum is defined by, in addition to the shape parameters (a, f) of the reference ellipsoid used, also fixing

   xi $\xi\Xi$
   eta ηH

   ○ The components of the plumbline deflections (ξ_P, η_P) in some starting point or *datum point* P (see figure 15.4) and

   ○ The height of the geoid N_P above the reference ellipsoid in datum point P.

This means that, in total, five (5) parameters must be fixed con-

ventionally to starting values in a datum point in order to obtain a datum definition. In this way the reference ellipsoid surface on which the geodetic calculations are done is positioned in space.

An example of such a geodetic datum is ED50 (European Datum 1950), which used the Frauenkirche, a church in Munich, Germany as its datum point.

2. We *minimise* the *regional* plumbline deflections and geoid heights in a least-squares sense:

$$\text{minimize} \quad \sum_{i=1}^{n}(\xi_i^2 + \eta_i^2) \quad \text{and} \sum_{i=1}^{n} N_i^2,$$

where ξ_i, η_i, and N_i are the plumbline deflections and geoid heights of all the measurements point (triangulation points) $i = 1, \ldots, n$ in the area. With this approach, their *mean values* over all these points will be set to zero.

In other words, with this approach there is not a single datum point but an *ensemble* of datum points. The *common centre of mass* of all these points together serves as the datum point.

In both approaches one may also, instead of using a starting point or ensemble of starting points in the measurement area, fix the *centre* of the reference ellipsoid. Also then, three parameters, the three co-ordinates[3] (X_0, Y_0, Z_0) of this centre relative to the Earth's centre of mass, need to be fixed. We again have a total of five parameters: a, f, X_0, Y_0, and Z_0.

In the example case of the ED50 datum, the values (X_0, Y_0, Z_0) are of order of magnitude 100 m. This is typical for traditional geodetic datums: one cannot, by traditional measurement techniques, determine the absolute deflection of the plumbline at the datum point — its deviation from the normal on a *geocentric* elllipsoid — any better than this.

For the sake of argument, let the size of this deviation amount to $1''$. This will cause at the centre of the Earth — a distance of 6378 km removed from the datum point — an "error" of $\frac{6378\,\text{km}}{(\rho \cdot 3600)} \approx 28\,\text{m}$.[4] This equals the size of the offset of the centre of the reference ellipsoid from the centre of mass of the Earth. And this offset will be different for every different geodetic datum, see figure 15.5.

---

[3]Not to be confused with the terrain co-ordinates of the camera optical centre, for which we use the same notation.

[4]Here, ρ is the length of one radian expressed in degrees; i.e., $\rho = 57.3$, and $\rho \cdot 3600 = 206266$.
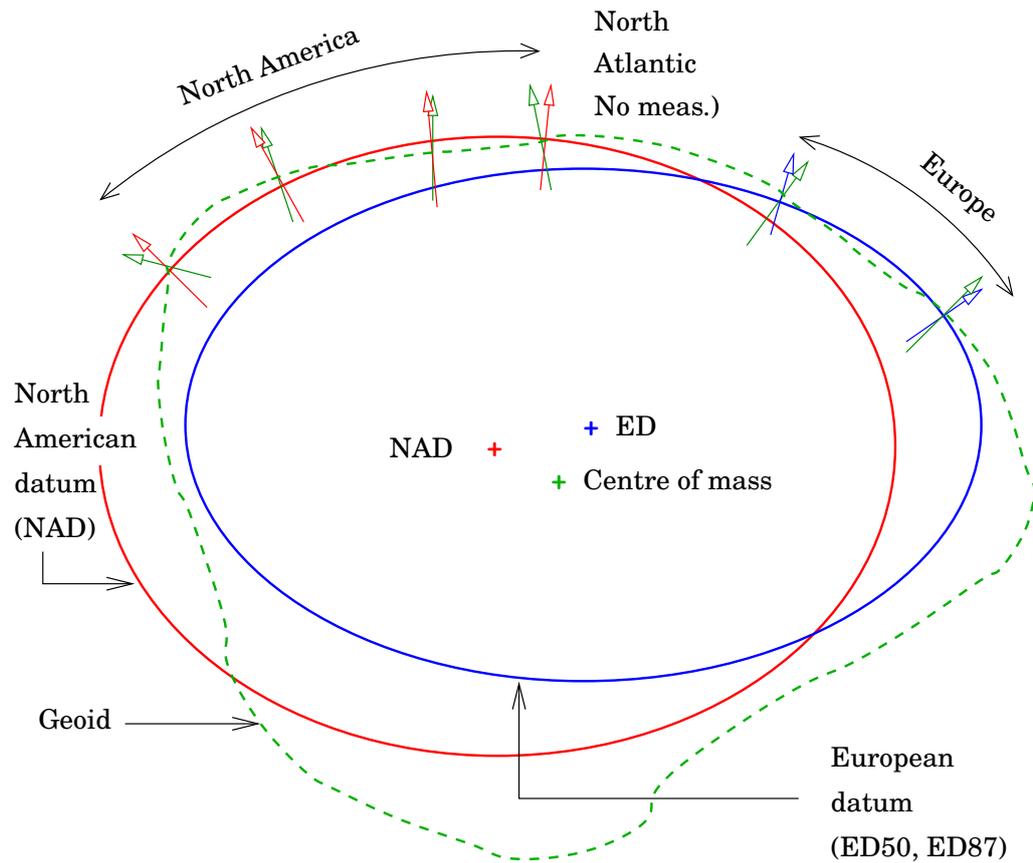
FIGURE 15.5. Regional datums.

Modern datums or co-ordinate reference frames are based on satellite positioning techniques. This includes the international geodetic community's ITRS (International Terrestrial Reference Frame), like ITRF2008, or in Europe, EUREF89, and the various WGS84 datums created by the United States Defense Department's operators of the Global Positioning System GPS. For such datums, due to the use of satellites orbiting the Earth, the co-ordinate origin will automatically be in the Earth's centre of mass. Locations are thus always obtained *geocentrically*. Only the uncertainties of the measurement process itself, including the satellite orbit determination, may still cause offsets from geocentricity on the several centimetre level.

15.7 Local and global datums: cases

We will discuss the issue using three different cases: the Finnish KKJ system, the newer geocentric Finnish system EUREF-FIN, and the Ethiopian Adindan datum.

In the following, "WGS84" is shorthand for the ITRS realisation

(ITRF*yy*) in which the satellite ephemeris used in the computation are given. For the best precision, this too needs to be handled correctly.

KKJ: This system is strongly non-geocentric: the offsets of the centre of the International Ellipsoid of 1924 it uses, relative to the centre of the ellipsoid of WGS84, are (Ollikainen, 1993) $\Delta X = 93\,\mathrm{m}$, $\Delta Y = 103\,\mathrm{m}$, and $\Delta Z = 123\,\mathrm{m}$. These offsets are in the direction WGS84 $\rightarrow$ KKJ. Additionally there are non-zero rotations and a non-unity scale factor.

A further complication is that, while KKJ is a map projection system (the Gauss-Krüger projection on the International Ellipsoid of 1924) based upon the European datum ED50, in the map plane there is a further Helmert transformation applied to achieve approximate compatibility (on the metre level) with an older system in use before 1970, when KKJ was officially introduced.

The "raw" transformation between KKJ and WGS84 only achieves an accuracy on the metre level. Thus, the transformation must be considered approximate. The National Land Survey has however created a triangulated affine transformation (between KKJ and EUREF-FIN, see below) based upon a Delaunay triangulation of the Finnish territory, which achieves better (several cm level) precision.

EUREF-FIN: This system is geocentric and based on satellite positioning. There are no translation parameters (shifts of origin) between WGS84 and EUREF-FIN, but there is a rotation matrix: EUREF-FIN is the Finnish national realisation of ETRS89, the European Terrestrial Reference System, which at epoch 1989.0 coincided with ITRS (the international geodetic community's version of WGS84), but has since rotated away due to the tectonic motion of the Eurasian plate of the Earth's crust, which ETRS eliminates. EUREF-FIN co-ordinates can be transformed to and from "WGS84" — so, to and from any realisation of ITRS — with high precision. The map projection used for mapping the whole country is UTM (Universal Transverse Mercator) zone 35 (central meridian 27°) on the GRS80 reference ellipsoid.

Adindan: Also this datum is strongly non-geocentric: $\Delta X = -162\,\mathrm{m}$, $\Delta Y = -12\,\mathrm{m}$, and $\Delta Z = 206\,\mathrm{m}$ (in the direction Adindan $\rightarrow$ WGS84) for the Ethiopian territory (Thomas Dubois, personal comm. April 30, 2013). This transformation is conventionally defined and may

be assumed exact.[5] So Ethiopia uses a non-geocentric GNSS-based national datum, the connection of which to WGS84 is exactly known.[6] The rotation parameters are set to zero and the scale to 1. The recommended projection for mapping the whole country is UTM zone 37 on the Clarke 1880 ellipsoid.

