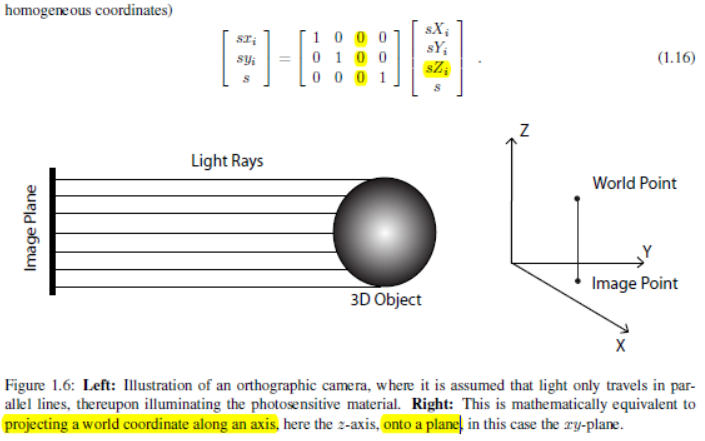
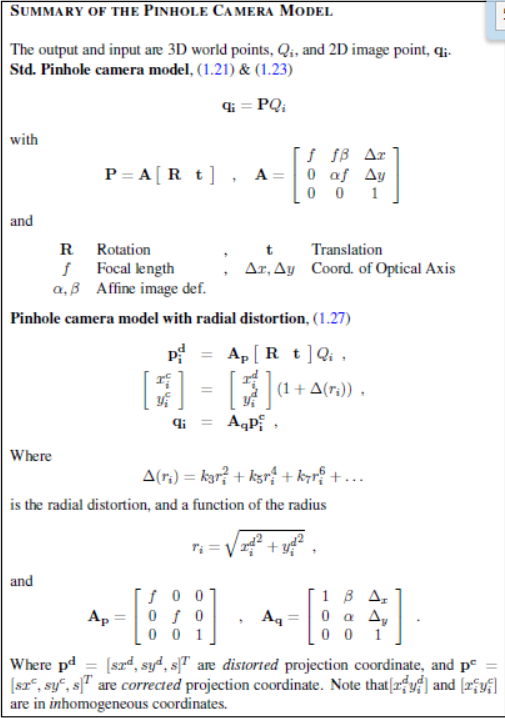
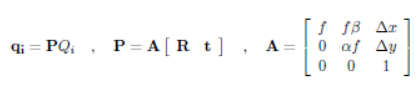
Part I

