Chapter 14 Analytic Photogrammetry

14.1 Introduction

Analytic photogrammetry includes the body of techniques by which, from measurements of one or more 2D-perspective projections of a 3D object, one can make inferences about the 3D position, orientation, and lengths of the observed 3D object parts in a world reference frame.

inference problems: can be construed as nonlinear least-squares problems

Photogrammetry provides a collection of methods for determining the position and orientation of cameras and range sensors in the scene and relating camera positions and range measurements to scene coordinates.

GIS: Geographic Information System

GPS: Global Positioning System

exterior orientation: determines position and orientation of camera in absolute coordinate system from projections of calibration points in scene

exterior orientation of camera: specified by all parameters of camera pose

pose parameters: perspectivity center position, optical axis direction...

exterior orientation specification: requires 3 rotation angles, 3 translations

interior orientation: determines internal geometry of camera
interior orientation of camera: parameters determining geometry of 3D rays from measured image coordinates
The parameters of interior orientation relate the geometry of ideal perspective projection to the physics of a camera.
parameters: camera constant, principal point, lens distortion ....
complete specification of camera orientation: interior and exterior orientation

relative orientation: determines relative position and orientation between 2 cameras from projections of calibration points in scene
relative orientation: to calibrate relation between two cameras for stereo
relative orientation: relates coordinate systems of two cameras to each other, not knowing 3D points themselves, only their projections in image
relative orientation of one camera to another constitutes stereo model: specified by 5 parameters: 3 rotation angles, 2 translations
relative orientation: assumes interior orientation of each camera known

absolute orientation: determines transformation between 2 coordinate systems or position and orientation of range sensor in absolute coordinate system from coordinates of calibration points

absolute orientation: to convert depth measurements in viewer-centered coordinates to absolute coordinate system for the scene
absolute orientation: orientation of stereo model in world reference frame
absolute orientation: determines scale, 3 translations, 3 rotations
absolute orientation: recovery of relation between two coordinate systems

$\begin{pmatrix} x \\ y \\ z \end{pmatrix}$: a point in the world reference frame
$\begin{pmatrix} x_0 \\ y_0 \\ z_0 \end{pmatrix}$: camera lens position

left-hand coordinate system: clockwise angle of rotation
ω: angle of rotation around x-axis of camera reference frame
ω: tilt angle
ϕ: angle of rotation around y-axis
ϕ: pan angle
κ: angle of rotation around z-axis
κ: swing angle

3×3 rotation matrix:

$$R(\omega) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\omega & \sin\omega \\ 0 & -\sin\omega & \cos\omega \end{pmatrix}$$

$$R(\phi) = \begin{pmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{pmatrix}$$

$$R(\kappa) = \begin{pmatrix} \cos\kappa & \sin\kappa & 0 \\ -\sin\kappa & \cos\kappa & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R(\omega, \phi, \kappa) = R(\kappa)R(\phi)R(\omega) = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} =$$

$$\begin{pmatrix} \cos\phi\cos\kappa & \sin\omega\sin\phi\cos\kappa + \cos\omega\sin\kappa & -\cos\omega\sin\phi\cos\kappa + \sin\omega\sin\kappa \\ -\cos\phi\sin\kappa & -\sin\omega\sin\phi\sin\kappa + \cos\omega\cos\kappa & \cos\omega\sin\phi\sin\kappa + \sin\omega\cos\kappa \\ \sin\phi & -\sin\omega\cos\phi & \cos\omega\cos\phi \end{pmatrix}$$

angles can be obtained directly from values r_{ij}:

$$\sin\phi = r_{31}$$

$$\tan\omega = (-r_{32})/r_{33}$$

$$\tan\kappa = (-r_{21})/r_{11}$$

point $(x, y, z)'$ in world frame represented by $(p, q, s)'$ in camera frame:
$'$: denotes matrix or vector transpose
$R^{-1} = R'$

$$\begin{pmatrix} p \\ q \\ s \end{pmatrix} = R(\omega, \phi, \kappa) \begin{pmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{pmatrix}$$

pinhole camera with image at distance f from camera lens, projection:

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{f}{s} \begin{pmatrix} p \\ q \end{pmatrix}$$

f: camera constant, related to focal length of lens

principal point: origin of measurement image plane coordinate
principal point (u_0, v_0): in image measurement plane coordinate system

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} u_0 \\ v_0 \end{pmatrix} + \frac{f}{s} \begin{pmatrix} p \\ q \end{pmatrix}$$

general fundamental perspective projection equations: collinearity equation:

$$\frac{x - x_0}{z - z_0} = \frac{r_{11}(u - u_0) + r_{21}(v - v_0) + r_{31}f}{r_{13}(u - u_0) + r_{23}(v - v_0) + r_{33}f}$$

$$\frac{y - y_0}{z - z_0} = \frac{r_{12}(u - u_0) + r_{22}(v - v_0) + r_{32}f}{r_{13}(u - u_0) + r_{23}(v - v_0) + r_{33}f}$$

They show that the relationship between the measured 2D-perspective projection coordinates and the 3D coordinates is a nonlinear function of
$u_0, v_0, x_0, y_0, z_0, \omega, \phi$, and κ.
=====Garfield 17:27=====

14.2 Nonlinear Least-Squares Solutions
noise model:

$$\alpha_k = g_k(\beta_1, ..., \beta_M) + \xi_k, \quad k = 1, ..., K$$

$\beta_1, ..., \beta_M$: unknown parameters governing nonlinear transformations
$g_1, ..., g_K$: nonlinear transformations
$\alpha_1, ..., \alpha_K$: observed values of $g_1, ..., g_k$
$\xi_1, ..., \xi_K$: additive mean zero Gaussian random variables with covariance $\not\Sigma$

maximum likelihood solution: $\beta_1, ..., \beta_M$ maximize $Prob(\alpha_1, ..., \alpha_K | \beta_1, ..., \beta_M)$

maximum likelihood solution minimizes least-squares criterion

$$\epsilon^2 = (\alpha - g)' \not\Sigma^{-1} (\alpha - g)$$

where

$$\alpha = \begin{pmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_K \end{pmatrix} \text{ and } g = \begin{pmatrix} g_1(\beta_1, ..., \beta_M) \\ g_2(\beta_1, ..., \beta_M) \\ \vdots \\ g_K(\beta_1, ..., \beta_M) \end{pmatrix}$$

assumption: overconstrained and unique solution, $K > M$

first-order Taylor series expansion of g_k taken around β^t

$$g_k(\beta^t + \Delta\beta) = g_k(\beta^t) + \Delta g_k(\Delta\beta; \beta^t)$$

$\beta^t = (\beta^t_1, ..., \beta^t_M)'$: tth iteration approximate solution
$\Delta\beta = (\Delta\beta_1, ..., \Delta\beta_M)'$: adjustments added for a better approximate solution
Δg_k: total derivative of g_k: linear function of adjustment vector $\Delta\beta$

$$\Delta g_k(\Delta\beta; \beta^t) = \left(\frac{\partial g_k}{\partial \beta_1}(\beta^t), ..., \frac{\partial g_k}{\partial \beta_M}(\beta^t) \right) \Delta\beta$$

G^t: Jacobian given by

$$G^t = \begin{pmatrix} \frac{\partial g_1}{\partial \beta_1} & \frac{\partial g_1}{\partial \beta_2} & \cdots & \frac{\partial g_1}{\partial \beta_M} \\ \frac{\partial g_2}{\partial \beta_1} & \frac{\partial g_2}{\partial \beta_2} & \cdots & \frac{\partial g_2}{\partial \beta_M} \\ \vdots & \vdots & \vdots & \vdots \\ \frac{\partial g_K}{\partial \beta_1} & \frac{\partial g_K}{\partial \beta_2} & \cdots & \frac{\partial g_K}{\partial \beta_M} \end{pmatrix}$$

adjustments added to the current β to form the next β:

$$\beta^{t+1} = \beta^t + \Delta\beta = \beta^t + (G^{t'} \not\Sigma^{-1} G^t)^{-1} G^{t'} \not\Sigma^{-1} (\alpha - g^t)$$