One should be careful using local, non-geocentric datums, especially if the transformations to and from WGS84 are *not precise*, as is always the case for datums based on *traditional* measurements. In that case, GNSS positioning of the aircraft is a complication we don't really need, as its superior precision is then lost down the processing chain. The problem in this case is that ground co-ordinates can only be *approximately* transformed to geocentric WGS84 co-ordinates, and conversely, aircraft image exposure co-ordinates (X_0, Y_0, Z_0) from GNSS can also only be obtained *approximately* in this local datum. In this case, one should use these transformed image exposure co-ordinates only as *approximate values* and allow the photogrammetric software to improve them by least-squares adjustment.

In fact, in this case there is no reason to introduce these aircraft image exposure co-ordinates at all: better to let the software handle the whole computation in local project co-ordinates.

This will give good results in practice for photogrammetry, but is not really an option for airborne laser scanning missions (section 17.1.1) which rely critically on the practice of GNSS positioning the aircraft: in that case a good solution is to handle the whole computation chain in WGS84 or a geocentric datum close to it.[7] A final transformation then obtains co-ordinates in the local datum.

It is also not an option if we have precisely geocentrically positioned ground control, but wish the mission result to be nevertheless in a local datum. Then we need to use the appropriate transformation and

---

[5]For the older, traditional Adindan datum, older documents list $\Delta X = -166\,\text{m}$, $\Delta Y = -15\,\text{m}$, and $\Delta Z = 204\,\text{m}$, with uncertainties of 5, 5, and 3 m. Rotation parameters and scale were presumably estimated and deemed insignificant against their uncertainties.

[6]. . . at least *at* the epoch of its determination, not many years ago. Over time, the movement of the African plate will assert itself, and Ethiopia will have to define a, presumably AFREF-based, precise national geodetic datum in which this effect has been eliminated.

[7]In practice one chooses the ITRF datum in which also the GNSS satellites' precise ephemeris used were provided by the international geodetic community.
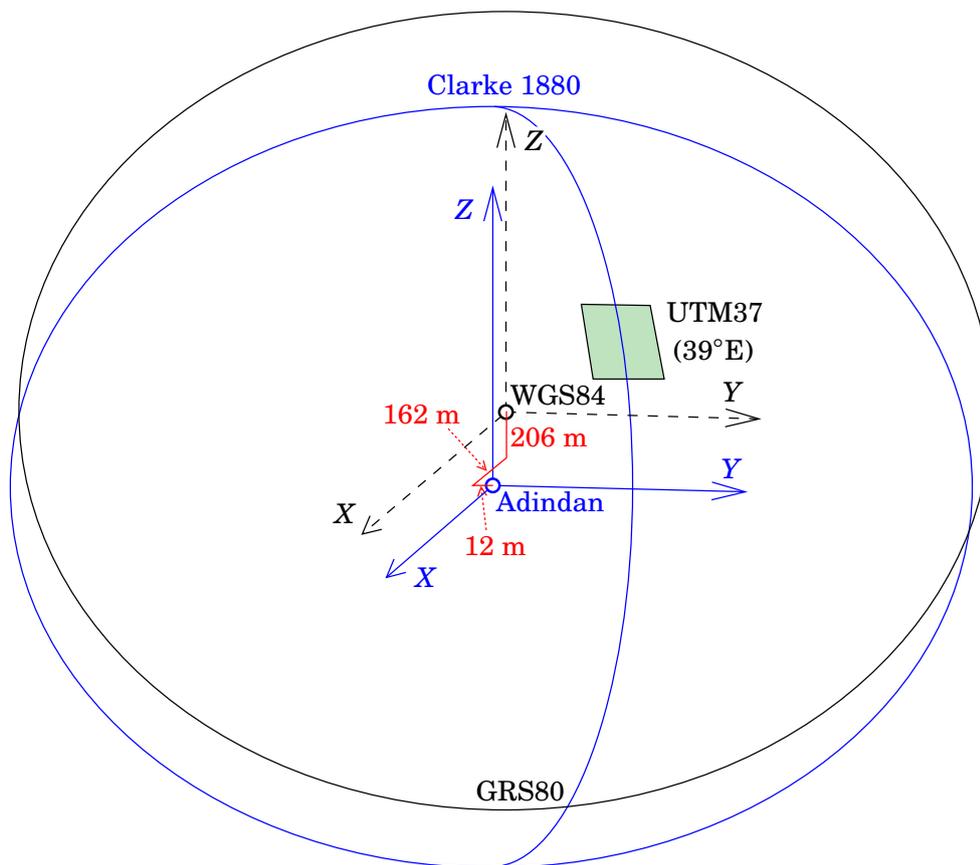
FIGURE 15.6. The Ethiopian "Adindan" datum in relation to WGS84.

projection logic, and make sure these are *precise*.

15.8 The geoid

The currently best *global* geoid, or geopotential, model is EGM2008, published in 2008 by the U.S. National Geospatial-Intelligence Agency. It has a resolution of approximately 2.5 minutes of arc, corresponding to approximately 4 km in position. It is already incorporated in modern geodetic and photogrammetric software.

Many countries have local geoid models, which are typically more detailed than EGM2008, and tied to the national height system. An Ethiopian precise geoid determination effort — or rather, its modern hi-tech equivalent — is documented in Bedada (2010). It uses the innovative approach, made possible by information technology, of computing directly the geopotential in three dimensions, by interpolating in a pre-computed 3-D grid, for any point on or above the terrain where GPS provides the geocentric co-ordinates $\left(\overline{X}, \overline{Y}, \overline{Z}\right)$. Instrumental in achieving a precise
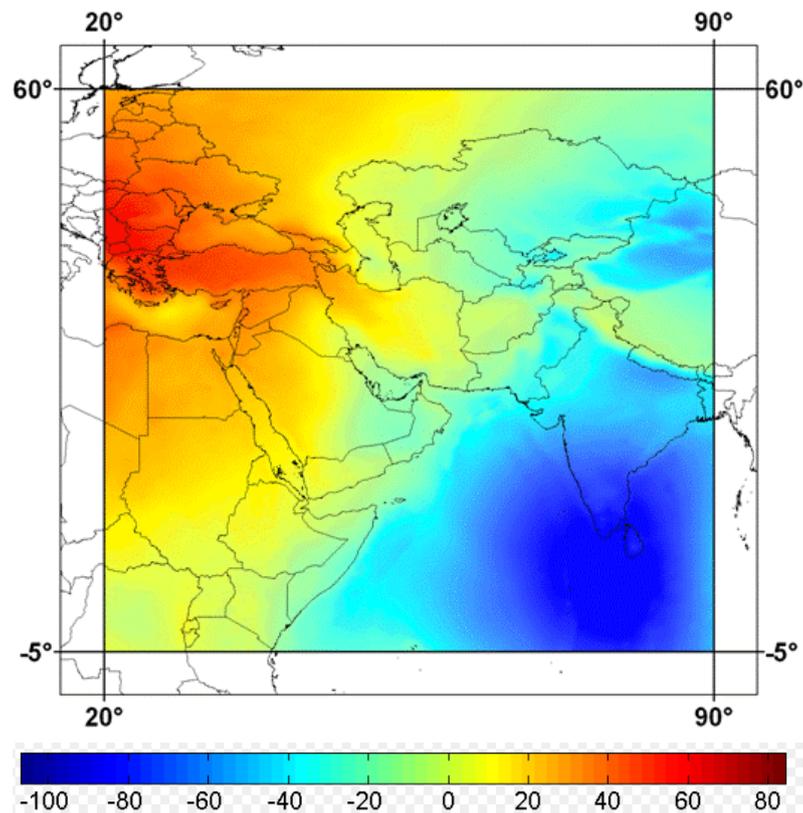
FIGURE 15.7. EGM2008 geoid for East Africa and the Middle East. Geoid scale in metres. © U.S. National Geospatial-Intelligence Agency (Pavlis et al., 2012).

result was the use of the Ethiopian Airborne Gravity Survey data (Bedada, 2010), as well as topographical data from the Shuttle Radar Topography Mission (SRTM).

The pre-computed grid can be incorporated into GNSS receivers and used in real time in the field.

Datum transformations

16

Traditionally, geodetic datums are defined on the reference ellipsoid, by performing a geodetic network adjustment in which the location of the network, and its orientation, are fixed with the aid of additional, astronomical information. This *datum fix* is done by

1. fixing, for either a single point, or for an ensemble of points, the *direction of the ellipsoidal normal* (φ, λ) and the height h *from the ellipsoid*. Or alternatively, by

   phi $\varphi\phi\Phi$
   lambda $\lambda\Lambda$

2. fixing the location of the centre X_0, Y_0, Z_0 of the reference ellipsoid used, relative to the Earth's centre of mass.

Here we will show how these two alternative ways of datum fixing are related to each other.

Rectangular geocentric co-ordinates may be written as follows, in *spherical approximation*:

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = (R + h) \begin{bmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ \sin\varphi \end{bmatrix} + \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix},
$$

where (φ, λ, h) are latitude, longitude and height above the reference ellipsoid, and the centre of this sphere is located at $\begin{bmatrix} X_0 & Y_0 & Z_0 \end{bmatrix}^\mathsf{T}$.

