3D Data Acquisition

Jorge A. Silva
DEI – FEUP

3D Imaging

- Summary
  - Overview of 3D data acquisition and representation
  - 3D external data acquisition techniques
    - how do humans see in 3D ?
    - overview of computer-based techniques
  - Passive stereo method
    - principle
    - geometric camera model & camera calibration
    - image acquisition & matching (correspondence problem)
    - 3D reconstruction (triangulation)
  - Structured-light (/ active stereo) methods
    - principle
    - application example

Types of 3D data

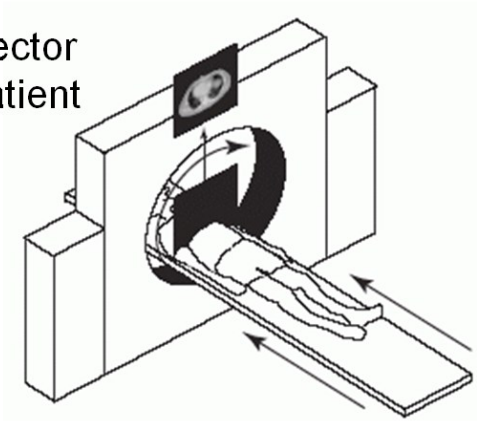
- External data
  - external object surfaces
  - several acquisition techniques
  - sometimes, partial views (only one object side)

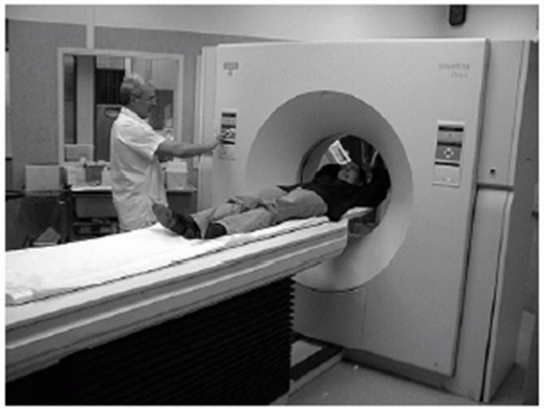
- Internal data (NOT THIS TALK)
  - “objects”
    - human body
    - industrial parts
    - earth (!)
    - …
  - acquisition techniques
    - CT – Computed Tomography
    - MRI – Magnetic Ressonance Imaging
    - PET – Positron Emission Tomography

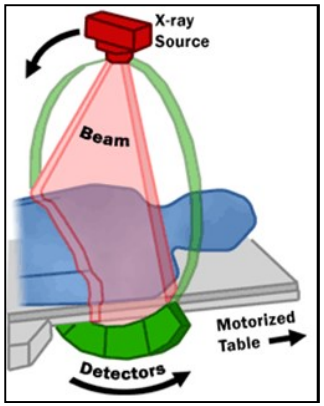
Internal 3D data acquisition

- X-ray Computed Tomography (CT)

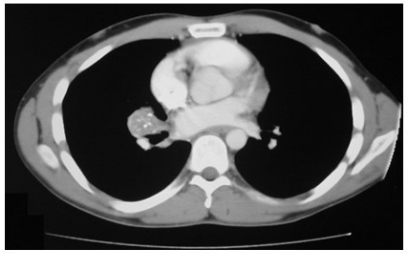
Scanning:
rotate source-detector
pair around the patient







data



reconstructed cross-
sectional slice

reconstruction routine

sinogram: a line for
every angle

Internal 3D data acquisition

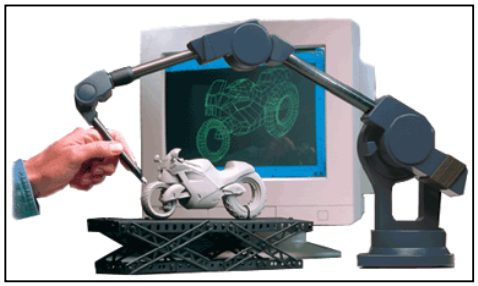
- CT scans: from X-Ray images to 3D Models

External 3D data acquisition

- Extraction of 3D information from an image (sequence) is important for
  - vision in general (= scene reconstruction)
  - many tasks (e.g. robot grasping and navigation,…)
  - not all tasks (e.g. image retrieval, …)
- Basic problem:
  - Projection from 3D to 2D causes loss of depth information
- Recovery of 3D information from 2D images is possible
  - by multiple cameras (e.g. binocular stereo, trinocular stereo, …)
  - by a single image + + some assumptions or prior knowledge about the scene
  - by a monocular image sequence with motion + + some assumptions

3D data acquisition techniques

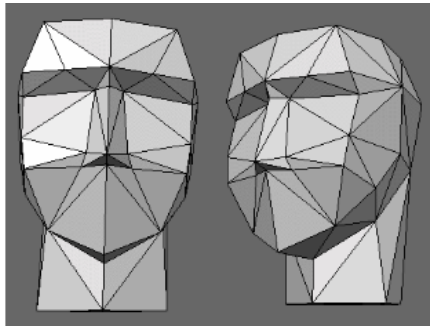
- contact techniques

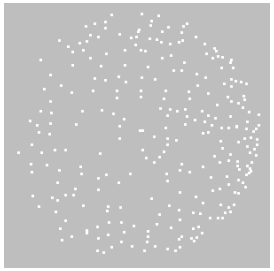


- non-contact techniques
  - optical
    (based on intensity image acquisition)
    (THIS TALK)
  - time-of-flight
    - light (LIDAR - LIght raDAR)
    - ultrasound

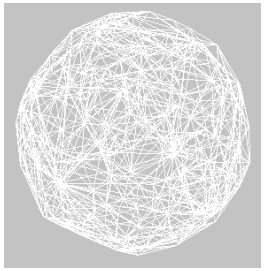
3D data representation

- Range image (/map) / Depth image (/map) / Digital Elevation Map / Digital Surface Map/ 2.5 image
  - 2D matrix
  - each pixel represents a depth value
- List of (x,y,z) coordinates
  - unorganized list
  - organized lists of points
    - many formats
      - STL - Stereolitography (triangles & normals)
      - DXF - Autocad
      - VRML - Virtual Reality Modeling Language
      - …
- Processing of raw acquired data is often needed

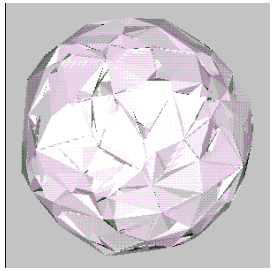




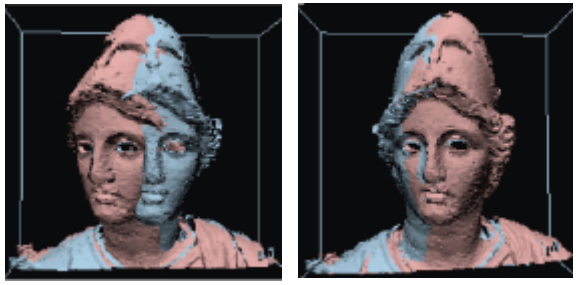
point cloud



Delaunay triangulation



surface representation



mesh alignment

How do humans see in 3D ?

- Many depth / orientation cues …
  - occlusion
  - texture gradient
  - perspective distortion
  - shading (illumination gradient)
  - shadow
  - …
  - stereo disparity

- … but no accurate measure capabilities

Depth cues

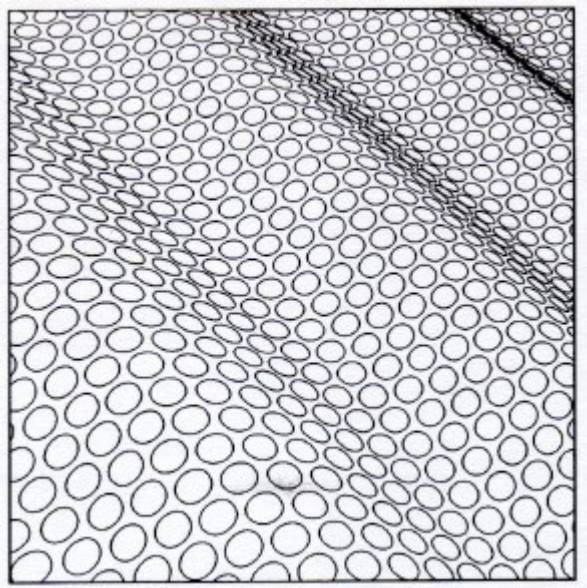
Occlusion & Perspective

Texture gradient

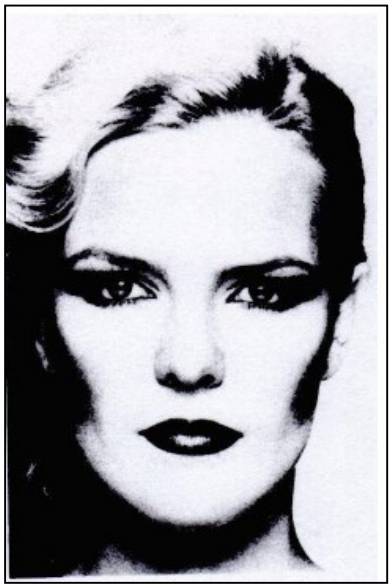
Shape cues

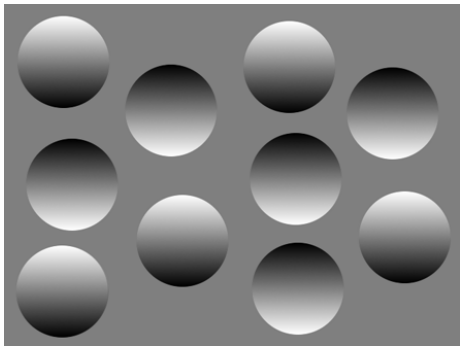
- Shape-from-X

Texture



Shading





Disparity

- Random Dot Stereograms (RDS)
  - pairs of images of random dots which when viewed with the aid of a stereoscope, or with the eyes focused on a point behind the images, produce a sensation of depth, with objects appearing to be in front of or behind the actual images
  - B. Julesz, 1971, theory on the basis of human stereo vision

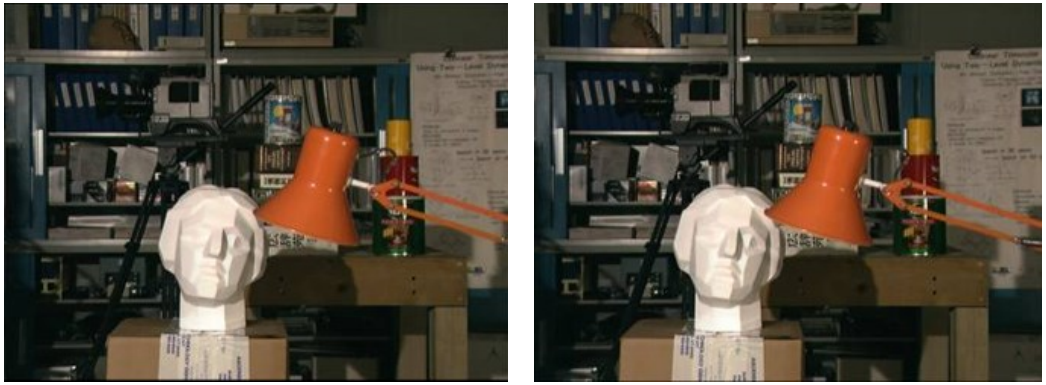
- Process used to develop the first RDS
  - 1. Create an image of suitable size.
    Fill it with random dots.
    Duplicate the image.
  - 2. Select a region in one image.
  - 3. Shift this region horizontally
    by a small amount.
    The stereogram is complete.