14.3 The Exterior Orientation Problem

one-camera exterior orientation problem:

known:

- $(x_n, y_n, z_n)'$: set of N 3D points with known positions, $n = 1, ..., N$
- $(u_n, v_n)'$: corresponding set of 2D-perspective projections, $n = 1, ..., N$

unknown:

- rotation (ω, ϕ, κ) and translation $(x_0, y_0, z_0)'$ that put the camera reference frame in the world reference frame

camera calibration problem: unknown position of camera in object frame
object pose estimation problem: unknown object position in camera frame
spatial resection problem in photogrammetries: 3D positions from 2D orientation

(p_n, q_n, s_n): point (x_n, y_n, z_n)'s coordinate in camera reference frame

$$\begin{pmatrix} p_n \\ q_n \\ s_n \end{pmatrix} = R(\omega, \phi, \kappa) \begin{pmatrix} x_n - x_0 \\ y_n - y_0 \\ z_n - z_0 \end{pmatrix}, \quad n = 1, ..., N$$

observed perspective projection

$$\begin{pmatrix} u_n \\ v_n \end{pmatrix} = \frac{f}{s_n} \begin{pmatrix} p_n \\ q_n \end{pmatrix}, \quad n = 1, ..., N$$

in the notation of the previous section, $\beta = (x_0, y_0, z_0, \omega, \phi, \kappa)$

$$g_{2n-1}(\beta) = u_n = f p_n / s_n$$

$$g_{2n}(\beta) = v_n = f q_n / s_n$$

14.3.1 Standard Solution

by chain rule for partial differentiation

$$\Delta u_n = f \frac{s_n \Delta p_n - p_n \Delta s_n}{s_n^2}$$

$$\Delta v_n = f \frac{s_n \Delta q_n - q_n \Delta s_n}{s_n^2}$$

in matrix form

$$\begin{pmatrix} \Delta u_n \\ \Delta v_n \end{pmatrix} = \frac{f}{s_n} \begin{pmatrix} 1 & 0 & -p_n/s_n \\ 0 & 1 & -q_n/s_n \end{pmatrix} \begin{pmatrix} \Delta p_n \\ \Delta q_n \\ \Delta s_n \end{pmatrix}$$

=====p. 134=====

..... (tedious mathematics, study by yourself)

14.3.2 Auxiliary Solution

Instead of iteratively adjusting the angles directly, we can reorganize the calculation such that we iteratively adjust the three auxiliary parameters of a skew symmetric matrix associated with the rotation matrix.

Then we determine the adjustments of the angles in terms of the adjustments of the parameters of the skew symmetric matrix associated with the rotation matrix.

14.3.3 Quaternion Representation

quaternion: one other representation for rotation matrices

from any skew symmetric matrix

$$S = \begin{pmatrix} 0 & -c & b \\ c & 0 & -a \\ -b & a & 0 \end{pmatrix}$$

a rotation matrix R can be constructed by choosing scalar d

$$R = (dI + S)(dI - S)^{-1}$$

this definition guarantees that $R'R = I$:

$$R'R = [(dI + S)(dI - S)^{-1}]'[(dI + S)(dI - S)^{-1}] = \cdots = I$$

expanding the equation for R, we have $R =$

$$\begin{pmatrix} d^2 + a^2 - b^2 - c^2 & 2(ab - cd) & 2(ac + bd) \\ 2(ab + cd) & d^2 - a^2 + b^2 - c^2 & 2(bc - ad) \\ 2(ac - bd) & 2(bc + ad) & d^2 - a^2 - b^2 + c^2 \end{pmatrix} \frac{1}{a^2 + b^2 + c^2 + d^2}$$