The precisely geocentric co-ordinates $\begin{bmatrix} X & Y & Z \end{bmatrix}^\mathsf{T}$ of the *same* point can now be written out in two *different* datums (meaning referred to two different *reference ellipsoids*). In other words, the vector expressions

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \left(R + h^{(1)} \right) \begin{bmatrix} \cos\varphi^{(1)}\cos\lambda^{(1)} \\ \cos\varphi^{(1)}\sin\lambda^{(1)} \\ \sin\varphi^{(1)} \end{bmatrix} + \begin{bmatrix} X_0^{(1)} \\ Y_0^{(1)} \\ Z_0^{(1)} \end{bmatrix}
$$

and

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \left(R + h^{(2)} \right) \begin{bmatrix} \cos\varphi^{(2)}\cos\lambda^{(2)} \\ \cos\varphi^{(2)}\sin\lambda^{(2)} \\ \sin\varphi^{(2)} \end{bmatrix} + \begin{bmatrix} X_0^{(2)} \\ Y_0^{(2)} \\ Z_0^{(2)} \end{bmatrix}
$$

must be *identical*. Here, the superscripts (1) ja (2) denote geodetic co-ordinates (φ, λ, h) computed in two different datums, on two different reference ellipsoids, with their origins $\begin{bmatrix} X_0 & Y_0 & Z_0 \end{bmatrix}^{\mathsf{T}}$ in two different geocentric locations. We assume here that the *orientations* of the co-ordinate axes are the same in both datums.

Let us define the following *difference quantities*:

$$
\begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix} = \begin{bmatrix} X_0^{(2)} \\ Y_0^{(2)} \\ Z_0^{(2)} \end{bmatrix} - \begin{bmatrix} X_0^{(1)} \\ Y_0^{(1)} \\ Z_0^{(1)} \end{bmatrix},
$$
$$
\Delta h = h^{(2)} - h^{(1)},
$$
$$
\Delta\varphi = \varphi^{(2)} - \varphi^{(1)},
$$
$$
\Delta\lambda = \lambda^{(2)} - \lambda^{(1)},
$$

then we obtain by subtraction (note the delta notation):

$$
0 = \left(\left(R + h^{(2)} \right) \begin{bmatrix} \cos\varphi^{(2)}\cos\lambda^{(2)} \\ \cos\varphi^{(2)}\sin\lambda^{(2)} \\ \sin\varphi^{(2)} \end{bmatrix} \right) -
$$
$$
- \left(\left(R + h^{(1)} \right) \begin{bmatrix} \cos\varphi^{(1)}\cos\lambda^{(1)} \\ \cos\varphi^{(1)}\sin\lambda^{(1)} \\ \sin\varphi^{(1)} \end{bmatrix} \right) + \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix} =
$$
$$
= \Delta \left\{ (R + h) \begin{bmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ \sin\varphi \end{bmatrix} \right\} + \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix}.
$$

Now, by *partial derivation* (assuming all Δ quantities to be small, so that *linearisation* is allowed):

$$
\frac{\partial}{\partial h} \left((R + h) \begin{bmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ \sin\varphi \end{bmatrix} \right) = \begin{bmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ \sin\varphi \end{bmatrix},
$$

and similarly with respect to φ and λ:

$$
0 = \begin{bmatrix} \dfrac{\partial}{\partial h} & \dfrac{\partial}{\partial \varphi} & \dfrac{\partial}{\partial \lambda} \end{bmatrix} \overbrace{\left((R+h) \begin{bmatrix} \cos\varphi\cos\lambda \\ \cos\varphi\sin\lambda \\ \sin\varphi \end{bmatrix} \right)} \begin{bmatrix} \Delta h \\ \Delta\varphi \\ \Delta\lambda \end{bmatrix} + \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix} =
$$

$$
= \underbrace{\begin{bmatrix} \cos\varphi\cos\lambda & -(R+h)\sin\varphi\cos\lambda & -(R+h)\cos\varphi\sin\lambda \\ \cos\varphi\sin\lambda & -(R+h)\sin\varphi\sin\lambda & +(R+h)\cos\varphi\cos\lambda \\ \sin\varphi & +(R+h)\cos\varphi & 0 \end{bmatrix}} \begin{bmatrix} \Delta h \\ \Delta\varphi \\ \Delta\lambda \end{bmatrix} + \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix}.
$$

Here we have used, as argued above, the *identity* of the geocentric co-ordinates $\begin{bmatrix} X & Y & Z \end{bmatrix}^{\mathsf{T}}$ in the two different datums.

Because $h \ll R$:

$$
\begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix} \approx \begin{bmatrix} -\cos\varphi\cos\lambda & +R\sin\varphi\cos\lambda & +R\cos\varphi\sin\lambda \\ -\cos\varphi\sin\lambda & +R\sin\varphi\sin\lambda & -R\cos\varphi\cos\lambda \\ -\sin\varphi & -R\cos\varphi & 0 \end{bmatrix} \begin{bmatrix} \Delta h \\ \Delta\varphi \\ \Delta\lambda \end{bmatrix} =
$$

$$
= \begin{bmatrix} -\cos\varphi\cos\lambda & +R\sin\varphi\cos\lambda & +R\cos\varphi\sin\lambda \\ -\cos\varphi\sin\lambda & +R\sin\varphi\sin\lambda & -R\cos\varphi\cos\lambda \\ -\sin\varphi & -R\cos\varphi & 0 \end{bmatrix} \begin{bmatrix} \Delta N \\ -\Delta\xi \\ -\Delta\eta/\cos\varphi \end{bmatrix}.
$$

This equation gives the relationship that exists between the amount of non-geocentricity of the reference ellipsoid used, and the datum defined by it in the point (φ, λ, h). At the same time it is also the equation by which the differences $(\Delta h, \Delta\varphi, \Delta\lambda)$ between two geodetic datums in a starting point may be converted to translation components of the origin $\begin{bmatrix} \Delta X_0 & \Delta Y_0 & \Delta Z_0 \end{bmatrix}^{\mathsf{T}}$. In other words the equation by which *transformation parameters* can be converted from their topocentric to their geocentric form. phi $\varphi\phi\Phi$ lambda $\lambda\Lambda$

We have also expressed $\begin{bmatrix} \Delta h & \Delta\varphi & \Delta\lambda \end{bmatrix}^{\mathsf{T}}$, the topocentric translation vector, into the alternative form $\begin{bmatrix} \Delta N & -\Delta\xi & -\Delta\eta/\cos\varphi \end{bmatrix}^{\mathsf{T}}$, where appear the geoid undulation N and the deflections of the plumbline (ξ, η), all three evaluated in the datum point. See figure 16.1. Note that[1] xi $\xi\Xi$ eta ηH 1

$$
N = h - H, \qquad \xi = \Phi - \varphi, \qquad \eta = (\Lambda - \lambda)\cos\varphi,
$$

where

N geoid undulation from the reference ellipsoid ("geoid height")

---

[1]The extra factor $\cos\varphi$ in the η equation comes from the meridian convergence. At higher latitudes a given change in longitude $\Delta\lambda$ corresponds to an ever smaller distance on the Earth's surface, and thus to an ever smaller change in the direction of the vertical.

≡ ↑ 🖾 ⊞ 🔍 🗐 ✛
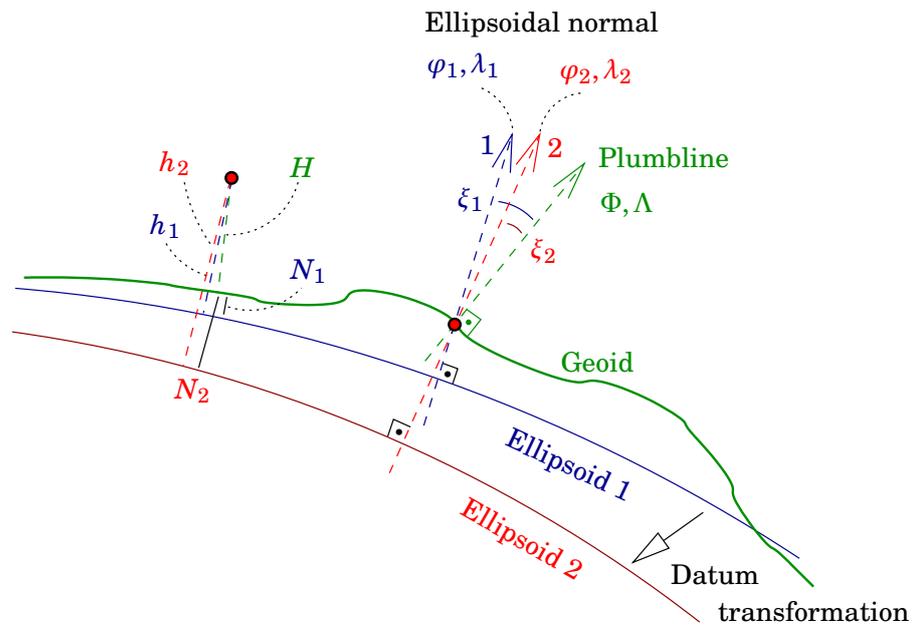
FIGURE 16.1. The effect of a datum transformation (reference ellipsoid shift) on the geodetic latitude and longitude λ and λ, the deflections of the plumbline ξ and η, the geoid undulation N, and the ellipsoidal height h of points.

h	height of the point above the ellipsoid ("ellipsoidal height")
H	height of the point above the geoid ("orthometric height")
ξ, η	deflections of the plumbline: the differences in direction between the astronomical vertical ("plumbline") and the normal on the reference ellipsoid — ξ in the north-south and η in the west-east direction
Φ, Λ	astronomical latitude and longitude: the direction of the plumbline
φ, λ	geodetic latitude and longitude: the direction of the ellipsoidal normal.

xi $\xi\Xi$
eta $\eta\mathrm{H}$

phi $\varphi\phi\Phi$
lambda $\lambda\Lambda$

[2] Because the quantities H, Φ, and Λ are physical quantities[2] — neither the direction of the local plumbline relative to the starry sky, nor the height of the point above the geoid or "sea level", depend on the reference ellipsoid chosen — it follows that

$$\Delta N = \Delta h, \qquad \Delta \xi = -\Delta \varphi, \qquad \Delta \eta = -\Delta \lambda \cos \varphi,$$

from which the alternative form above follows.