Random dot stereogram from Julesz, 1971.

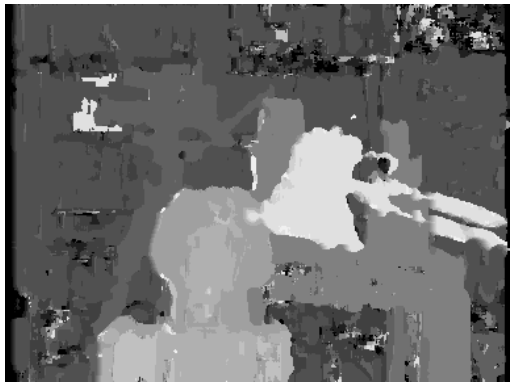
- The shifted region produces the binocular disparity necessary to give a sensation of depth. Different shifts correspond to different depths.
- **Main conclusion:**
  - **depth can be perceived in the absence of any identifiable objects**

Disparity

- Disparity image from stereo pair



Tsukuba stereo pair



Calculated disparity image

Each point
represents the disparity
(difference in image location)
between corresponding points
in the two images of the stereo pair

How do humans see in 3D ?

- … all the previous cues

  +

- lots of acquired knowledge during one's lifetime

Relative size

How do humans see in 3D ?

- There have been many human biology/psychology experiments done in depth perception
  in which people are asked which object is front or back in scene.

- We are able to infer relative depth of some objects to others
  but our judgement about true depth is usually off.
  This means, we have poor abilities for stereo reconstruction.

- Human perception is okay for relative depth
  but is poor for real depth.

Overview of techniques for 3D data acquisition

- **Direct**
  - Triangulation
    - passive (stereo) – ambient light
      - 2 images
      - multiple images (static / moving cameras)
    - active (structured-light *or* active stereo) – special projected light patterns
      - serially (beam scanning)
      - in parallel (pattern projection)
  - Time-of-flight
    - laser light
      - pulse time delay
      - amplitude modulated phase-shift
      - frequency modulated beat
    - ultrasound
- **Indirect**
  - passive – ambient light
    - range
      - depth-from-focus
      - depth-from-known-geometry
    - surface orientation
      - shape-from-perspective
      - shape-from texture
  - point-light sources
    - range
      - shadows
      - Moiré fringe patterns
    - surface orientation
      - shape-from-shading

THIS TALK

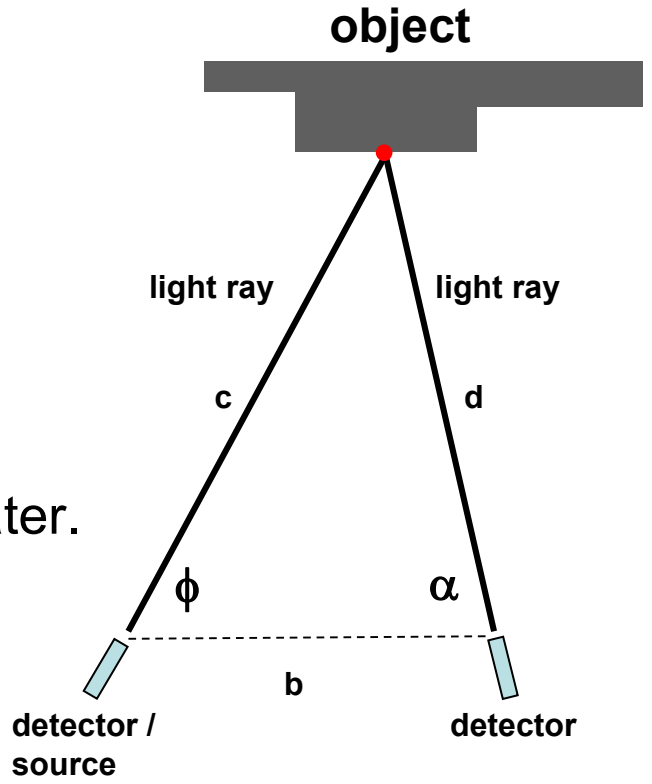
most of these only give relative depth or orientation
(except depth-from-focus)

Passive vs. Active sensors

- Passive sensors
  - Use only ambient light to illuminate the scene
  - Use 2D ordinary cameras to acquire images (1 or more)
- Active sensors
  - Use a light source such as a laser beam to aid in obtaining 3D data
  - They project light onto the scene and get depth by seeing projected light in the environment
  - Useful when the scene has no features
  - Easier to obtain dense 3D data
  - Can work in many kind of lighting conditions
  - Not always eye-safe
  - Many possibilities, but two common principles
    - triangulation (camera + projector) and time-of-flight

Triangulation

- Basic principle:
  If you know ϕ, α and **b**,
  you can compute **c** and **d.**
- Principle used in navigation systems.
  It has been used since ancient times.
- If you increase baseline **b**,
  you get more accurate results
  (errors in ϕ and α cause less harm) …
- …but the sensor is larger
  and partial occlusion problems are greater.
- Typical range:
  - short (~10 cm) to medium (~5 meters)
- Typical accuracy:
  - from 50 microns to 2 millimeters

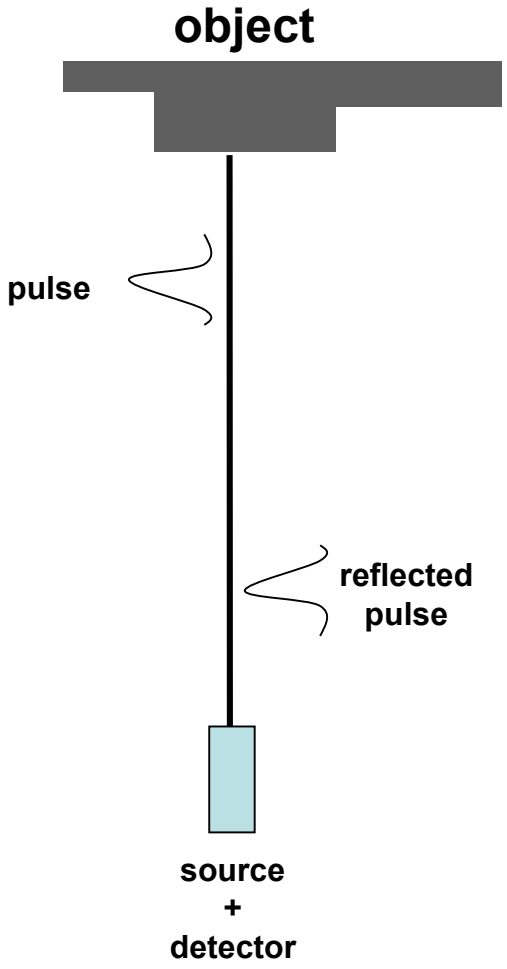


detector + detector → passive triangulation
source + detector → active triangulation

source = light / laser projector
detector = laser detector / camera

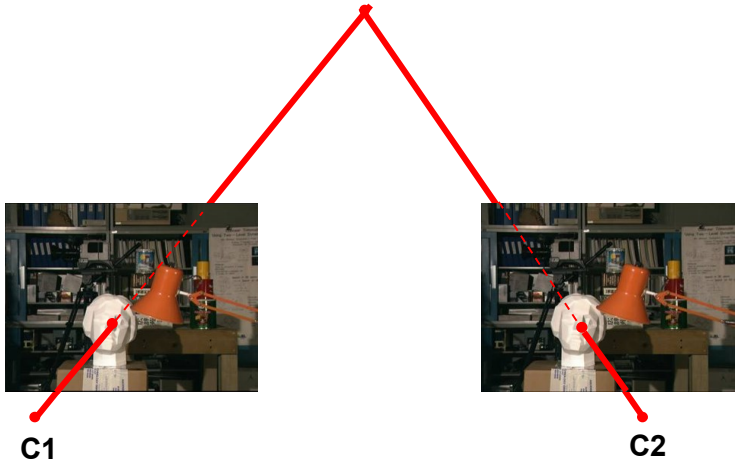
Time-of-flight

- Basic principle:
  send light (usually laser), and
  measure the time it takes to return.
- Same principle as used in
  ultrasound measurement
- Typical range:
  - medium (5 to 50 m) to
    long (50 m to some km, 5-20km) distances.
- Typical accuracy:
  - Better than passive stereo
    lower than active stereo
- For both triangulation and time-of-flight the
  required laser power
  increases with the distance to the object.
  - The longer the distance,
    the more dangerous the laser.



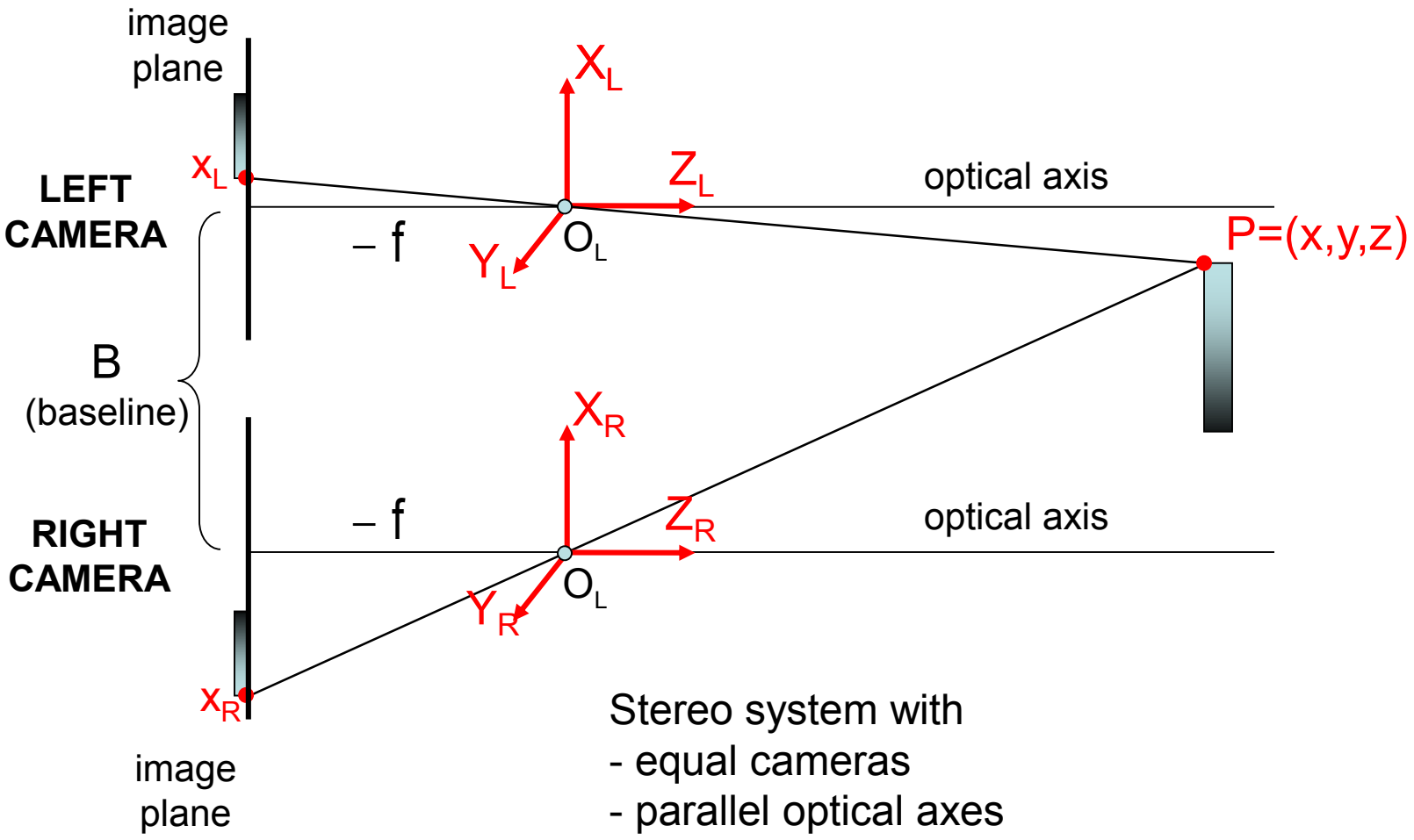
Stereopsis / Stereo vision

- Basic principle
  - triangulation
- Main steps
  - acquire 2 images from different viewpoints
  - match points between the images
  - for each point of a matching pair determine its line-of-sight ($\Rightarrow$ camera calibration)
  - intersect the 2 lines-of-sight
- Main problems
  - scenes without features
  - feature matching
  - occlusion vs. accuracy