parameters a, b, c, and d can be constrained to satisfy

$$a^2 + b^2 + c^2 + d^2 = 1$$

in terms of quaternion parameters

$$\Delta R = \frac{\partial R}{\partial a}\Delta a + \frac{\partial R}{\partial b}\Delta b + \frac{\partial R}{\partial c}\Delta c + \frac{\partial R}{\partial d}\Delta d$$

so that

$$\Delta R R' = \frac{\partial R}{\partial a}R'\Delta a + \frac{\partial R}{\partial b}R'\Delta b + \frac{\partial R}{\partial c}R'\Delta c + \frac{\partial R}{\partial d}R'\Delta d$$

calculating the matrices directly, we obtain

$$\frac{\partial R}{\partial a}R' = 2\begin{pmatrix} a & b & c \\ -b & a & -d \\ -c & d & a \end{pmatrix}$$

..... (tedious mathematics, study by yourself)
=====Oldie 33:76=====

14.4 Relative Orientation

The transformation from one camera station to another can be represented by a rotation and a translation.
The relation between the coordinates, r_l and r_r of a point P can be given by means of a rotation matrix and an offset vector.
=====Horn, *Robot Vision*, Fig. 13.3=====
Relative orientation is typically concerned with the determination of the position and orientation of one photograph with respect to another, given a set of corresponding image points.
$(x_L, y_L, z_L), (x_R, y_R, z_R)$: 3D positions of left and right camera lenses
$(\omega_L, \phi_L, \kappa_L), (\omega_R, \phi_R, \kappa_R)$: rotation angles specifying exterior orientation
$\{(u_{Ln}, v_{Ln})\}_{n=1}^N$: set of N image points from left image
$\{(u_{Rn}, v_{Rn})\}_{n=1}^N$: corresponding set of N image points from right image
separation of two camera lenses $(x_R - x_L)$: as constant controlling scale
relative orientation: specified by five parameters

$$(y_R - y_L), (z_R - z_L), (\omega_R - \omega_L), (\phi_R - \phi_L), (\kappa_R - \kappa_L)$$

assumption:

- camera interior orientation known
- image positions expressed to identical scale and w.r.t. principal point

14.4.1 Standard Solution

rotation matrix associated with exterior orientation of left image

$$Q'_L = R(\omega_L, \phi_L, \kappa_L)$$

rotation matrix associated with exterior orientation of right image

$$Q'_R = R(\omega_R, \phi_R, \kappa_R)$$

f_R: distance between right image plane and right lens
f_L: distance between left image plane and left lens
from perspective collinearity equation

$$\begin{pmatrix} u_{Ln} \\ v_{Ln} \\ f_L \end{pmatrix} = \frac{1}{\lambda_{Ln}} Q'_L \begin{pmatrix} x_n - x_L \\ y_n - y_L \\ z_n - z_L \end{pmatrix} \text{ and } \begin{pmatrix} u_{Rn} \\ v_{Rn} \\ f_R \end{pmatrix} = \frac{1}{\lambda_{Rn}} Q'_R \begin{pmatrix} x_n - x_R \\ y_n - y_R \\ z_n - z_R \end{pmatrix} \text{ and}$$

Hence

$$\begin{pmatrix} x_n \\ y_n \\ z_n \end{pmatrix} = \begin{pmatrix} x_L \\ y_L \\ z_L \end{pmatrix} + \lambda_{Ln} Q_L \begin{pmatrix} u_{Ln} \\ v_{Ln} \\ f_L \end{pmatrix} = \begin{pmatrix} x_R \\ y_R \\ z_R \end{pmatrix} + \lambda_{Rn} Q_R \begin{pmatrix} u_{Rn} \\ v_{Rn} \\ f_R \end{pmatrix}$$

where $\lambda_{Ln} = \frac{s_{Ln}}{f_L}$ and $\lambda_{Rn} = \frac{s_{Rn}}{f_R}$
..... (tedious mathematics, study by yourself)

14.4.2 Quaternion Solution

Hinsken sets up the problem in a slightly different way.
Instead of determining the relative orientation of the right image with respect to the left image, he aligns a reference frame having its x-axis along the line from the left image lens to the right image lens.
The relative orientation is then determined by the angles $(\omega_R, \phi_R, \kappa_R)$, which rotate the right image into this reference frame, and the angles $(\omega_L, \phi_L, \kappa_L)$, which rotate the left image into this reference frame.