## 1.3 The Orthographic Projection Model





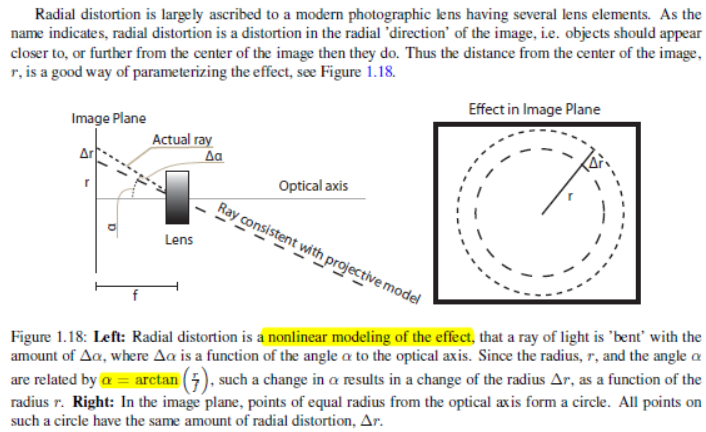


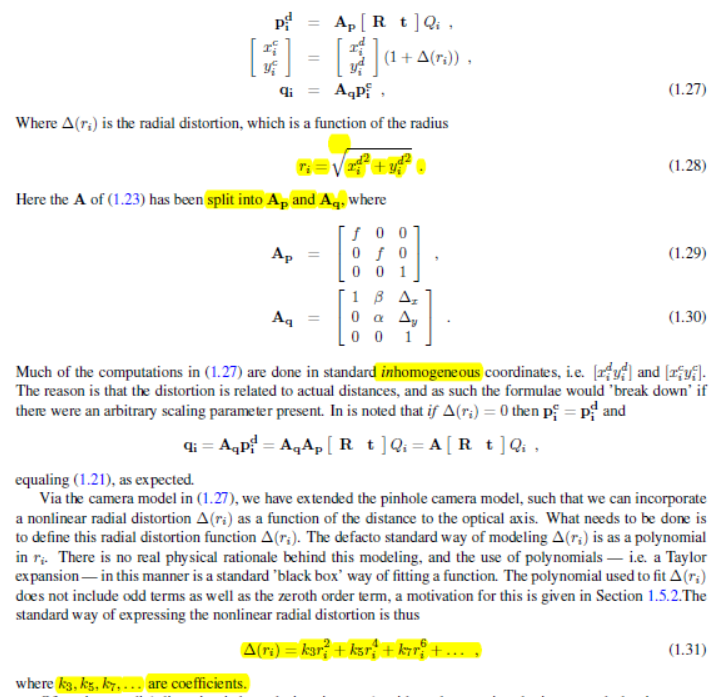
If this is not clear please derive this, by

setting dx = 0; dy = 0; B= 0, a = 1.and assuming that the camera coordinate system and the global

coordinate system are identical – i.e. R = I and t = 0.

1.5 Radial Distortion - Refined Pinhole Model





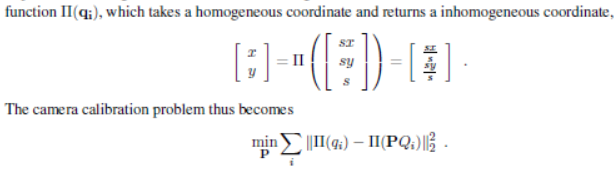
## 1.6 Camera Calibration

How to obtain the parameters of such models, given a camera. This is also known as camera calibration.

Two ways:

Most advanced: will do it automatically from an image sequence

Most standard way: taking images of known 3D objects, typically in the form of known 3D points Qi.



2.1.1 Back Projection of an Image Line

Plane, L

Line , l

# 

image plane

# lTqi = 0

# 2.2 Epipolar Geometry

Two cameras are viewing the same 3D point, Q, with unknown coordinates. Assume also that the coordinate system of the first camera is equal to the global coordinate system.

Image plane 2

Image plane 1

Rp1 +t n

t

q2 is constrained on the red dash that is *epipolar line.*

Centers of cameras one and two and p1 lie on a plane. This is the *epipolar plane. Green plane*

*n =* *[t] x Rp1*

Any point, p, on this plane *pT n = 0 = pT* *[t] x R p1.*

*E = [t] x R. when A1 = A2 = I, E = F.*

*F=𝐴2−𝑇 𝐸𝐴1−1*

*l2 = Fq1* *qT2**Fq1 = 0. Fq1* describes an [epipolar line](http://en.wikipedia.org/wiki/Epipolar_line" \o "Epipolar line) on which the corresponding point *qT2* on the other image must lie.

Epipolar Planes

Image plane 2

Epipoles dash: epipole lines

All the epipolar lines will intersect in a common point, the projection of these camera centers are called the *Epipoles.* Projection of camera center one into image two,

*e2 = A2 t*

2.3 Homographies for Two View Geometry

Two view geometry in the cases where a planar surface is viewed and/or there is no motion between the views (i.e. t = 0), latter case [t]\_ = 0, and E, F are 0. When epipolar geometry fails,

A (full rank) 3 by 3 matrix, H, that maps between 2D homogeneous coordinates q1 and q2

q1 = Hq2

Any point Q in that plane C can be described as:

*Q = aA + bB + C a point [a b 1]T*

H = P [ A B C; 0 0 1], a plane C and two vectors AB on that plane.

Two Cameras Viewing a Plane

If two cameras are viewing a plane, then the relationship between the two images taken is also a homography.

we have a pair of corresponding points q1 and q2 from the two images respectively and that they are depictions of Q. Then two homographies, H1 and H2, exist such that

q1 = H1qp q2 = H2qp qp = H-12 q2.

q1 = H1qp = H1 H-12 q2

H = H1 H-12

q1 = Hq2

# 2.3.4 Two View Geometry Without a Baseline

Baseline is the distance between two camera centers. Without baseline (t=0), makes E,F model useless. So H can be used.