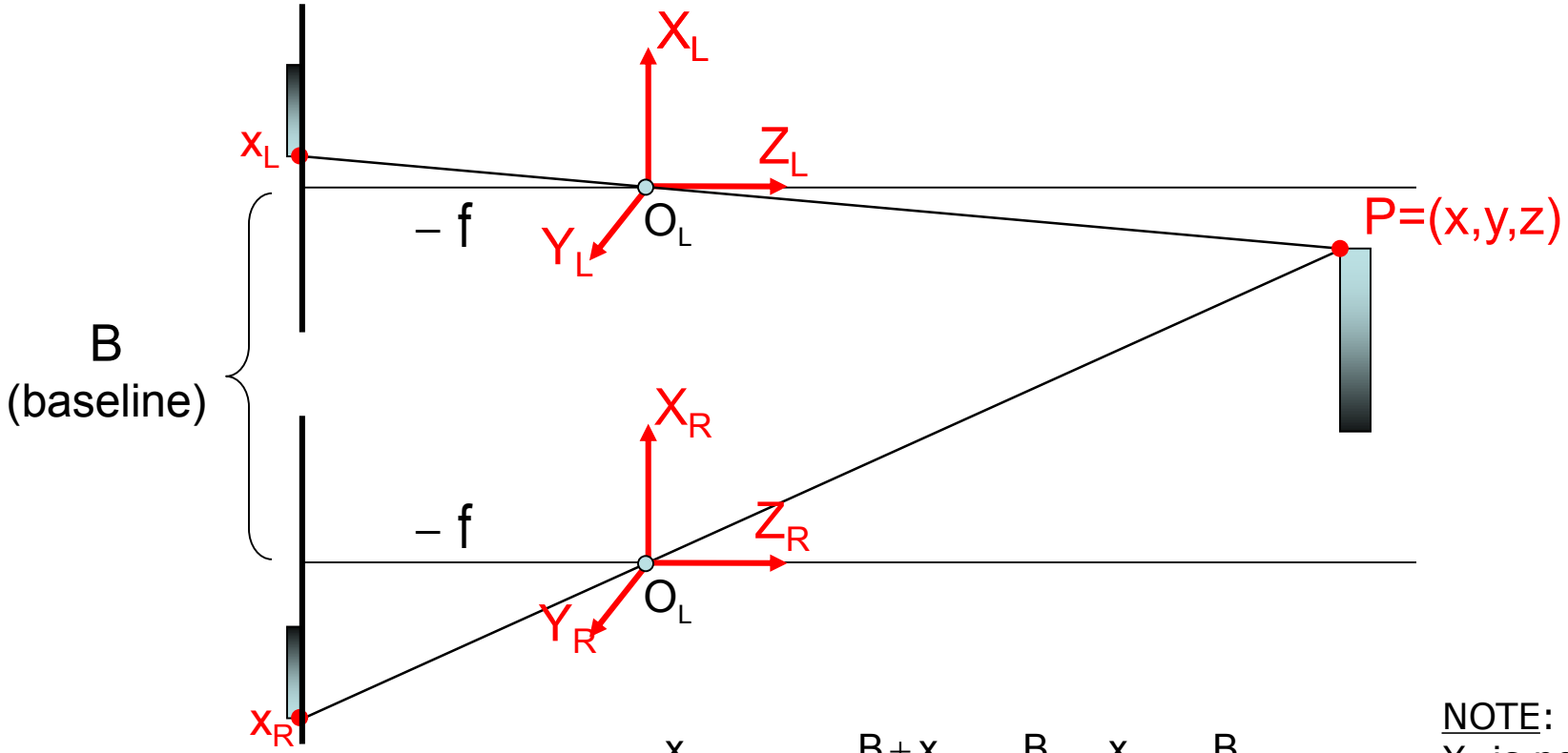
Note: in fact, the images are formed behind points C1 and C2 and are inverted …

A simple stereo system



Stereo system with
- equal cameras
- parallel optical axes
- baseline perpendicular to the optical axes

A simple stereo system



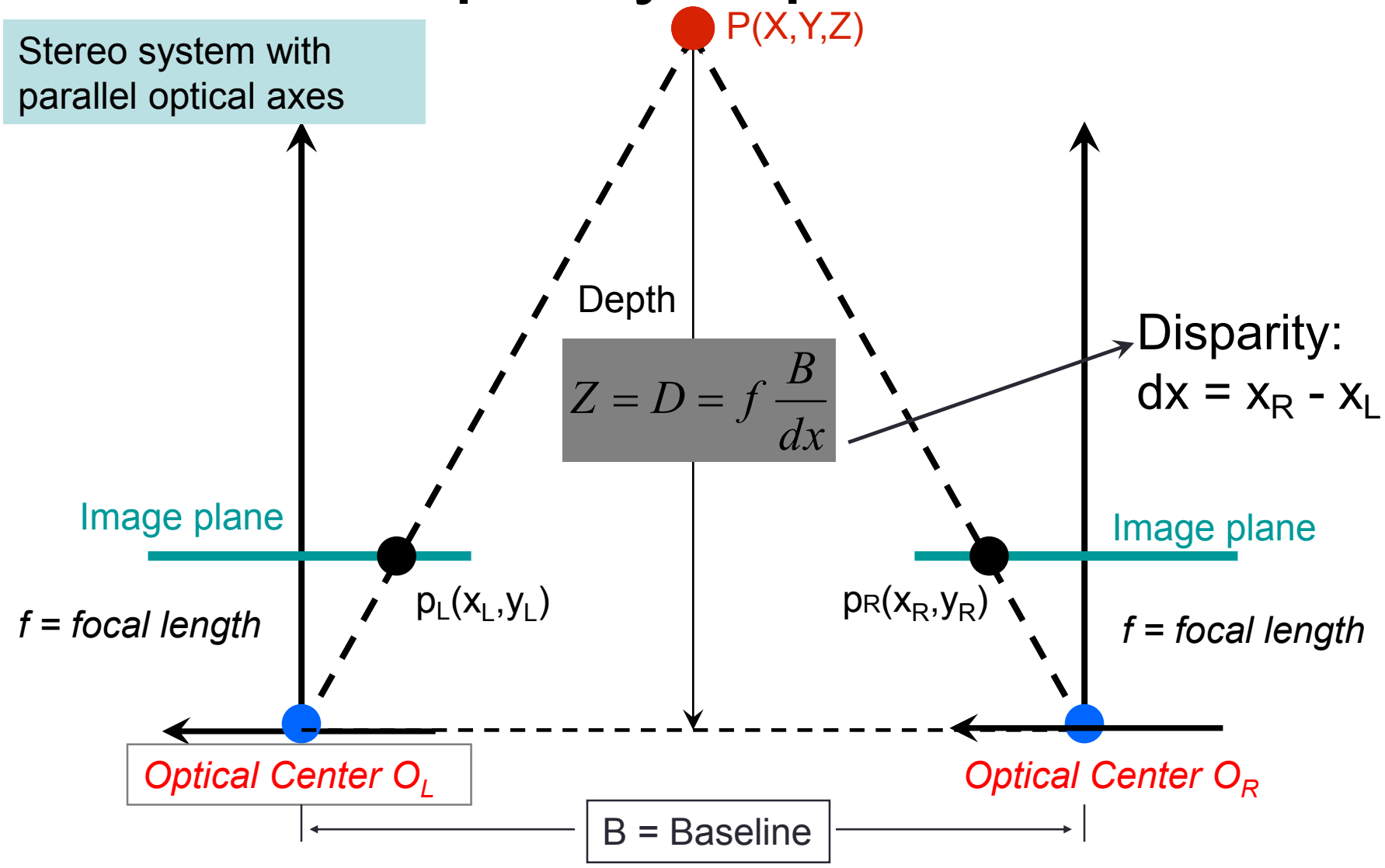
$$x_L = -f\frac{x}{z}, \quad x_R = -f\frac{B+x}{z} = -f\frac{B}{z} - f\frac{x}{z} = -f\frac{B}{z} + x_L$$