14.5 Interior Orientation

interior orientation: inner orientation: of a camera is specified by

- camera constant f: distance between image plane and camera lens
- principal point (u_p, v_p): intersection of optic axis with image plane in measurement reference frame located on image plane
- geometric distortion characteristics of the lens; assuming isotropic around the principal point

14.6 Stereo

optical axes parallel to one another and perpendicular to baseline

simple camera geometry for stereo photography

=====Horn, *Robot Vision*, Fig. 13.1=====

stereo pair taken by Viking Lander I on the surface of Mars

=====Horn, *Robot Vision*, Fig. 13.2=====

parallax: displacement in perspective projection by position translation

(x, y, z): 3D point position

(u_L, v_L): perspective projection on left image of stereo pair

(u_R, v_R): perspective projection on right image of stereo pair

b_x: baseline length in x-axis

position of the left camera:

$$\begin{pmatrix} -b_x/2 \\ 0 \\ 0 \end{pmatrix}$$

position of the right camera:

$$\begin{pmatrix} b_x/2 \\ 0 \\ 0 \end{pmatrix}$$

$v_L = v_R$, so that the y-parallax is zero

$$\begin{pmatrix} u_L \\ v_L \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x + b_x/2 \\ y \end{pmatrix} \text{ and } \begin{pmatrix} u_R \\ v_R \end{pmatrix} = \frac{f}{z} \begin{pmatrix} x - b_x/2 \\ y \end{pmatrix} \text{ and}$$

x-parallax: the difference $u_L - u_R$

$$u_L - u_R = \frac{f}{z}[x + \frac{b_x}{2} - (x - \frac{b_x}{2})] = \frac{f}{z}(b_x)$$

Hence

$$z = \frac{f b_x}{u_L - u_R}$$

=====Example 14.3=====

relation $z = f b_x/(u_L - u_R)$ close to being useless in real-world, because

- observed perspective projections are subject to measurement errors so that $v_L \neq v_R$ for corresponding points
- left and right camera frames may have slightly different orientations
- when two cameras used, almost always $f_R \neq f_L$

derive least-squares solution as in the textbook

14.7 2D-2D Absolute Orientation

The relationship between two coordinate systems is easy to find if we can measure the

coordinates of a number of points in both systems.
It takes three measurements to tie two coordinate systems together uniquely.
(a) A single measurement leaves three degrees of freedom of motion.
(b) A second measurement removes all but one degree of freedom.
(c) third measurement rigidly attaches two coordinate systems to each other.
=====Horn, *Robot Vision*, Fig. 13.4=====
2D-2D absolute orientation problem: 2D-2D pose detection problem:
to determine from matched points more precise estimate of rotation matrix R and translation t such that $y_n = Rx_n + t, n = 1, ..., N$:
determine R and t that minimize weighted sum of residual errors:

$$\epsilon^2 = \sum_{n=1}^{N} w_n ||y_n - (Rx_n + t)||^2$$

weights $w_n, n = 1, ..., N$: satisfy $w_n \geq 0$ and $\Sigma_{n=1}^{N} w_n = 1$
if no prior knowledge: can be defined to be equal, $w_n = 1/N$
..... (tedious mathematics, study by yourself)