---

[2]Orthometric height H can be determined by spirit levelling from the coast (together with gravimetric measurements for reduction), and astronomical latitude and longitude Φ, Λ by astronomical measurements. Neither technique assumes a reference ellipsoid.

The inverse formula of the above is:

$$
\begin{bmatrix} \Delta N \\ -\Delta \xi \\ -\Delta \eta / \cos \varphi \end{bmatrix} = \begin{bmatrix} \Delta h \\ \Delta \varphi \\ \Delta \lambda \end{bmatrix} =
$$

$$
= \frac{1}{R} \begin{bmatrix} -R \cos \varphi \cos \lambda & -R \cos \varphi \sin \lambda & -R \sin \varphi \\ +\sin \varphi \cos \lambda & +\sin \varphi \sin \lambda & -\cos \varphi \\ +\dfrac{\sin \lambda}{\cos \varphi} & -\dfrac{\cos \lambda}{\cos \varphi} & 0 \end{bmatrix} \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix}.
$$

These equations simplify somewhat, if we write the topocentric translation into the form[3] $\begin{bmatrix} \Delta N & -R \Delta \xi & -R \Delta \eta \end{bmatrix}^{\mathsf{T}}$, because then the matrices are clean rotation matrices:

xi $\xi \Xi$

eta ηH

3

$$
\begin{bmatrix} \Delta N \\ R \Delta \xi \\ R \Delta \eta \end{bmatrix} = \begin{bmatrix} \Delta h \\ R \Delta \varphi \\ R \cos \varphi \Delta \lambda \end{bmatrix} =
$$

$$
= \begin{bmatrix} -\cos \varphi \cos \lambda & -\cos \varphi \sin \lambda & -\sin \varphi \\ -\sin \varphi \cos \lambda & -\sin \varphi \sin \lambda & +\cos \varphi \\ -\sin \lambda & +\cos \lambda & 0 \end{bmatrix} \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix}.
$$

16.2   Helmert transformations in three dimensions

Above the transformation between two geodetic datums on reference ellipsoids was described as a *translation, a shift, in space only*: a three-parameter transformation. For traditional geodetic datums defined on non-geocentric reference ellipsoids this is usually a good enough approximation.

However, satellite positioning technology (GNSS, Global Navigation Satellite Systems), changed the game: the superior positioning precision makes also more precise transformation models necessary, and the model that offers itself is the seven-parameter Helmert transformation in three-dimensional space.

The Helmert transformation formula in three dimensions is[4]

4

$$
\mathbf{X}' = \mu R \left(\mathbf{X} - \Delta \mathbf{X}_0 \right), \tag{16.1}
$$

---

[3]Also referred to as the "North-East-Up" form — though here, arbitrarily, "Up" comes first.

[4]Note the similarity with the *collinearity equations* 8.1 for connecting terrain and camera co-ordinates! There too, we have a shift vector, a rotation matrix, and a scale factor.

where the three-dimensional, geocentric vectors of location are

$$\mathbf{X}' = \begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix}, \qquad\qquad \mathbf{X} = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix},$$

$$\Delta\mathbf{X}_0 = \begin{bmatrix} \Delta X_0 \\ \Delta Y_0 \\ \Delta Z_0 \end{bmatrix} = \begin{bmatrix} X'_0 \\ Y'_0 \\ Z'_0 \end{bmatrix} - \begin{bmatrix} X_0 \\ Y_0 \\ Z_0 \end{bmatrix}$$

is the shift between the origins of the $\mathbf{X}$ and $\mathbf{X}'$ systems, and

$$R = R_3(\alpha_3)\,R_2(\alpha_2)\,R_1(\alpha_1)$$

mu μM   is a "chaining" of three rotations. μ is the scale factor. Equation 16.1 is called the three-dimensional *Helmert* or *similarity transformation*. The elements of the R matrix are complicated expressions in the sines and

alpha αA   cosines of the angles α_1, α_2, and α_3 and we will not derive them here.

🗺️ 16.3   Transformation in case of small rotation angles

Often the axes of two co-ordinate systems are almost aligned with each
5   other. In that case, the angles of rotation are small[5] and one may make the approximations $\sin\alpha \approx \alpha$ and $\cos\alpha \approx 1$. Then all equations simplify.

mu μM   If furthermore we assume that the scale factor μ is close to unity, we may write

$$\mu = 1 + \Delta\mu.$$

Furthermore

$$R_1(\alpha_1) \approx \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \alpha_1 \\ 0 & -\alpha_1 & 1 \end{bmatrix},$$

$$R_2(\alpha_2) \approx \begin{bmatrix} 1 & 0 & -\alpha_2 \\ 0 & 1 & 0 \\ \alpha_2 & 0 & 1 \end{bmatrix},$$

$$R_3(\alpha_3) \approx \begin{bmatrix} 1 & \alpha_3 & 0 \\ -\alpha_3 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

alpha αA   When all α_i are small, one may also assume that all products $\alpha_i\alpha_j \approx 0$,

---

[5]An example from aerial photogrammetry is the camera tilt angles ω,φ, which are always small, so the camera z, or c, axis will be almost parallel with the local terrain vertical axis Z, or H.

≡ ↑ 🖼 ▦ 🔍 📄 ✧

and it follows

$$
R = R_3(\alpha_3)\,R_2(\alpha_2)\,R_1(\alpha_1) \approx
\begin{bmatrix}
1 & \alpha_3 & -\alpha_2 \\
-\alpha_3 & 1 & \alpha_1 \\
\alpha_2 & -\alpha_1 & 1
\end{bmatrix}
= I + \Delta R,
$$

where I is the 3×3 unit matrix, and

$$
\Delta R =
\begin{bmatrix}
0 & \alpha_3 & -\alpha_2 \\
-\alpha_3 & 0 & \alpha_1 \\
\alpha_2 & -\alpha_1 & 0
\end{bmatrix}
$$

is a skew-symmetric (antisymmetric) matrix: $\Delta R^{\mathsf{T}} = -\Delta R$.

From this follows the correction formula for co-ordinates:

$$
\mathbf{X}' - \mathbf{X} = (\Delta\mu + \Delta R)\,(\mathbf{X} - \Delta\mathbf{X}_0) =
\begin{bmatrix}
\Delta\mu & \alpha_3 & -\alpha_2 \\
-\alpha_3 & \Delta\mu & \alpha_1 \\
\alpha_2 & -\alpha_1 & \Delta\mu
\end{bmatrix}
(\mathbf{X} - \Delta\mathbf{X}_0), \quad (16.2)
$$

where the parameters $\Delta\mu, \alpha_1, \alpha_2, \alpha_3, \Delta\mathbf{X}_0$, as well as the left-hand side $\mathbf{X}' - \mathbf{X}$, are all *small* — but $\mathbf{X}$ is *large*.

mu μM
alpha αA

The form 16.2 is that of the general similarity transformation in the case of two co-ordinate reference system realisations that are close together, for example between two different ITRS realisations. In this case the rotation angles α_i are of order of magnitude fraction of a second of arc, and the translation or shift vector $\Delta\mathbf{X}_0$ is under 10 cm.

16.4   Case: transformation between ED50 and EUREF89

ED50 (European Datum 1950) is a *traditional* geodetic datum for Western Europe, created shortly after the Second World War by combining and re-adjusting the primary triangulation networks of the Western European countries. The reference ellipsoid used in the joint adjustment was the Hayford or International Ellipsoid of 1924.

EUREF89 is a *modern, satellite based, geocentric* geodetic reference system, created in the 1990s by the IAG's Subcommission for the European Reference Frame (EUREF) for Europe. The modern geodetic datums of the European member states are based on it.

This transformation between them is of type 16.1, into a precisely geocentric system. Because the Hayford ellipsoid — on which the ED50 datum is defined — is not (precisely) geocentric, it becomes necessary to translate the origin of the co-ordinate system to the centre of mass

TABLE 16.1. Helmert transformation parameters EUREF89 → ED50 determined for the Finnish territory by Ollikainen (1993). The parameters differing significantly ($> 3\sigma$ level) from zero are in red and underlined.