$$\Rightarrow z = \frac{fB}{x_L - x_R} = \frac{fB}{d}$$

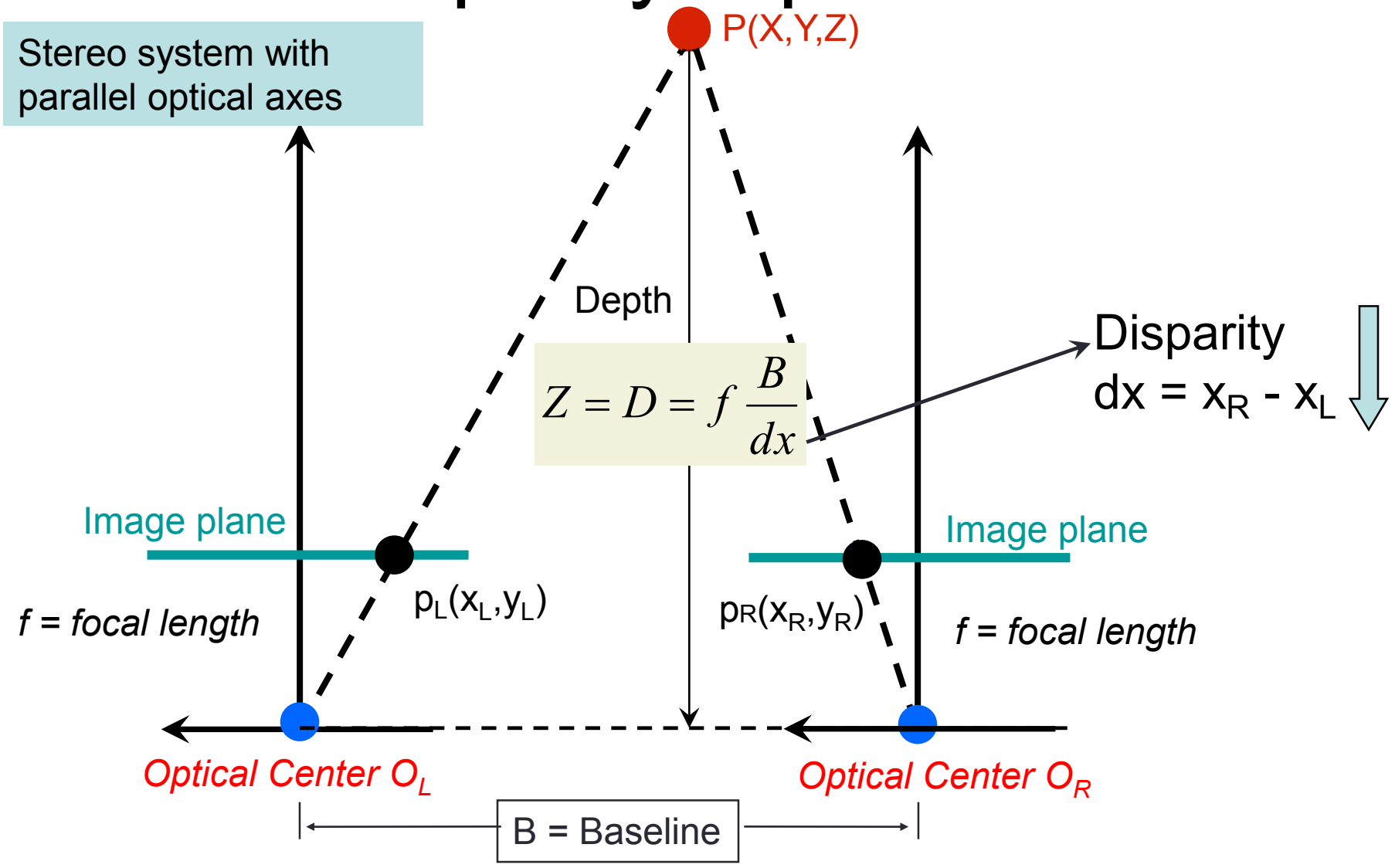
$$d = \text{disparity} = \frac{fB}{Z}$$

NOTE:
X_L is positive
X_R is negative

Disparity equation



Disparity equation



Depth from disparity

Image I(x,y)

Disparity map D(x,y)

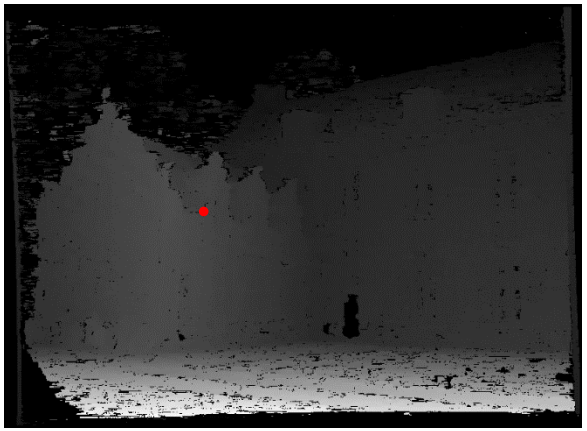


Image I'(x',y')

$(x',y')=(x+D(x,y), y)$

James Tompkin

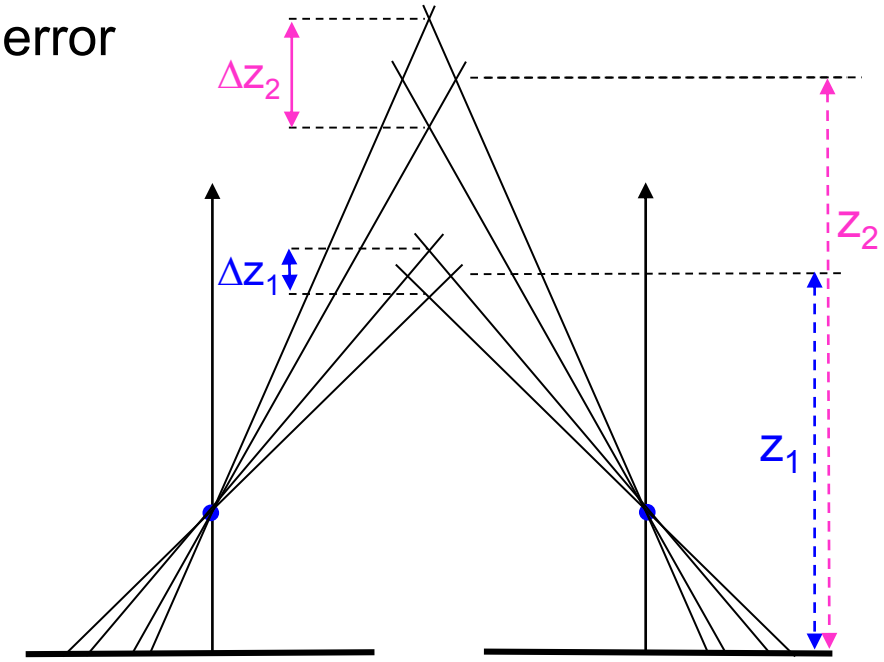
So if we could find the corresponding points in two images, we could estimate relative depth… (depth $\propto$ disparity)

Depth accuracy & resolution

- Given the same feature localization error (pixel size)

- Depth Accuracy (Depth Resolution) vs. Baseline
  - Depth Error $\propto$ 1/B
  - Long baseline
    - (+) better depth estimation
    - (–) smaller common FOV
    - (–) greater partial occlusion

- Depth Accuracy (Depth Resolution) vs. Depth
  - Disparity (>0) $\propto$ 1/ Depth
  - Depth Error $\propto$ Depth^2
  - The nearer the point, the better the depth estimation



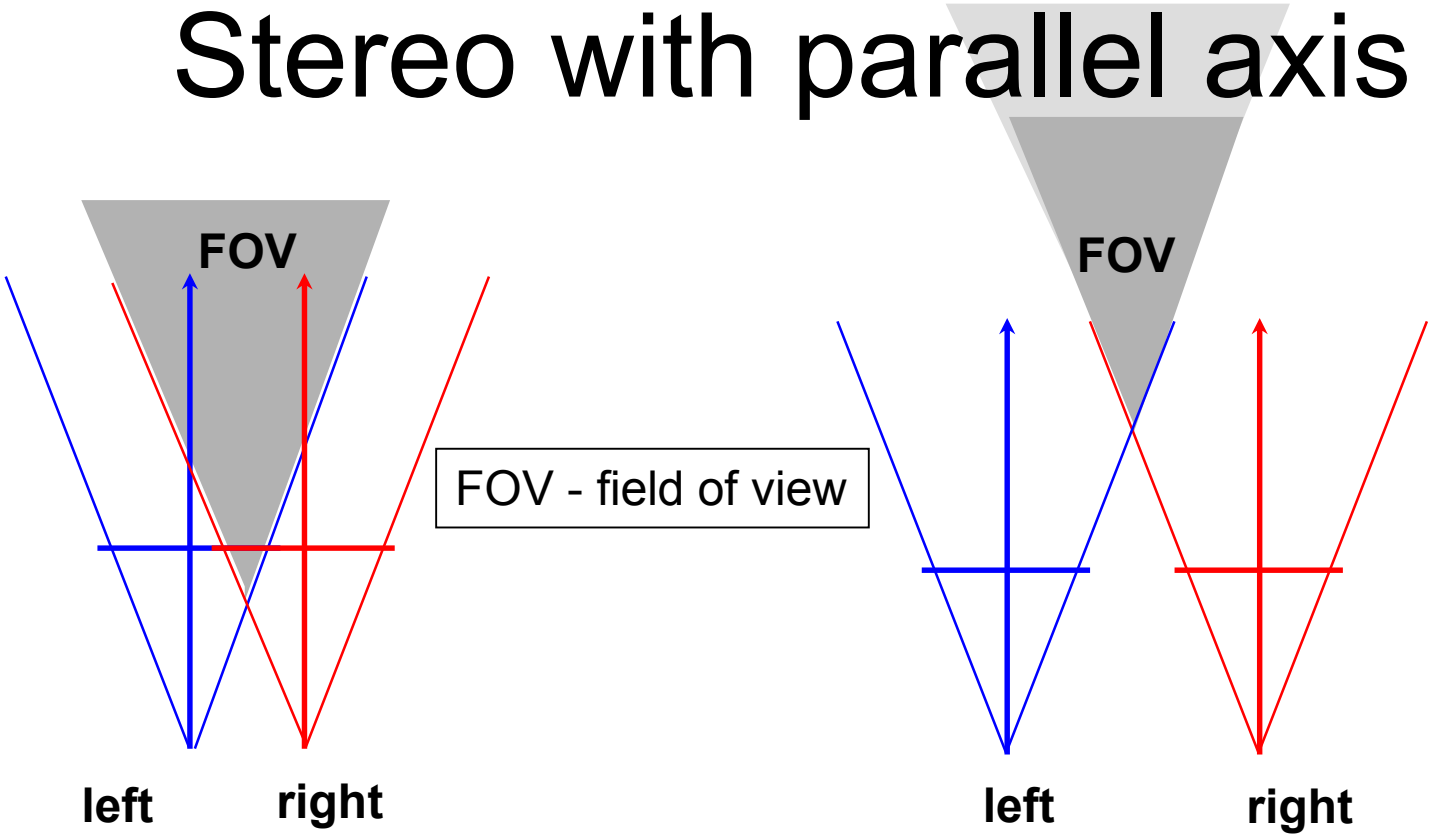
Absolute error

$$\Delta Z = \frac{Z^2}{fB}\Delta d$$

Relative error

$$\frac{\Delta Z}{Z} = \frac{Z}{fB}\Delta d$$

Stereo with parallel axis

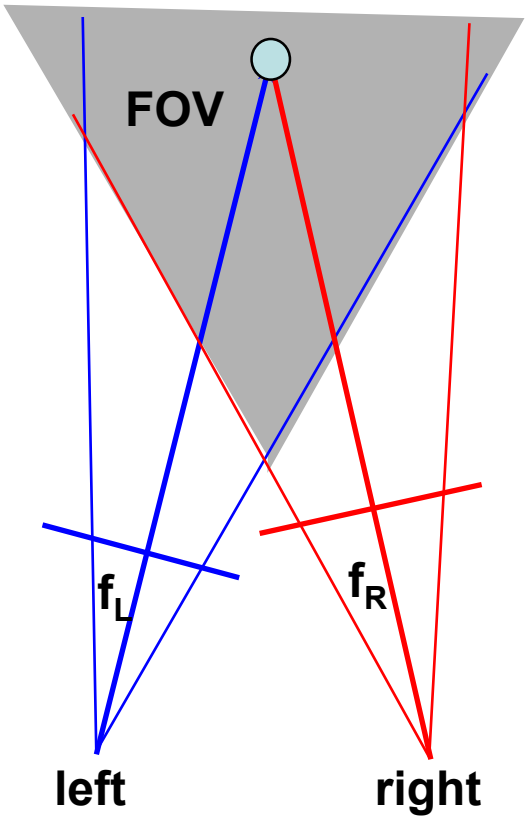


- Short baseline
  - large common FOV
  - large depth error
- Long baseline
  - small common FOV
  - small depth error
  - more occlusion problems