14.8 3D-3D Absolute Orientation

$p_1, ..., p_N$: N 3D points relative to camera reference frame
$q_1, ..., q_N$: N 3D points relative to object model reference frame

we must determine rotation matrix R and translation vector t satisfying

$$p_n = Rq_n + t, \quad n = 1, ..., N$$

constrained least-squares problem to minimize

$$\sum_{n=1}^{N} w_n ||p_n - (Rq_n + t)||^2$$

The least-squares problem can be modeled by a mechanical system in which corresponding points in the two coordinate systems are attached to each other by means of springs.
The solution to the least-squares problem corresponds to the equilibrium position of the system, which minimizes the energy stored in the springs.
=====Horn, *Robot Vision*, Fig. 13.5=====
Lagrange multipliers can be introduced and a solution found
..... (tedious mathematics, study by yourself)
=====joke=====

14.9 Robust M-Estimation

least-squares techniques are ideal when random data perturbations
or measurement errors are Gaussian distributed
this section describes some robust techniques for nonlinear regression

14.9.1 Modified Residual Method

r_i: residual of i-th datum
r_i: difference between i-th observation and its fitted value
standard least-squares method: tries to minimize $\Sigma_i r_i^2$
standard least-squares method: unstable if outliears present in data
outlying data: strong effect to distort estmated parameters
M-estimators: try to reduce outlier effect by replacing squared residuals
M-estimators: min $\Sigma_i \rho(r_i)$
ρ: symmetric, positive-definite function with unique minimum at zero
ρ: chosen to be less increasing than square
=====p. 168=====
=====p. 169=====

14.9.2 Modified Weights Method

14.9.3 Experimental Results

14.10 Error Propagation

$x_1, ..., x_N$: input parameters
$\Delta x_1, ..., \Delta x_N$: random errors

quantity y depends on input parameters through known function f:

$$y = f(x_1, ..., x_N)$$

quantities $x_1 + \Delta x_1, ..., x_N + \Delta x_N$ observed and quantity $y + \Delta y$ calculated:

$$y + \Delta y = f(x_1 + \Delta x_1, ..., x_N + \Delta x_N)$$

error propagation analysis: determines expected value and variance of $y + \Delta y$
known information about $\Delta x_1, ..., \Delta x_N$: mean, variance:

$$E\left[\begin{pmatrix} \Delta x_1 \\ \vdots \\ \Delta x_N \end{pmatrix}\right] = 0$$

$$E\left[\begin{pmatrix} \Delta x_1 \\ \vdots \\ \Delta x_N \end{pmatrix} (\Delta x_1, ..., \Delta x_N)\right] = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ \sigma_{N1} & \sigma_{N2} & \cdots & \sigma_{NN} \end{pmatrix}$$

14.10.1 Implicit Form

error propagation: can also be done in implicit form
known function f has the form

$$f(x_1, x_2, ..., x_N, y) = 0$$

The quantities $(x_1 + \Delta x_1, ..., x_N + \Delta x_N)$ are observed, and the quantity $y + \Delta y$ is determined to satisfy $f(x_1 + \Delta x_1, ..., x_N + \Delta x_N, y + \Delta y) = 0$.

14.10.2 Implicit Form: General Case

general case: y is not a scalar but a $L \times 1$ vector β
$x_1, ..., x_K$: are K $N \times 1$ vectors representing true values
$x_1 + \Delta x_1, ..., x_K + \Delta x_K$: are K $N \times 1$ vectors representing noisy observed values
$\Delta x_1, ..., \Delta x_K$: random perturbations
β: is a $L \times 1$ vector representing unknown true parameters
noiseless model:

$$f(x_k, \beta) = 0, \quad k = 1, ..., K$$

with noisy observations, the idealized model:

$$f(x_k + \Delta x_k, \beta + \Delta \beta) = 0, \quad k = 1, ..., K$$

14.11 Summary

we have shown how to

- take a nonlinear least-squares problem
- linearize it
- solve by iteratively solving successive linearized least-squares problems

=====joke=====

Project due April 19:
camera calibration: interior orientation:
to determine principal point
do camera motion in z-axis, find focus of expansion/contraction
use lens of focal length: 16mm, 25mm, 50mm