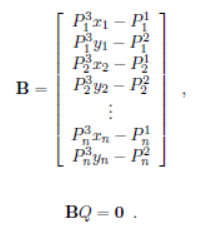
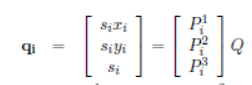
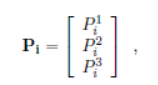
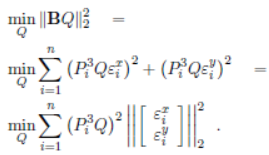
q2 = A2 R A-11 q1

H = A2 R A-11

Point Triangulation - P, q, 🡪Q

Finding 3D point Q from it’s 2D points qi, i = [1; : : : ; n], in n known cameras, Pi.

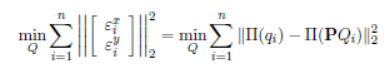
# 2.4.1 A Linear Algorithm

# 

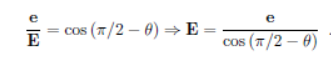
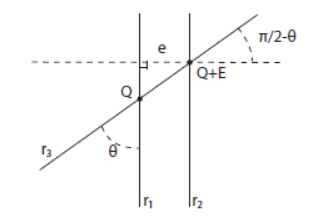
We assume 2d point x,y noiseless, Least square reduce the noise. B should be well formed meaning points span 3D space well, all the points are not located on or near a plane.

# Non-linear Minimization



function II takes homogeneous coordinates and produces the inhomogeneous correspondent. Minimizing the squared Euclidian distance of obs q and modeled q.

Camera Measurement Setup

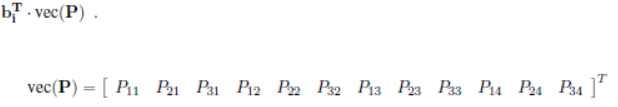
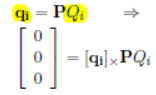


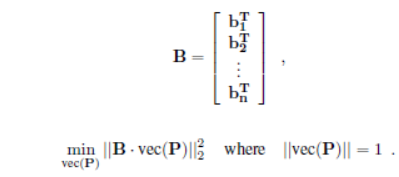
Distance between r1 and r2 in the vicinity of Q by e,

2.7 Camera Resection Q q 🡪 P 6 points needed

Estimate camera from known 3D points and the corresponding 2D projections.

Linear algorithm





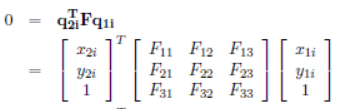
Avoid trivial solution, B is 3 x 12 but only having 2 ranks, P has 11 df, so 6 2D points needed at least.(2\*6=12) The 3D points used are not located on or near a plane.

# 2.8 Estimating the Epipolar Geometry q1,q2🡪 F (7df)

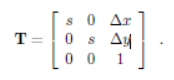
## The 8-Point Algorithm

Fundamental matrix can be used as a constraint on point correspondences, it can also be estimated

from such point correspondences.

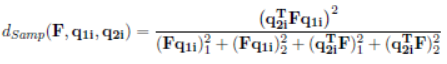


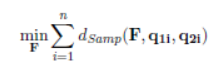
2.8.2 Normalization of Points



2.8.3 Non-Linear Optimization

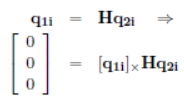
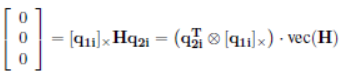
Sampson distance:





2.9 Estimating a Homography q1,q2🡪 H 4 points needed

Linear algorithm

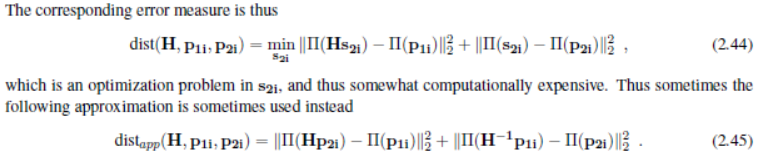
 

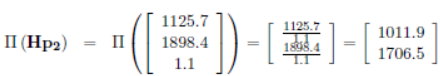
Minimum number of point correspondences needed to estimate a homography is 4.