Stereo - general case

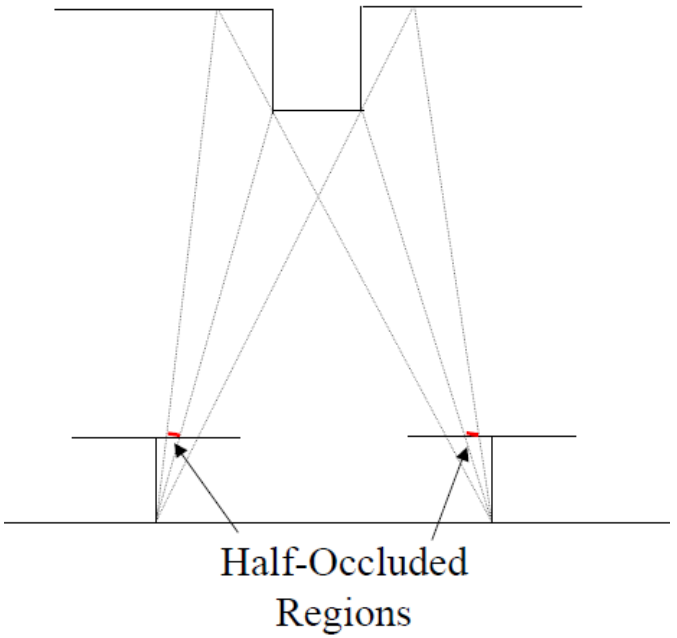
- Converging optical axis
  - the common FOV increases
- Different cameras (not common)
  - different focal length
  - different image size
  - …

➔ increased matching difficulties ☹



Half occlusion

- In certain situations
  there may be points in the scene
  that are visible from one of the two cameras
  but not from the other.
  These are referred to as half-occluded regions
- These regions are frequently associated with
  depth discontinuities in the scene.
- Half occluded regions
  pose a challenge to most
  simple stereo correspondence algorithms.
- Consequently many stereo algorithms
  have problems at depth discontinuities
- The human vision system, by contrast,
  seems to be sensitive to half-occluded regions
  and uses them to detect surface boundaries.

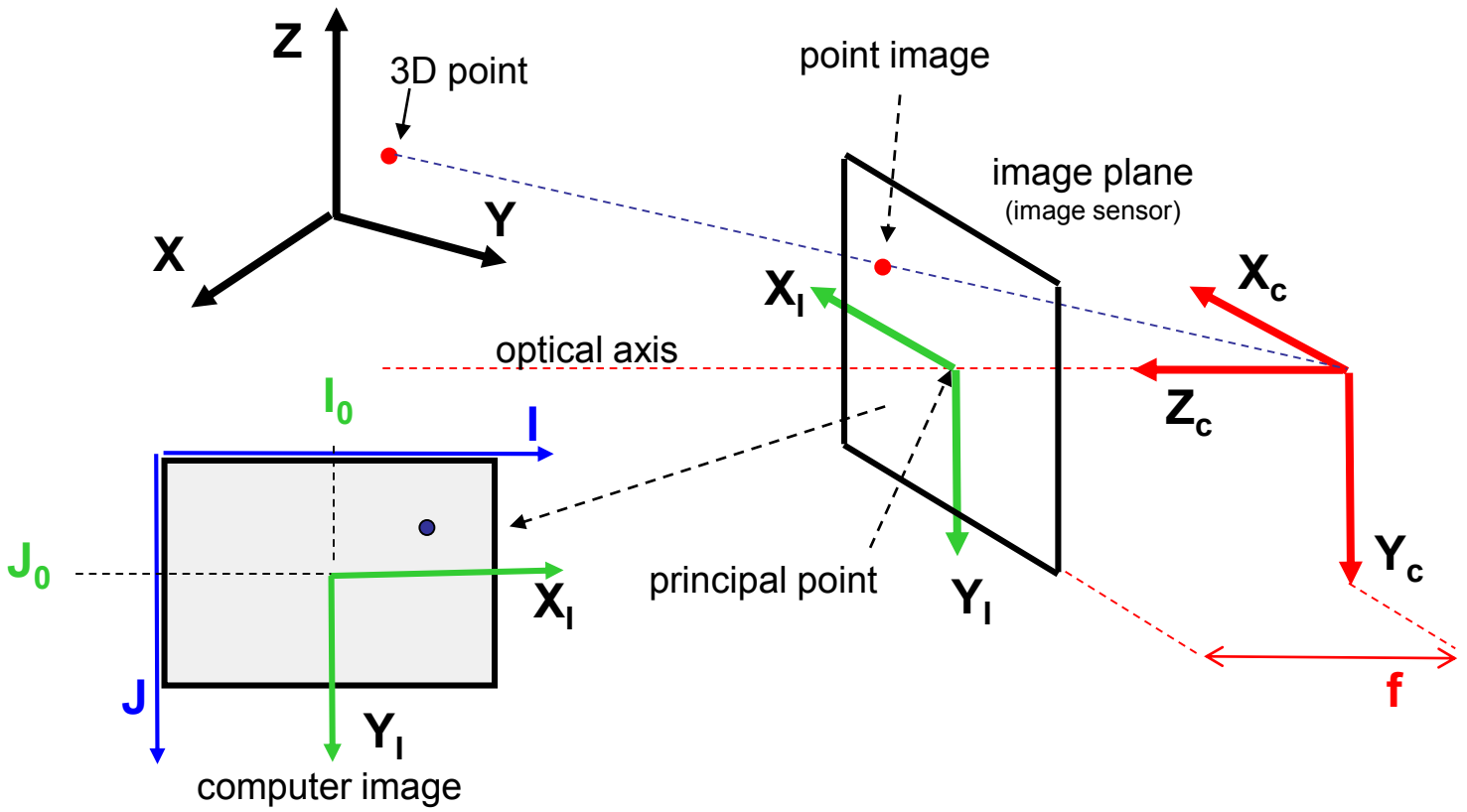


The 3D data acquisition process

- Camera calibration

- Stereo image pair acquisition
- Image matching and rectification
- Triangulation

Perspective projection matrix



Combining all the transformations:

$$\begin{bmatrix} wi \\ wj \\ w \end{bmatrix} = \begin{bmatrix} S_i f & 0 & I_0 & 0 \\ 0 & S_j f & J_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_X \\ R_{21} & R_{22} & R_{23} & T_Y \\ R_{31} & R_{32} & R_{33} & T_Z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} K_X & 0 & I_0 \\ 0 & K_Y & J_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_X \\ R_{21} & R_{22} & R_{23} & T_Y \\ R_{31} & R_{32} & R_{33} & T_Z \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = [K][R \mid T] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

$K_X = S_i f$
$K_Y = S_j f$

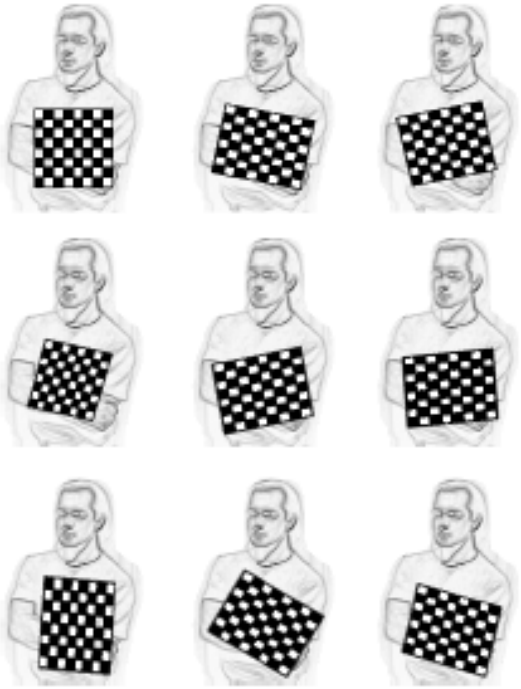
Perspective projection matrix

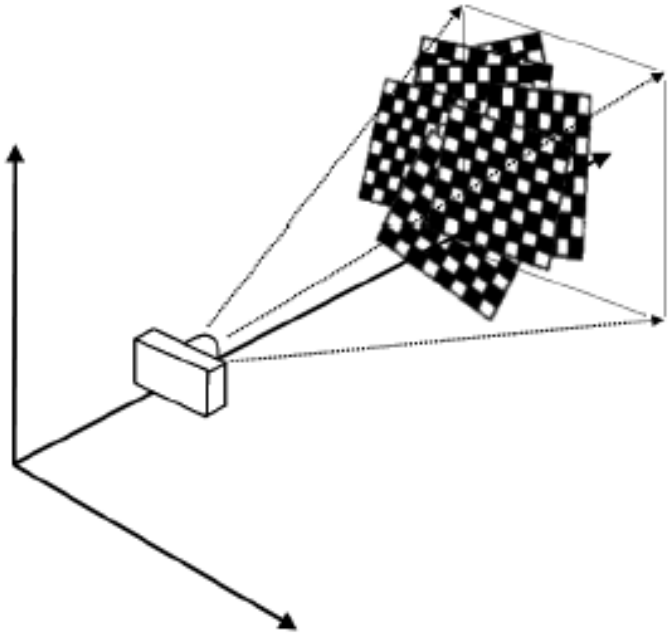
$$[C] = [K][R \mid T] = \begin{bmatrix} K_X & 0 & I_0 \\ 0 & K_Y & J_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_X \\ R_{21} & R_{22} & R_{23} & T_Y \\ R_{31} & R_{32} & R_{33} & T_Z \end{bmatrix} = \begin{bmatrix} K_X R_{11} + I_0 R_{31} & K_X R_{12} + I_0 R_{32} & K_X R_{13} + I_0 R_{33} & K_X T_X + I_0 T_Z \\ K_Y R_{21} + J_0 R_{31} & K_Y R_{22} + J_0 R_{32} & K_Y R_{23} + J_0 R_{33} & K_Y T_Y + J_0 T_Z \\ R_{31} & R_{32} & R_{33} & T_Z \end{bmatrix}$$

$$[C] = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{bmatrix} \qquad \begin{bmatrix} wi \\ wj \\ w \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

- $[C]$ - Perspective Projection Matrix of the camera
  - using it, the 2D image coordinates of a known 3D point can be obtained
  - also known as Direct Linear Transform (DLT) matrix
- $[K]$ - Intrinsic Parameter Matrix / Camera Matrix
  - represents the internal caracteristics of the camera
- $[R \mid T]$ - Extrinsic Parameter Matrix
  - represents the position and orientation of the camera relatively to the world coordinate system