Parameter	Value	Uncertainty (s.d.)	Unit
ΔX	93.477	±3.345	m
ΔY	103.453	±5.534	m
ΔZ	123.431	±2.736	m
α_1	−0.246	±0.168	″
α_2	0.109	±0.106	″
α_3	0.068	±0.112	″
$\Delta\mu$	−2.062	±0.417	ppm

of the Earth, the geocentre. The translation components are of order of magnitude of a hundred metres, and also the scale change is significant, cf. Ollikainen (1993) page 15 and table 2 on page 13:

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{(2)} = (1 + \Delta\mu) \begin{bmatrix} 1 & \alpha_3 & -\alpha_2 \\ -\alpha_3 & 1 & \alpha_1 \\ \alpha_2 & -\alpha_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{(1)} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix},
$$

where in the case EUREF89 → ED50 the transformation parameters are according to the least-squares fit by Ollikainen, for the Finnish territory, given in table 16.1.[6]

ED50 (European Datum 1950) is a traditional European reference frame that is the basis of KKJ, the older Finnish national map grid coordinate system, and which dates from before the satellite era. As the table shows, it is strongly non-geocentric. The rather large precision numbers given in the table reflect the circumstance that co-ordinates determined over a large area by traditional geodetic techniques just aren't very precise.

For illustration, the parameters that are larger than three times their own uncertainty (standard deviation) are marked red. These are *statistically significant* at the three-sigma (3σ) level. The rotation angles α_1, α_2, and α_3 are *not* significant at this level and one could argue that they could be left out without seriously degrading the solution.

alpha αA

---

[6]For other parts of Europe, slightly different values apply, determined by fits to modern GNSS measurements within those territories.

16.5 Transformations between pairs of modern geodetic datums

Between pairs of modern geodetic datums of the satellite era — typically named ITRFyy, with the acronym ITRF standing for International Terrestrial Reference Frame, and yy for the year of realisation — there are general seven-parameter Helmert transformations of the form

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{(2)} = (1+\Delta\mu) \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{(1)} + \begin{bmatrix} 0 & -\alpha_3 & \alpha_2 \\ \alpha_3 & 0 & -\alpha_1 \\ -\alpha_2 & \alpha_1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}^{(1)} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix}.
$$

In this equation, the parameters $\Delta\mu$, α_1, α_2, α_3, ΔX, ΔY, and ΔZ are actually treated as linear functions of time:

mu μM
alpha αA

$$
P(t) = P(t_0) + (t-t_0)\dot{P},
$$

where t is the epoch for which the co-ordinates in datum (2) must be computed, and t_0 is the *epoch of tabulation* of the parameters on the web site — currently, for example, (ITRF) — of the IERS (International Earth Rotation and Reference Systems Service).

16.6 Transformations between local satellite based and global datums

Since the advent of satellite positioning, also local and national datums are usually created using this technology. The global satellite based datums, type ITRFyy, are seldom directly suitable for national or regional use. The user community expects the co-ordinates of points in the local datum to be *fixed*, i.e., not change with time. This means that, for the highest, geodetic precision, the local datum frame must be attached to the *tectonic motion* of the local plate, typically a few centimetres per year.

As an example may serve the European situation. Europe is located on the Eurasian tectonic plate. The co-ordinate reference system used (by agreement within EUREF, the IAG Subcommission for the European Reference Frame) is called ETRS, or ETRS89. ETRS stands for European Terrestrial Reference System. National datums for individual European countries, created by national GNSS measurement campaigns, are *realisations* of this system.

However, geodetic satellite positioning will give locations in the same system as in which the precise orbital elements of the GPS satellites are

given, so-called *precise ephemeris*, for example ITRF2005. Here, ITRF
stands for International Terrestrial Reference Frame, and 2005 is the
year of determination or *realisation* of this reference frame. Then, the
following transformation formula is commonly used (note the slightly
different notation):

$$
\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\text{ETRS89}} (t) = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\text{ITRF2005}} (t) + \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}_{\text{ITRF2005}}^{\text{ETRS89}} +
$$

$$
+ \begin{bmatrix} 0 & -\dot{R}_3 & \dot{R}_2 \\ \dot{R}_3 & 0 & -\dot{R}_1 \\ -\dot{R}_2 & \dot{R}_1 & 0 \end{bmatrix}_{\text{ITRF2005}}^{\text{ETRS89}} \times (t - 1989.0) \times \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\text{ITRF2005}} (t),
$$

where the dot on the R parameters means derivation with respect to
time. Note that this formula is approximate and applies only (to good
approximation) within the European area.

Values for the T and $\dot{R}$ parameters are given in Boucher and Altamimi
(2007), tables 3 and 4. They are orders of magitude smaller than in
the earlier described example of a transformation between a modern
(ETRS89) and a traditional (ED50) European datum. Both systems
discussed here, ETRS89 ja ITRF2005, are in fact geocentric on the centi-
metric level of approximation.

16.7 The Proj4 map-projection and datum software

All the major geospatial software packages contain a great many national
geospatial datums and geoid models, and for those using free software,
there is proj4 (PROJ contributors, 2021). However, it is important to use
a transformation that is *locally sufficiently precise*. This may mean the
extra work of deriving a local transformation based on a set of points
known in both local and WGS84 co-ordinate systems.

In the EPSG data base used in proj4, we have, e. g., for the Finnish
KKJ:

```
# KKJ / Finland Uniform Coordinate System
<2393> +proj=tmerc +lat_0=0 +lon_0=27 +k=1
 +x_0=3500000 +y_0=0 +ellps=intl
 +towgs84=-96.062,-82.428,-121.753,4.801,0.345,-1.376,1.496
 +units=m +no_defs <>
```

Here we see the definition of the transformation *from* local *to* WGS84:
a translation vector, a rotation matrix, and a scaling:

+towgs84=ΔX, ΔY, ΔZ, α_1'', α_2'', α_3'', $\Delta\mu^{\mathrm{ppm}}$,

to be used as (PROJ contributors):

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\mathrm{WGS84}} = \left(1 + 10^{-6} \cdot \Delta\mu^{\mathrm{ppm}}\right) \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\mathrm{local}} +$$

$$+ \frac{2\pi}{360 \cdot 60 \cdot 60} \begin{bmatrix} 0 & -\alpha_3'' & \alpha_2'' \\ \alpha_3'' & 0 & -\alpha_1'' \\ -\alpha_2'' & \alpha_1'' & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{\mathrm{local}} + \begin{bmatrix} \Delta X \\ \Delta Y \\ \Delta Z \end{bmatrix}.$$

Note that $\Delta\mu^{\mathrm{ppm}}$ is given in ppm (parts per million) and the rotations α_i'' in seconds of arc, and these need all to be converted.

The *projection* is defined as a *generic* transversal Mercator (+proj=tmerc) with a re-scaling factor of one (+k=1) (signifying a plain Gauss-Krüger), a central meridian of 27° East (+lon_0=27), a "false easting" of 500 km (+x_0=3500000, note that the "3" is the Finnish zone number which is prepended), and all on the International (+ellps=intl) or Hayford reference ellipsoid...

The equivalent definition for Adindan would be (EPSG:20137)

```
 # Adindan / UTM zone 37N
<20137> +proj=utm +zone=37 +ellps=clrk80
 +towgs84=-162,-12,206,0,0,0,0
 +units=m +no_defs  <>
```

where now the projection is UTM (no further detailed prescriptions needed, except +zone=37) and the reference ellipsoid is +ellps=clrk80.

A *warning* is in place here on the use of proj4: the software is *not* intended for transforming geodetic-quality co-ordinates. This shows, e. g., in the cavalier use of the term "WGS84" in the code, as well as the lack of time dependent transformation parameters. Geodetic-quality work requires that all point co-ordinate data, be they ground control points or in-flight exposure locations, have been carefully transformed to — or originally obtained in — exactly the same geodetic datum. If any of that co-ordinate data has been obtained by GNSS, the datum to be used will also be a modern, GNSS-based datum.

Advanced subjects

17

17.1   Airborne laser scanning (ALS)

17.1.1   *Airborne laser scan terrain height surveys*

Airborne laser scanning has in a short while become the method of choice to build detailed, precise digital elevation models (DEMs) over large areas. In airborne laser scanning, three-dimensional terrain data is directly geo-located geocentrically with the aid of a GNSS/IMU assembly, consisting of a satellite positioning receiver (GNSS) and an inertial measurement unit, together tracking geocentric location and attitude of the aircraft in real time.

The laser scanner typically is of the whiskbroom type, which changes the nature of the exterior orientation problem: exterior orientation needs to be performed for every measurement epoch, many times a second. Doing this with ground control points would be very challenging: GNSS/IMU is inevitable. It is however possible to specify *verification points* in the terrain, or even *verification surfaces*, for example Dahlqvist et al. (2011).