B (3x9) has max rank two, H has 9 parameters, subtract one for scaling, we need 8 constraints.

8/2=4.

Non-Linear Optimization





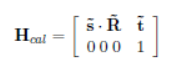
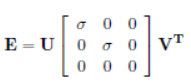
Uncalibrated pinhole camera model has 12 df, calibrated has 6 df. 3 for R (highly non-linear,) and 3 for t. calibrated is used in practice, as noise reduction.

For any full rank 4 x 4 matrix, H, the uncalibrated pinhole camera can be manipulated:



A solution can only be an arbitrary transformation of a full rank 4 \_ 4 matrix.

Calibrated case, to keep P and P tilt (3x4) consistent with A, H is constrained to be:

H is an arbitrary definition of global coordinate system.

# 3.2 Estimating the Essential Matrix 5 point needed (5df)

## 5 point algorithms

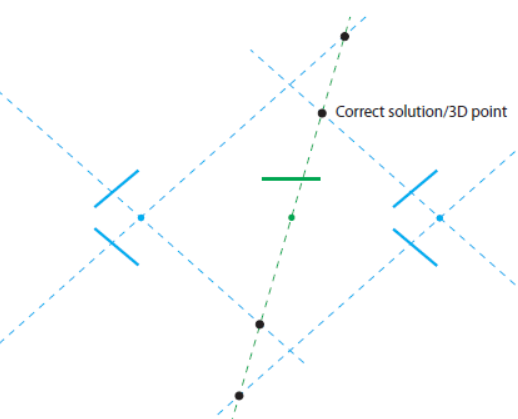
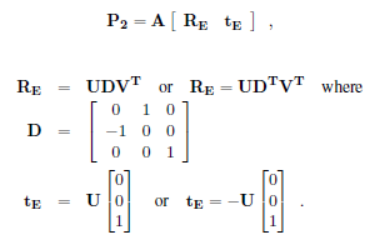
*B vec (E) = 0*

*E = xX + yY + zZ + wW 4 dimensional solution🡪 4 possible camera configurations*

 *gives 10 degree polynomial*

Given an essential matrix, the rotation and translation (direction not length) between the two cameras can be derived.

E, P1, 🡪 P2 [Re te]

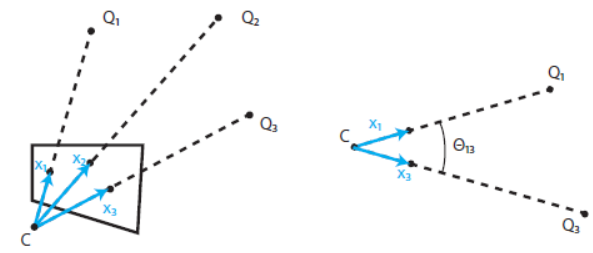
 

Using point triangulation to estimate 3D points for 4 cam conf.

Determine if the point is in front of cam by transforming Q into cam coor system. Check if z is positive.



3.3 Calibrated Camera Resectioning 3points needed.



Constrain C to a specific circle in any plane through Qi and Qj .

3.4 Bundle Adjustment non-linear optimization

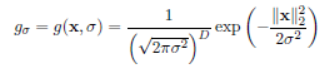


Part II

# Extracting Features

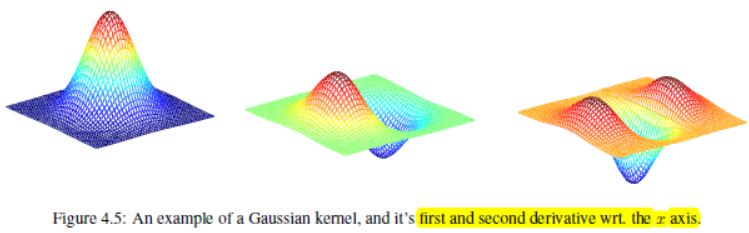
Scale: Large filter gives dominant edge.