OpenCV - camera calibration



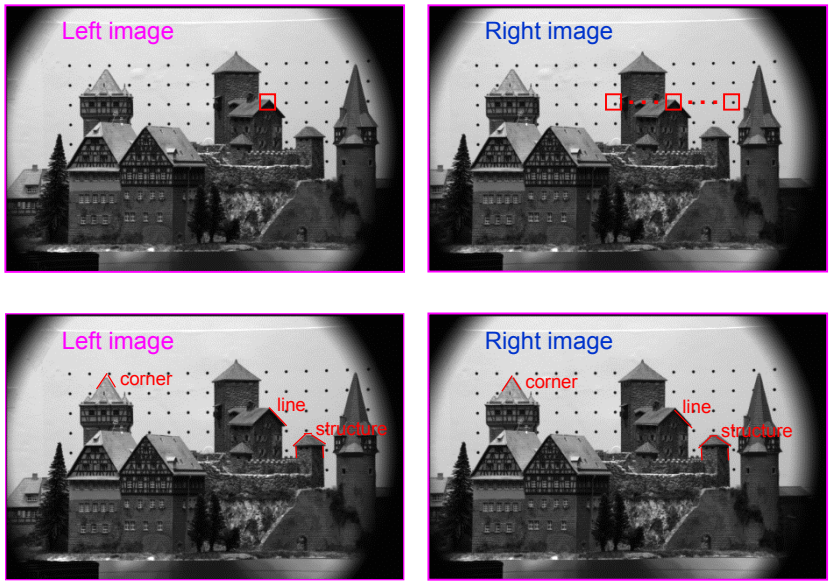


Images of a chessboard being held at various orientations (left)
provide enough information to completely solve for
the locations of those images in global coordinates (relative to the camera)
and the camera intrinsics

source: Learning OpenCV 3, Adrian Kaehler & Gary Bradski

Stereo for 3D data acquisition: The correspondence problem

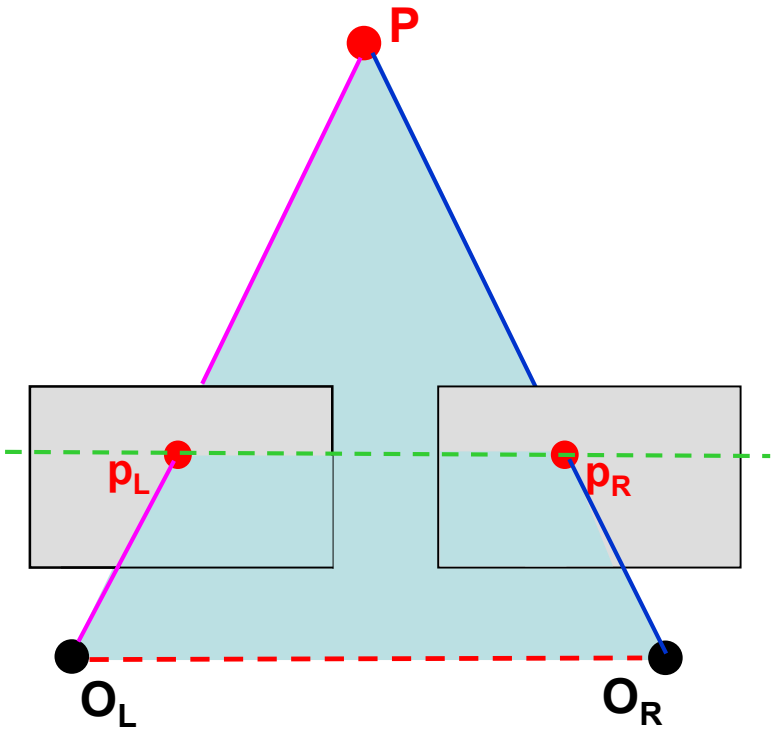
- Purpose
  - Given two images I_L and I_R of the same scene,
    for a point p_L in I_L, determine which point p_R in I_R it corresponds to.
    The term "correspond" means that they are the images of the same physical point P.
- Methods:
  - correlation-based / intensity-based
    - line-based
    - area-based
  - feature-based
- Problems:
  - scenes with no features
  - scenes with repeating features
  - different perspectives
  - perspective distortion
  - occlusion
  - photometric differences between views
  - different zooming
  - …matching errors

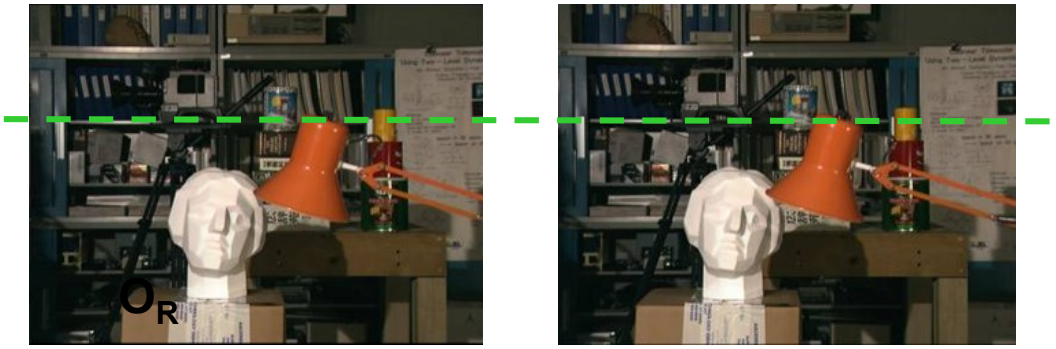


- There are several ways to attenuate these problems
- The ambiguity of correspondence search may be reduced by several, partly heuristic, constraints (epipolar, uniqueness, ordering, disparity limit, …)

The correspondence problem

- Matching is easier if the cameras are perfectly equal and have parallel axis
  - corresponding points are on the same lines, in each image

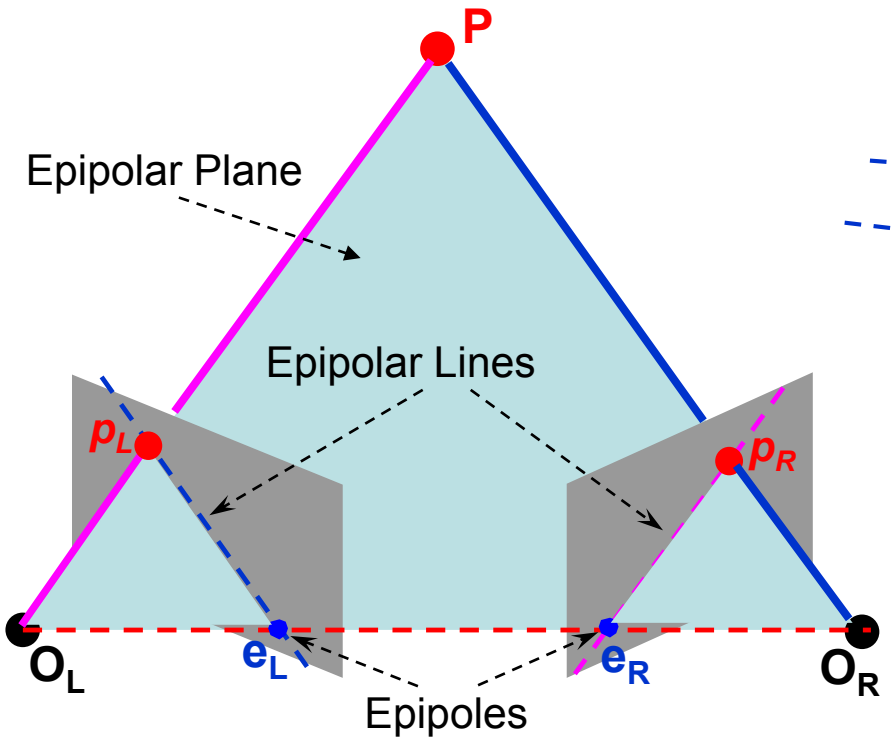


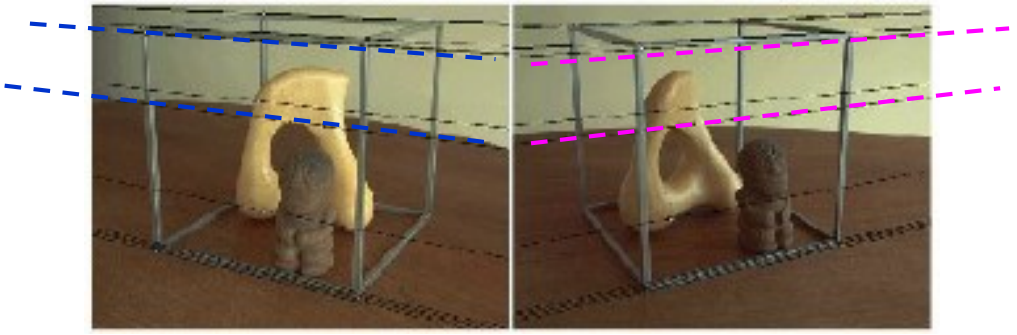


Tsukuba stereo pair

The correspondence problem

- When the camera axis are not parallel …
- Epipolar constraint
  - corresponding matches for point $\boldsymbol{p}_L$ must be searched along the epipolar line in the right image;
  - *or* corresponding matches for point $\boldsymbol{p}_R$ must be searched along the epipolar line in the lef timage.

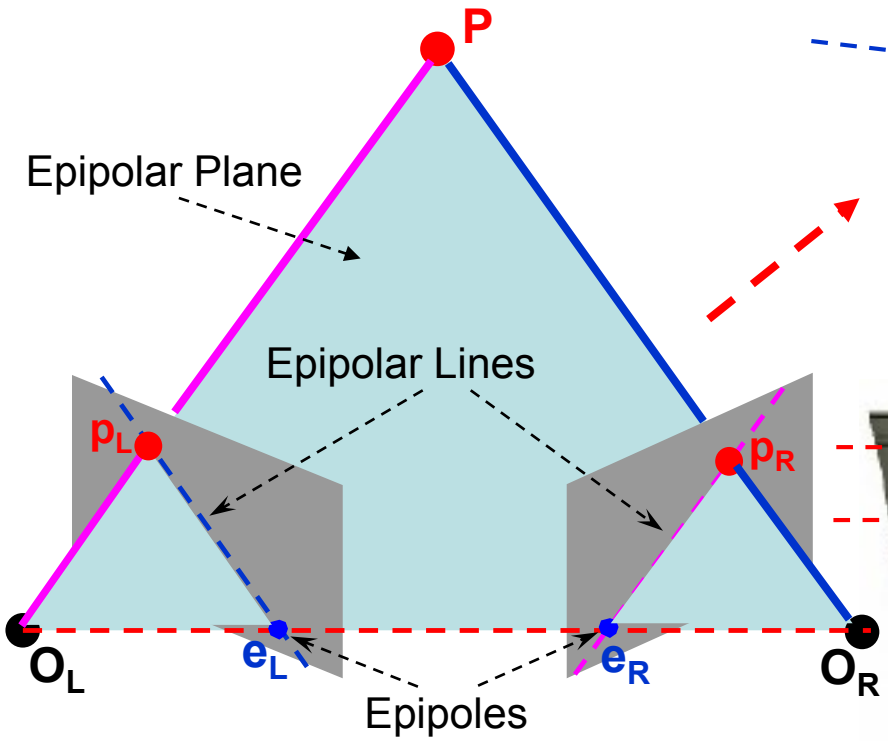


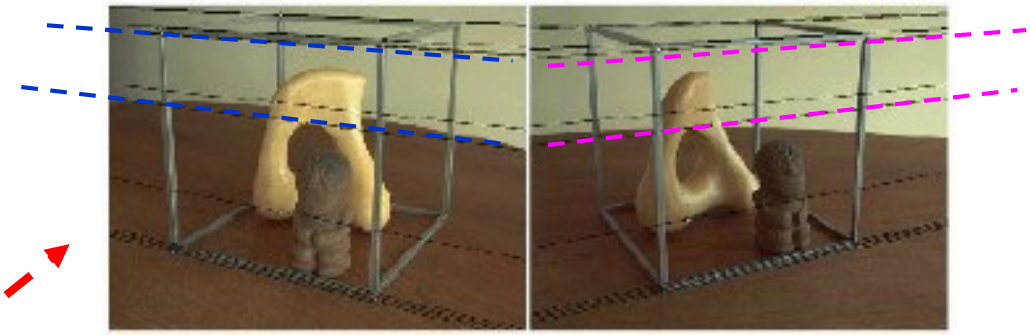


Acquired image pair
overlayed with several epipolar lines.