Generally, the on-board GNSS system measures the location of the GNSS antenna — more precisely, the *electromagnetic centre* of the antenna, which must be determined by antenna calibration. This centre may not be well defined, especially close to the large metal body of the aircraft which causes reflections of the GNSS signal ("multipath"). GNSS antennas are mounted on top of the aircraft fuselage for a good view of the sky.

The laser scanning device on the other hand is mounted on the bottom of the aircraft, for a good view of the ground. The inertial measurement unit, or IMU, which, integrated with GNSS, mainly keeps track of the *rotational motions* of the aircraft platform, can be mounted anywhere. One should however try to place all three units, laser scanner, GNSS receiver,
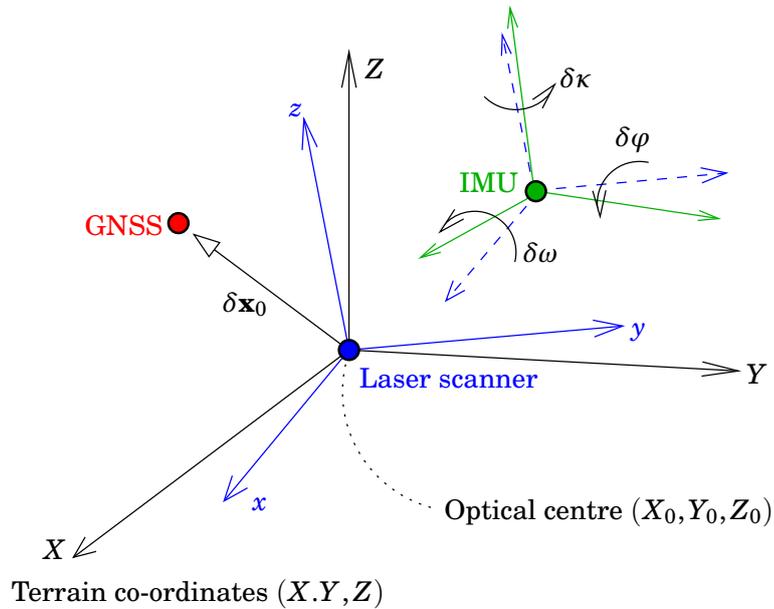
FIGURE 17.1. Geometry of airborne laser scanning. All three devices are mounted close together on the aircraft.

and IMU, as close as possible to each other, and rigidly interconnected. This is because the airframe is elastic and will undergo deformations under varying load.

17.1.2 System calibration

Now, the following *offsets* will exist:

1. A three-dimensional offset vector, here given in aircraft body co-ordinates (x, y, z):

$$\delta\mathbf{x}_0 = \left[\begin{array}{ccc} \delta x_0 & \delta y_0 & \delta z_0 \end{array}\right]^{\mathsf{T}} = \mathbf{x}_{\text{GNSS}} - \mathbf{x}_{\text{ALS}}$$

between the GNSS antenna location and the laser scanner's origin.

omega $\omega\Omega$
phi $\varphi\phi\Phi$
kappa κK

2. A rotational bias triad

$$\delta\omega = \omega_{\text{IMU}} - \omega_{\text{ALS}}, \qquad \delta\varphi = \varphi_{\text{IMU}} - \varphi_{\text{ALS}}, \qquad \delta\kappa = \kappa_{\text{IMU}} - \kappa_{\text{ALS}}$$

between the aircraft body axes orientation as maintained by the GNSS-IMU system, and the actual orientation of the laser scanning sensor. These angles are typically small.

All these six offsets could in principle be determined by careful measurement, but this has turned out to be difficult.[1] In practice, *field calibration*

[1] E. g., the aircraft airframe *deforms* between standing on the ground and being airborne, and with changes in the amount of fuel and payload on board.

is applied, where these quantities are simply added as *unknowns* to be determined. This means that, unlike the situation in traditional photogrammetry where we have one set of exterior orientation unknowns

$$(X_0)_i, (Y_0)_i, (Z_0)_i, \omega_i, \varphi_i, \kappa_i$$

for every image exposure i, here we have *one* set of six unknowns

$$\delta x_0, \delta y_0, \delta z_0, \delta \omega, \delta \varphi, \delta \kappa$$

for the whole mission.

This remark applies not only to airborne laser scan campaigns, but just as well to modern aerial photogrammetry projects where the aircraft platform is located and oriented using a GNSS/IMU combination, which is quite common nowadays. However, in this case it may be possible to *improve* the exterior orientation parameters per image exposure using good quality ground control points. It is then worthwhile keeping these individual elements in the computation.

17.2 Airborne geophysical surveys

Airborne geophysical surveys are used to study a great many quantities of interest, such as the magnetic field, gamma radiation from minerals in the underlying rock, ground penetrating radar, etc.

An interesting example is measuring *gravity* from the air, something that has become possible only because it is now possible to precisely position the aircraft using satellite positioning (GNSS). As an example, we mention the Ethiopian Airborne Gravity Survey executed in 2008 (Bedada, 2010). See figure 17.2.

Note that in the case of airborne gravity survey measurements, GNSS gives not only the location of measurement, but also *a correction to* the measurement: the motion of the aircraft causes varying disturbances to the gravity measurement by the on-board gravimeter. Using GNSS, it is possible to determine and eliminate this disturbance.

The gravimeter used in airborne gravimetry is very similar to that used in sea gravimetry: it is strongly damped in order to make it perform reasonably well in spite of the motions of the aircraft. It is mounted on a gyroscopically stabilised platform inside the aircraft, so that it measures the acceleration of free fall only along the direction of the local vertical or plumbline.
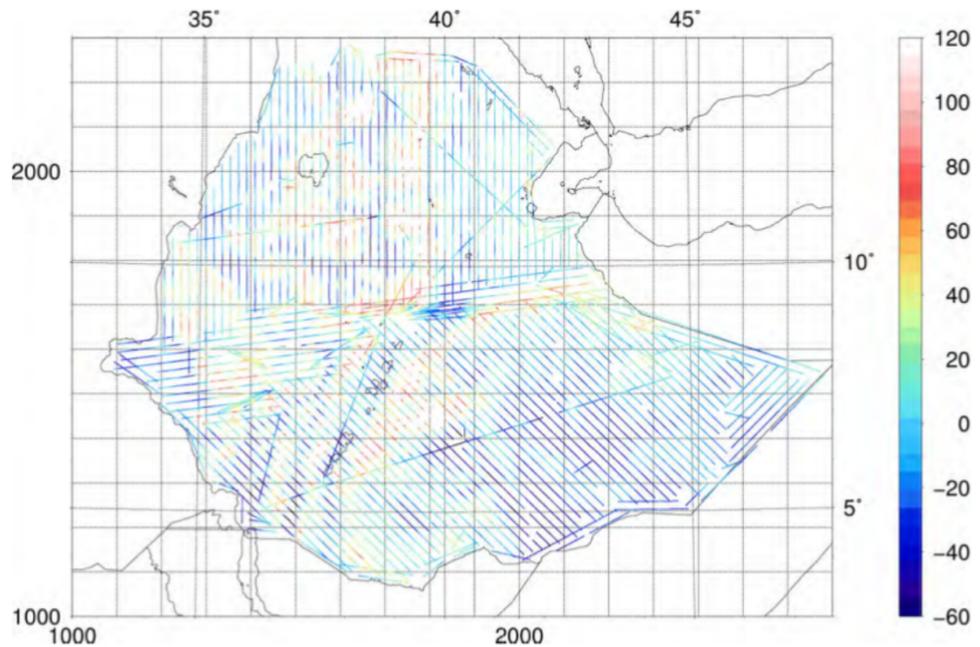
FIGURE 17.2. The tracks of the Ethiopian Airborne Gravity Survey of 2008. The gravity anomalies in mGal ($10^{-5}\,\text{m/s}^2$) are colour coded. Total flight path is $90\,000\,\text{km}$.

The precision of measurement achievable by an airborne gravimeter is of order $2-5\,\text{mGal}$: a $2-5$ millionth part of ambient gravity, or $2\cdot 10^{-5}-5\cdot 10^{-5}\,\text{m/s}^2$. The motions of the aircraft cause accelerations that may easily be $10\,000$ times greater, and these have to be eliminated. This is where GNSS comes in. If it measures at a rate of one position $\mathbf{x}_i$ per second, the spatial acceleration at time t_i can be estimated as

$$\mathbf{a}_i = 2\mathbf{x}_i - \mathbf{x}_{i-1} - \mathbf{x}_{i+1},$$

and the part of this acting in the local vertical direction $\mathbf{n}_i$ at that point of the track is simply $a_i = \langle \mathbf{n}_i \cdot \mathbf{a}_i \rangle$.

Note that the gravity values obtained, typically expressed as *gravity anomalies*,[2] are at flight height, when most users (geophysicists, geodesists) are interested in values at terrain height. Transforming the complete set of measurements from flight height to terrain height (or sea

---

[2]Gravity anomalies are obtained by subtracting from the measured acceleration of gravity, the gravity acceleration computed from a smooth, three-dimensional, global, ellipsoidal or *normal gravity field* model, for the location, and especially the height, of measurement. Gravity anomalies contain the locally rapidly varying deviations of true gravity from the smooth model gravities of the normal field, and thus extract the local features of interest from the global overall field.

level) is called (harmonic) *downward continuation*, and belongs to the domain of physical geodesy. The accuracy of the gravity anomalies at terrain level is slightly poorer than at flight level.