4.1.1 Gaussian Kernels



sigma is the measure of scale. Without altering the size of the filter itself.

When doing feature extraction, a very commonly used filter, f, is a derivative of some order.

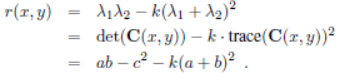


4.2 The Aperture Problem: location of a feature can only be uniquely determined across an image gradient. Extraction algorithms build on gradients, or derivatives of some order

4.3 Harris Corner Detector



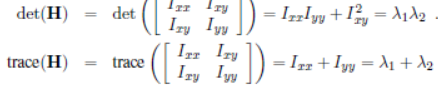
In a sense, it is a weighted variance matrix of the pairwise pixel differences. If we have a corner, with a large gradient in all directions, the rate of change should be large in all directions, implying that (4.4) should be large for ∆x; ∆y pointing in any direction.

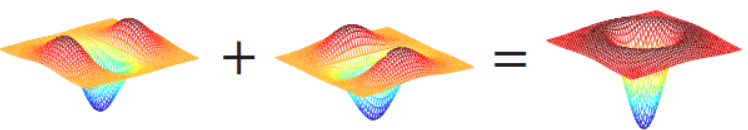
Corners are found at places where both eigenvalues are large. Finding the threshold is not enough, since images are often smooth, several pixels will be marked as corners. Applying Non-Maximum Suppression to find the harris corner (checking neighbors pixels to make sure the center is >= neighbors).

4.4 Blob Detection has a high degree of change in all directions.

For 2D image the second order derivative or hessian, H, is a 2 \_ 2 matrix. How to measure the size, such that a blob is found in presence of a large Hessian. Two ways:



the trace of the hessian is also called the Laplacian:



4.4.1 Difference of Gaussian (DoG) & the SIFT Detector

4.5 Canny Edge Detector

Big sigma gives dominant edge, but threshold has priority.

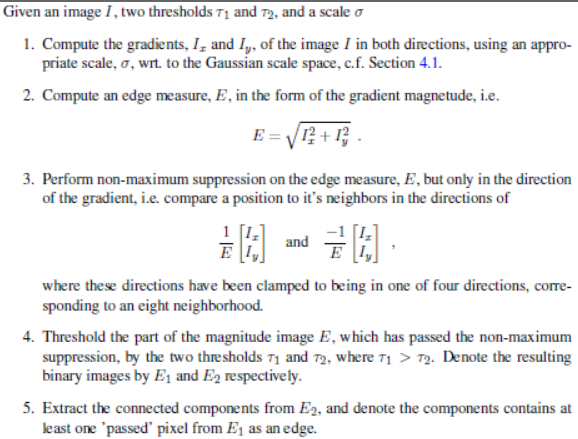
Edge measured uses gradient magnitude, Non-Maximum Suppression: an edge point is most perpendicular to the edge, used for comparison.

2 threshold

Gradient magnitude >T1 labelled as edge, as long as at least one pass NMS

T1 < gradient magnitude < T2 labelled as edge, only if they are part of a line where part of it is above the T1 threshold. (Canny)

Assumption that all edge pixels pass the non-maximum suppression criteria.

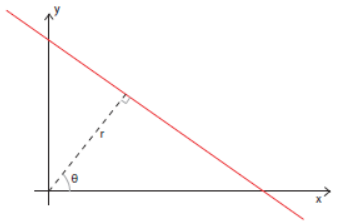
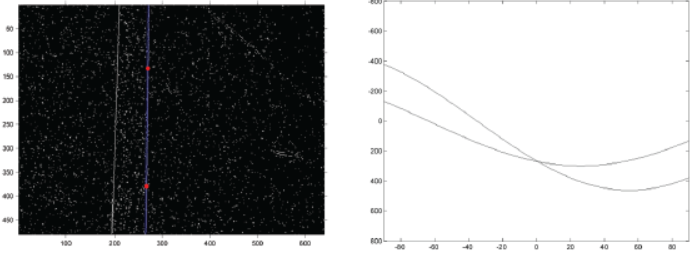


## Robust Model Fitting

Considering the features as parameterized models that should be fitted to the

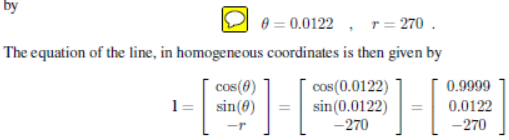
image content.