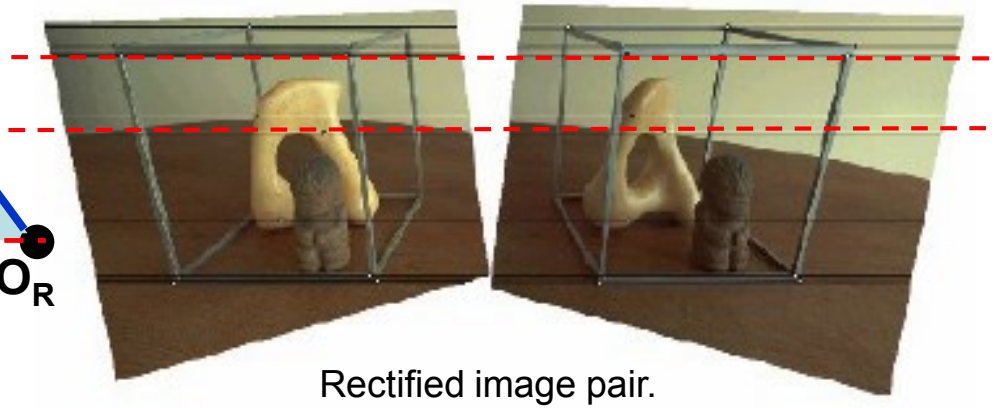
The correspondence problem

- If the axis are not parallel the images can be rectified





Original image pair
overlayed with several epipolar lines.



Rectified image pair.
The epipolar lines are horizontally aligned.

Stereo pair rectification

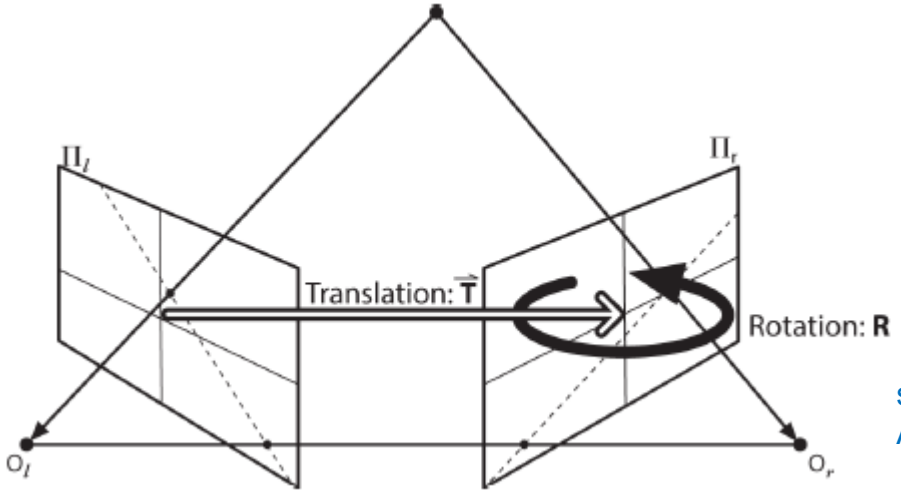
- Given a pair of stereo images,
  rectification determines
  a transformation (or warping) of each image
  such that pair of conjugate epipolar lines become
  collinear and parallel to one of the image axis,
  usually the horizontal one.
- The importance of rectification is that
  the correspondence problem,
  which involves 2D search in general,
  is reduced to 1D search on a scanline identified trivially
  (the same line in the other image).
  - The output of this step are images that are row-aligned
    - the two image planes are coplanar and
    - corresponding image rows on the two imagers are in fact collinear relative to each other.

Stereo pair rectification

- Assumptions and Problem Statement:
  - Given
    - a stereo pair of images
    - the intrinsic parameters of each camera
    - the extrinsic parameters of the system, R and T
  - Compute
    - the image transformation that makes conjugate epipolar lines collinear and parallel to the horizontal image axis

Stereo pair rectification

- The ingredients to do the rectification are:
  - **E** - the Essential Matrix and
  - **F** - the Fundamental Matrix
- The matrix **E** contains information about the translation and rotation that relate the two cameras in physical space (see figure)

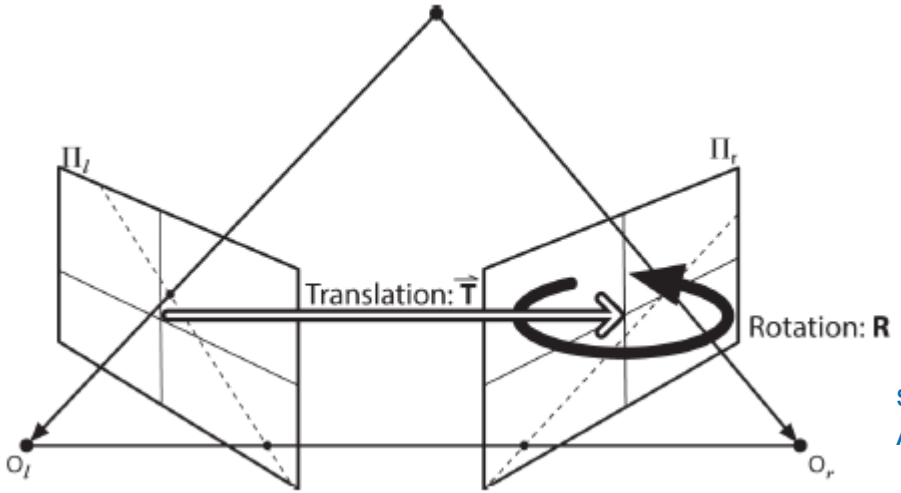


source: Learning OpenCV 3, Adrian Kaehler & Gary Bradski

- The matrix **F** contains the same information as **E** in addition to information about the intrinsics of both cameras

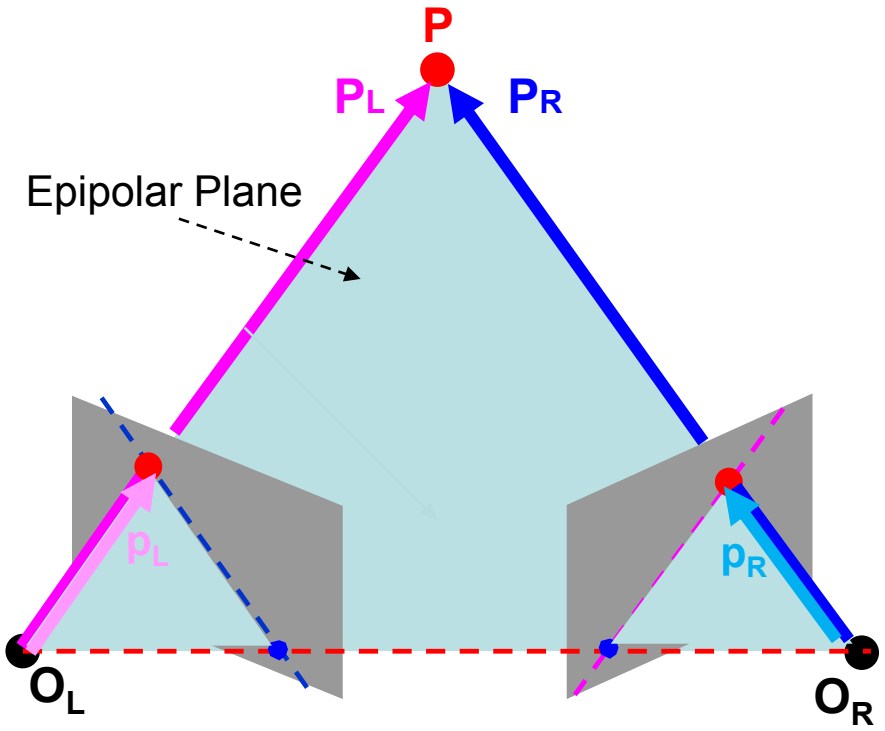
The essential matrix

- Given a 3D point P we would like to derive a relation that connects the observed 3D locations P_L and P_R on the two cameras axis systems



source: Learning OpenCV 3, Adrian Kaehler & Gary Bradski

The essential matrix



$\mathbf{P}_L = (X_L, Y_L, Z_L)$ and $\mathbf{P}_R = (X_R, Y_R, Z_R)$ identify the same point in space, **P**, using 2 different coordinate systems (of the 2 cameras)

$\mathbf{p}_L = (x_L, y_L, z_L)$ and $\mathbf{p}_R = (x_R, y_y, z_R)$ identify two different points in space, using 2 different coordinate systems (of the 2 cameras)

It can be demonstrated that:

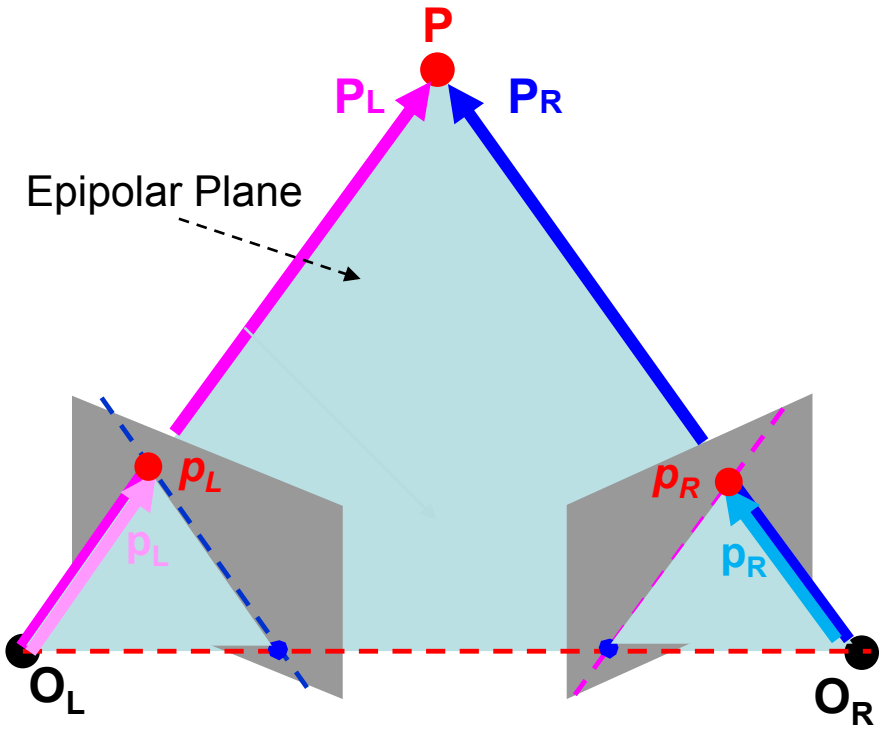
$$\mathbf{P}_R^T \mathbf{E}\, \mathbf{P}_L = 0$$

and

$$\mathbf{p}_R^T \mathbf{E}\, \mathbf{p}_L = 0$$

where **E** is the essential matrix (3x3)
and
T denotes the vector transposition operation