Rotation matrices
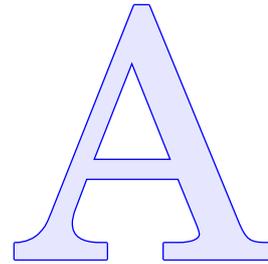
A.1 Introduction

Always when we change the orientation of the axes of a co-ordinate system, we have, written in rectangular co-ordinates, a *multiplication with a rotation matrix*.

Let us investigate the matter in two dimensions in the (x, y) plane, figure A.1.

The new x co-ordinate, after rotation by an angle α, is

$$x'_P = OU = OR\cos\alpha,$$

where

$$OR = OS + SR = x_P + PS\tan\alpha = x_P + y_P\tan\alpha.$$

By substitution we find

$$x'_P = (x_P + y_P\tan\alpha)\cos\alpha = x_P\cos\alpha + y_P\sin\alpha.$$

Similarly the new y co-ordinate is

$$y'_P = OT = OV\cos\alpha,$$
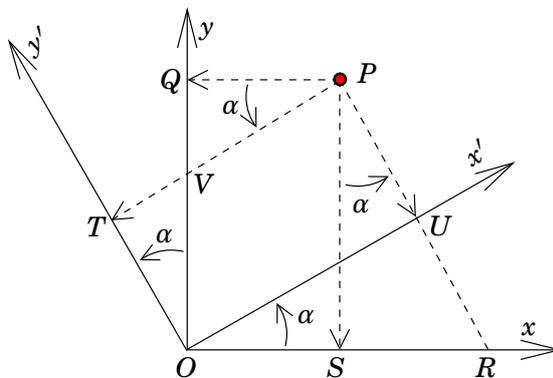
FIGURE A.1. Rotation in the plane.

– 183 –

where

$$OV = OQ - VQ = y_P - PQ \tan \alpha = y_P - x_P \tan \alpha,$$

so substitution yields

$$y'_P = (y_P - x_P \tan \alpha) \cos \alpha = -x_P \sin \alpha + y_P \cos \alpha.$$

We summarise in a matrix equation:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}.$$

The place of the minus sign in this matrix is the easiest to ascertain by making a *paper sketch*: draw both pairs of axes, mark the angle α, and infer graphically whether, for a point on the positive x axis (i.e., $y = 0$), the new y' co-ordinate is positive or negative.

In the above case

$$y' = \begin{bmatrix} -\sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} x \\ 0 \end{bmatrix} = -\sin \alpha \cdot x,$$

so $y' < 0$ for $\alpha > 0$. And so the minus sign is indeed in the lower left corner of the matrix.

A.2   Chaining matrices in three dimensions

In a three-dimensional co-ordinate system we may write a two-dimensional rotation matrix as follows:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}.$$

So the z co-ordinate is just copied, $z' = z$, while x and y transform into each other according to the above formula.

If there are several transformations in sequence, we obtain the final or compound transformation by "chaining" the transformation matrices. This means that, if

$$\mathbf{r}'' = S\mathbf{r}', \qquad\qquad \mathbf{r}' = R\mathbf{r},$$

in which

$$R = \begin{bmatrix} \cos \alpha & \sin \alpha & 0 \\ -\sin \alpha & \cos \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \beta & \sin \beta \\ 0 & -\sin \beta & \cos \beta \end{bmatrix}, \quad \mathbf{r} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \text{ etc.,}$$
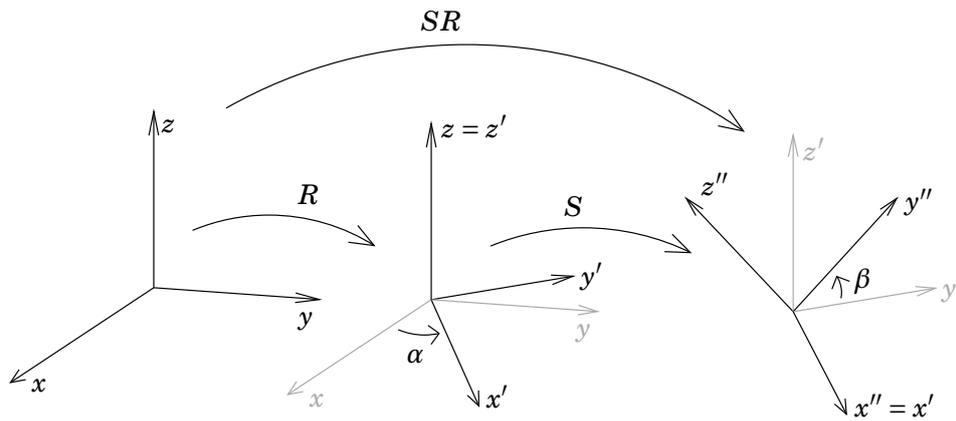
FIGURE A.2. The associativity of rotations.

then (*associativity*):

$$\mathbf{r}'' = S\left(R\mathbf{r}\right) = \left(SR\right)\mathbf{r},$$

so that the matrices themselves may be multiplied with each other to obtain the compound transformation.

Remember that

$$RS \neq SR,$$

in other words, matrices and transformations are *not* commutative![1] See figure A.2.

A.3   Orthogonal matrices

Rotation matrices are *orthogonal*:

$$RR^{\mathsf{T}} = R^{\mathsf{T}}R = I, \tag{A.1}$$

meaning that their inverse matrix is the same as their transpose.

For example:

$$\begin{bmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{bmatrix}^{-1} = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix} \left(= \begin{bmatrix} \cos\left(-\alpha\right) & \sin\left(-\alpha\right) \\ -\sin\left(-\alpha\right) & \cos\left(-\alpha\right) \end{bmatrix} \right),$$

completely understandable, because this is a rotation around the same axis, by the same amount, but *in the opposite direction*.

equation A.1 can be written in index notation:

$$\sum_{i=1}^{n} R_{ij}R_{ik} = \begin{cases} 1 & \text{if } j = k, \\ 0 & \text{if } j \neq k. \end{cases}$$

---

[1]Two-dimensional rotations *are* in fact commutative; they can be described also by complex numbers, or by simple rotation angles to be added.

We can also say that the *columns $R_{\cdot i}$* of a rotation matrix *are orthonormal*, their norm (length) is 1 and they are mutually orthogonal. This can be seen for the case of our example matrix:

$$\|R_{\cdot 1}\| = \sqrt{\sum_{i=1}^{2} R_{i1}} = \sqrt{\cos^2\alpha + (-\sin\alpha)^2} \;\; = \;\; 1,$$

$$\|R_{\cdot 2}\| = \sqrt{\sum_{i=1}^{2} R_{i2}} = \sqrt{\sin^2\alpha + \cos^2\alpha} \;\; = \;\; 1,$$

$$\langle R_{\cdot 1} \cdot R_{\cdot 2}\rangle = \sum_{i=1}^{2} R_{i1}R_{i2} = \cos\alpha \cdot \sin\alpha + (-\sin\alpha)\cdot\cos\alpha \;\; = \;\; 0.$$

Often we encounter less obvious orthogonal matrices:

1. The mirroring matrix for an axis, e. g., the y axis:

$$M_2 \doteq \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

inverting the direction, or algebraic sign, of the y co-ordinate:

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = M_2 \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x \\ -y \\ z \end{bmatrix}.$$

2. The matrix that interchanges the x and y axes (a *permutation*):

$$P_{12} \stackrel{\text{def}}{=} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix} \Longleftrightarrow \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = P_{12} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} y \\ x \\ z \end{bmatrix}.$$

3. *Inversion* of all axes:

$$X = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \Longleftrightarrow \begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = X \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} -x \\ -y \\ -z \end{bmatrix}.$$

Both M and P differ from rotation matrices in this way, that their *determinant* is -1, when for rotation matrices it is $+1$. The determinant of the X matrix is $(-1)^n$, with n the number of dimensions (in the above example, 3). A determinant of -1 means that the transformation changes a right handed co-ordinate axes frame into a left handed one, and conversely.

If we multiply, e. g., M_2 and P_{12}, we obtain

$$M_2 P_{12} = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The determinant of this is $+1$. However, it is again a rotation matrix:

$$M_2 P_{12} = R_3\left(90°\right) = \begin{bmatrix} \cos 90° & \sin 90° & 0 \\ -\sin 90° & \cos 90° & 0 \\ 0 & 0 & 1 \end{bmatrix}!$$

All orthogonal transformations having positive determinants are rotations.

(We state still without proof, that *all* orthogonal transformations in $\mathbb{R}^3$ can be written as either a rotation around a certain axis, or a mirroring through a certain plane.)