The Hough transform works by parameterizing all possible lines by (theta; r), where r denotes the minimum distance of the line to the origo, theta is the angle of the shortest distance to the line relative to the x-axis, and [cos(theta); sin(theta)]’denotes the normal of the line.

Tow red dots induced two curves in Hough space, each curve is the parameterization of all lines that go through the red dot. Intersection (Peak) of the curves corresponding to the line defined by the two red dots.

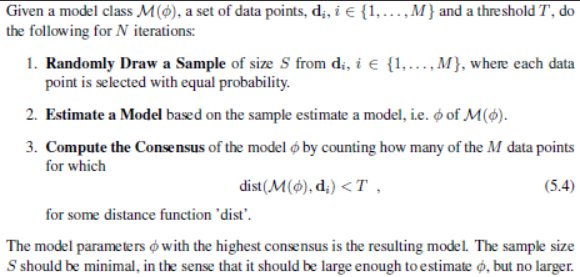




# 5.3 Random Sampling Consensus – Ransac

A modified version of the Hough transform, only a sparse set of the Hough space is computed.

The aim is then to fit a model to the inliers and leave the outliers out of the estimation.

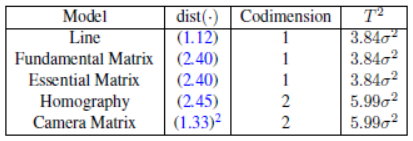


Explicitly, the data point classification into inliers and outlier is determined by the model, and the model is given by the data points defined as inliers. Points are randomly proposed in Hough space to compute.

Sample size: line S equals two 2D points, for a circle S equals three 2D points and for a plane in 3D S equals three 3D points.

The consensus is computed for the model, which is finding the place in Hough space (Phi), the score. Do this N interations, Phi with hightest consensus is chosen.

Threshold with 95% confident level :



A point is inlier if:



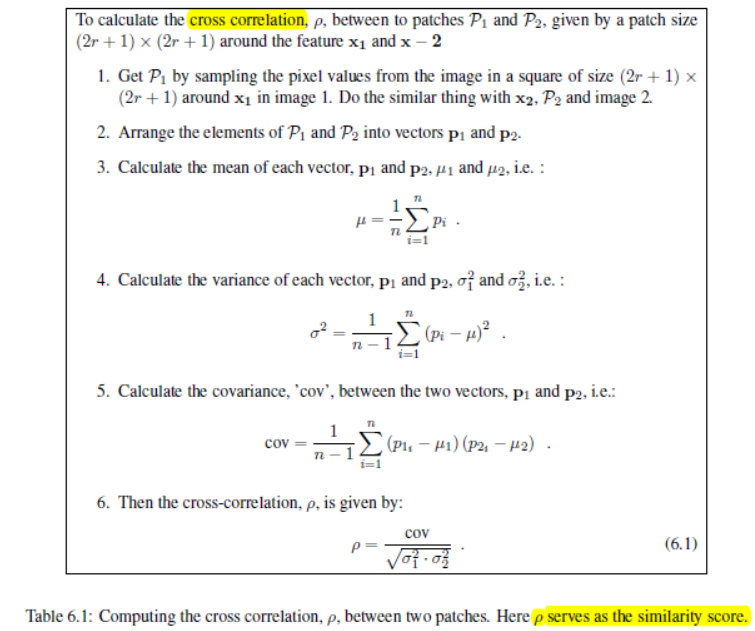
Chapter 6

Feature Based Image Correspondence

**Sampson**(F, m1, m2) valuate the first order approximation of the geometric error of the fit of F with respect to a set of matched points m1, m2.

Matching features by finding the **Features** with the most similar neighborhoods.

Using **correlation** can be interpreted as seeing one **patch (neighbors)** as a kernel and convolving the other image with it.



6.2.2 SIFT Descriptors

One of the best general-purpose feature descriptors is the SIFT descriptor.