Bibliography

BCDEGHIJKLMNOPRSVW

B

Tulu Besha Bedada. *Absolute geopotential height system for Ethiopia*. PhD thesis, University of Edinburgh, 2010. URL https://era.ed.ac.uk/handle/1842/4726. Accessed 30th September, 2021. 163, 164, 179

Claude Boucher and Zuheir Altamimi. Memo : Specifications for reference frame fixing in the analysis of a EUREF GPS campaign, 2007. URL http://users.auth.gr/kvek/20070327-MEMO-ver6.pdf. 174

Jorge Luis Nunes e Silva Brito, Rafael Alves Aguiar, Marcelo Teixeira Silveira, Luiz Carlos Teixeira Coelho Filho, Irving da Silva Badolato, Paulo André Batista Pupim, Patrícia Farias Reolon, João Araujo Ribeiro, Jonas Ribeiro, Orlando Bernardo Filho, and Guilherme Lúcio Abelha Mota. E-FOTO: Development of an Open-Source Educational Digital Photogrammetric Workstation. pages 356–361, Barcelona, Espanha, October 2011. ISBN 978-1-61208-165-6. URL http://www.thinkmind.org/index.php?view=article&articleid=icsea_2011_15_10_10194. 2, 17

Duane C. Brown. Decentering distortion of lenses. *Photogrammetric Engineering*, 32(3):444–462, 1966. URL https://web.archive.org/web/20180312205006/https://www.asprs.org/wp-content/uploads/pers/1966journal/may/1966_may_444-462.pdf. 36, 52

C

B. F. Chao, Y. H. Wu, and Y. S. Li. Impact of Artificial Reservoir Water Impoundment on Global Sea Level. *Science*, 320:212–214, 2008. URL https://doi.org/10.1126/science.1154580. 150

Alexander Eugen Conrady. Decentered lens systems. *Monthly Notices of the Royal Astronomical Society*, 79:384–390, 1919. URL https://doi.org/10.1093/mnras/79.5.384. 36

D

Satu Dahlqvist, Petri Rönnholm, Panu Salo, and Martin Vermeer. Evaluating the Correctness of Airborne Laser Scanning Data Heights Using Vehicle-Based RTK and VRS GPS Observations. *Remote Sensing*, 3(9): 1902–1913, 2011. ISSN 2072-4292. doi: 10.3390/rs3091902. URL http://www.mdpi.com/2072-4292/3/9/1902. 177

E

E-foto. E-foto download site. URL https://sourceforge.net/projects/e-foto/files/stable/. Accessed 17th September, 2021. 5

E-foto home. A free GNU/GPL educational photogrammetric workstation. URL http://www.efoto.eng.uerj.br/. Accessed 18th September, 2021. 18

EPSG codes. Spatial Reference. URL https://www.spatialreference.org. Accessed 17th September, 2021. 17, 61

EPSG:20137. Adindan / UTM zome 37N. URL epsg.io/20137. Accessed 18th September, 2021. 175

G

Eric Gaba, 2007. URL https://commons.wikimedia.org/wiki/File: Tutorial_optimization_shaded_relief_map_05.jpg. Accessed 18th September, 2021. 129

Michael Gruber, Martin Ponticelli, Stefan Bernögger, Bernhard Reitinger, Hannes Hegenbarth, Zanin Cosic, and Alexander Wiechert. UltraCam Eagle, understanding the new sensor. In *ASPRS 2012 Annual Conference, Sacramento, California, March 19-23*, 2012. URL http://www.asprs.org/a/ publications/proceedings/Sacramento2012/files/Gruber.pdf. 32

Maurice S. Gyer. Methods for computing photogrammetric refraction corrections for vertical and oblique photographs. *Photogrammetric Engineering and Remote Sensing*, 62(3):301–310, March 1996. URL https://www.asprs.org/wp-content/uploads/pers/1996journal/mar/1996_mar_ 301-310.pdf. Accessed 17th September, 2021. 25

H

Zerfu Hailu and David Harris. Rural land registration in Ethiopia increased transparency for 26,000,000 land holders. In *Annual World Bank Conference on Land and Poverty*, Washington DC, 2014. URL https://www.oicrf.org/documents/40950/43224/Rural+land+registration+in+

Ethiopia+increased+transparency+for+26+000+000+land+holders.pdf/
b5496e00-9d41-a6a8-9d0e-d4bc7cbf3d5b?t=1510228917399. Accessed 18th
September, 2021. 143

I

ITRF. Transformation parameters. URL https://itrf.ensg.ign.fr/trans_para.php.
Accessed 18th September, 2021. 173

J

JUHTA. Drafting a Local Zoning Map (in Finnish). Web page, Advisory Board
for Information Management in Public Administration, JHS 185. URL
https://www.suomidigi.fi/ohjeet-ja-tuki/jhs-suositukset/
jhs-185-asemakaavan-pohjakartan-laatiminen. Accessed 18th September,
2021. vii, 148

K

Karl Kraus. *Photogrammetry: Geometry from Images and Laser Scans. Second
edition*. Walter de Gruyter Textbook. De Gruyter, 2007. 459 pages, transl. I.A.
Hartley and S. Kyle. 7, 25

L

Leica Geosystems. Leica RC30 Aerial Camera System. Data sheet, 2002. URL
http://w3.leica-geosystems.com/downloads123/zz/airborne/rc30/
documentations/RC30_product_description.pdf. Accessed 18th September,
2021. 39

Linifiniti Consulting CC. The Free Quantum GIS Training Manual, 2008,
maintained. URL https://docs.qgis.org/2.8/en/docs/training_manual/. CC BY
4.0, Accessed May 8, 2019. 17

D. G. Lowe. Object recognition from local scale-invariant features. In
International Conference on Computer Vision, pages 1150–1157, Corfu,
Greece, 1999. URL https://doi.org/10.1109/ICCV.1999.790410. Accessed 18th
September, 2021. 123

M

C. Mueller and K. Neumann. Leica DMC III calibration and geometric sensor
accuracy. *The International Archives of the Photogrammetry, Remote Sensing
and Spatial Information Sciences*, XL-3/W4:1–9, 2016. URL
https://doi.org/10.5194/isprs-archives-XL-3-W4-1-2016. 33

N

NASA Earth Observatory. The Earth-Sensing Legacy. URL
https://earthobservatory.nasa.gov/features/EO1/eo1_2.php. Accessed 18th
September, 2021. 31

O

Matti Ollikainen. Transforming GPS Co-ordinates to the National Map Grid
Co-ordinate System KKJ (in Finnish). Tiedote 8, Finnish Geodetic Institute,
Helsinki, 1993. vii, 161, 172

Optical Metrology Centre. *Camera Calibration Methods*, pages 4–1 to 4–15.
undated. URL https://www.xnatmap.org/adnm/pub/camcal/hbook4.pdf.
Accessed 18th September, 2021. 37

P

Nikolaos K. Pavlis, Simon A. Holmes, Steve C. Kenyon, and John K. Factor. The
development and evaluation of the Earth Gravitational Model 2008
(EGM2008). *Journal of Geophysical Research*, 117(B4), 2012. URL
https://doi.org/10.1029/2011JB008916. 164

PROJ contributors. General parameters. URL
http://proj.maptools.org/gen_parms.html. Accessed 18th September, 2021. 175

PROJ contributors. PROJ coordinate transformation software library, 2021.
URL https://proj.org/. Accessed 17th September, 2021. 17, 174

Public Domain Aeronautical Software. Properties Of The U.S. Standard
Atmosphere 1976, 2014. URL https://era.ed.ac.uk/handle/1842/4726.
Accessed 7th September, 2017. 25

R

Relief Shading. URL http://www.reliefshading.com/. Accessed 18th September,
2021. 129

S

G. H. Schut. Photogrammetric refraction. *Photogrammetric Engineering*, 35(1):
79–86, 1969. 25

M. T. Silveira. Considerações Técnicas sobre o Submódulo de Extração do MDE
da Versão Integrada do e-foto (versão educacional). Technical manual, State
University at Rio de Janeiro, Rio de Janeiro, 2011. 131

M. T. Silveira, R. Q. Feitosa, K. Jacobsen, and J. L. Brito. Efficient matching steps of the SIFT for constructing a dense map of conjugate points in remote sensing images, *Correspondência Eficiente de Descritores SIFT Para Construção de Mapas Densos de Pontos Homólogos em Imagens de Sensoriamento Remoto* (in Portuguese). *Boletim de Ciências Geodésicas*, 17: 130–160, 2011. URL https: //www.scielo.br/j/bcg/a/bZfXmVWGDCYtcsxYxvJ5ZyF/abstract/?lang=en. 123

SRTM. Shuttle Radar Topography Mission. URL https://www2.jpl.nasa.gov/srtm/. Accessed 18th September, 2021. 164

V

Patrick Jayver Vicente. GE 178 Lecture 7: Flight Planning. Slide show. URL https://dokumen.tips/reader/f/ge-178-lecture-7-flight-planning. Accessed 17th September, 2021. 5

W

Wikipedia, Dispersed Settlement. URL https://en.wikipedia.org/wiki/Dispersed_settlement. Accessed 18th September, 2021. 148

Wikipedia, Marquardt Algorithm. URL https://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt_algorithm. Accessed 18th September, 2021. 93

Index

ABCDEFGHIKLMNOPQRSTUVWYZ