The fundamental matrix



It can be demonstrated that:

$$\mathbf{F} = \mathbf{K}_R^{-T}\, \mathbf{E}\, \mathbf{K}_L^{-1}$$

where $\mathbf{K}_R$ and $\mathbf{K}_L$ are
the intrinsic parameters matrices

Let
$\boldsymbol{p}_L = (i_L, j_L, 1)$ and $\boldsymbol{p}_R = (i_R, j_R, 1)$
be the pixel homogeneous coordinates of
$\mathbf{p}_L = (x_L, y_L, z_L)$ and $\mathbf{p}_R = (x_R, y_y, z_R)$
in each one of images
acquired by the 2 cameras

If the intrinsic parameters of each camera
are known, it can be demonstrated that:

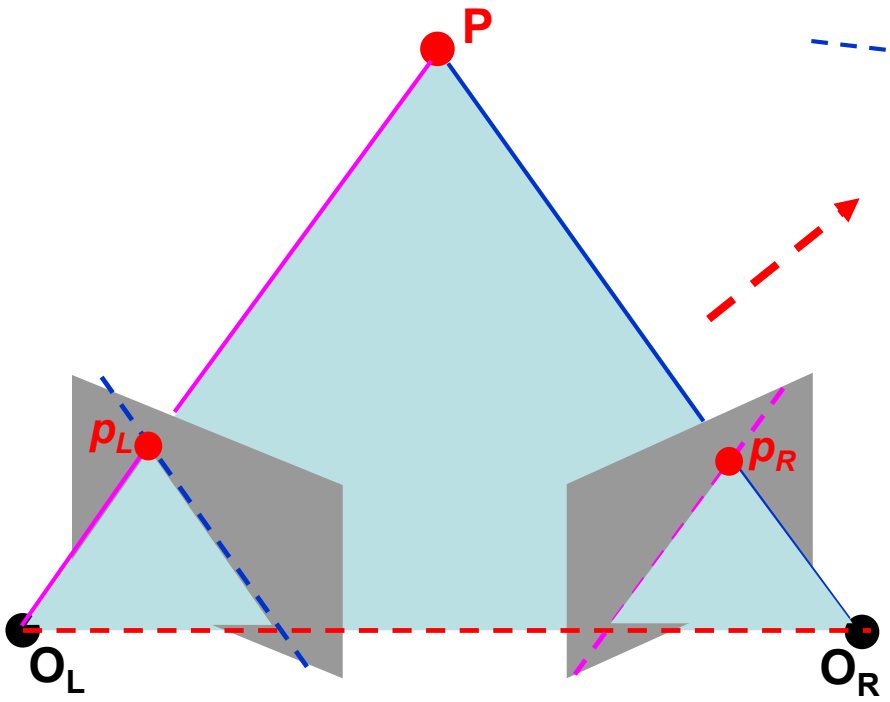
$$\boldsymbol{p}_R^{T}\, \mathbf{F}\, \boldsymbol{p}_L = 0$$

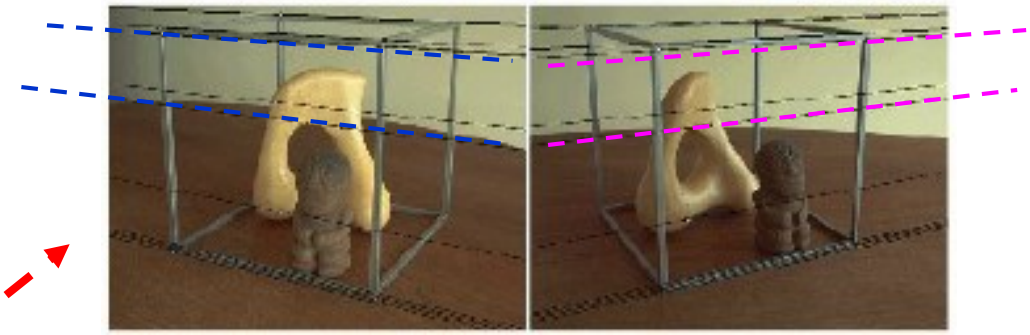
where **F** is the fundamental matrix (3x3)
and
T denotes the vector transposition operation

So, the fundamental matrix **F**
is just like the essential matrix **E**, except that
F operates in image pixel coordinates
whereas **E** operates in physical coordinates

Image rectification

- The images can be rectified using E and F matrices

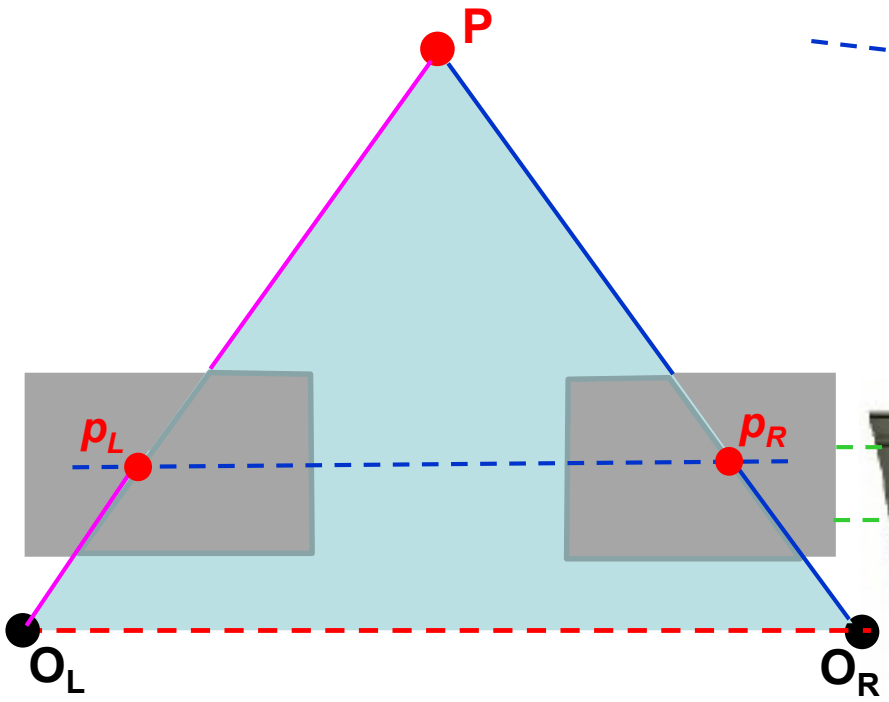


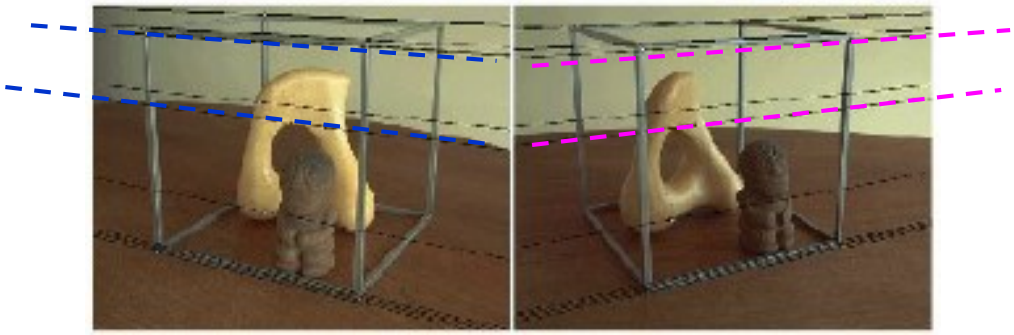


Original image pair
overlayed with several epipolar lines.

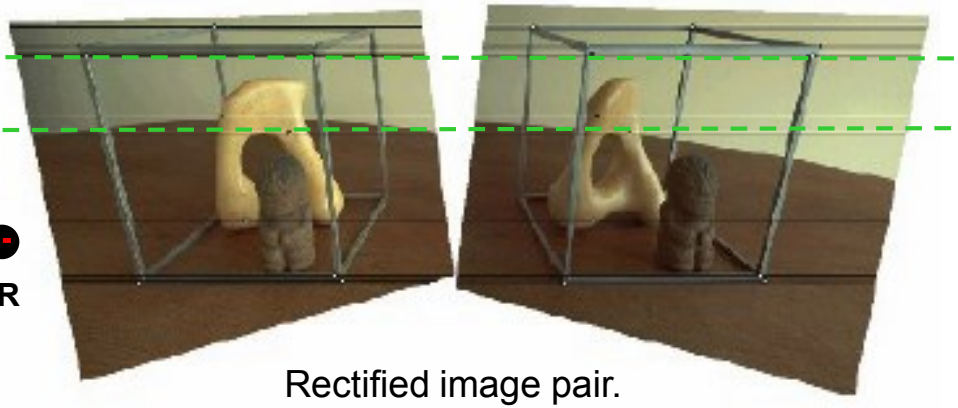
Image rectification

- The images can be rectified using E and F matrices



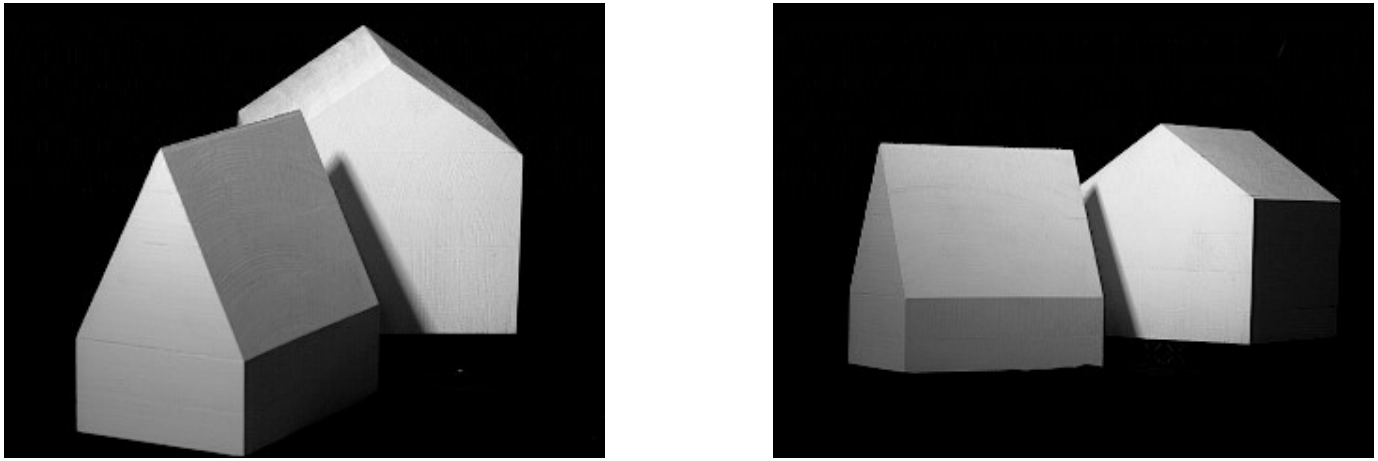


Original image pair
overlayed with several epipolar lines.

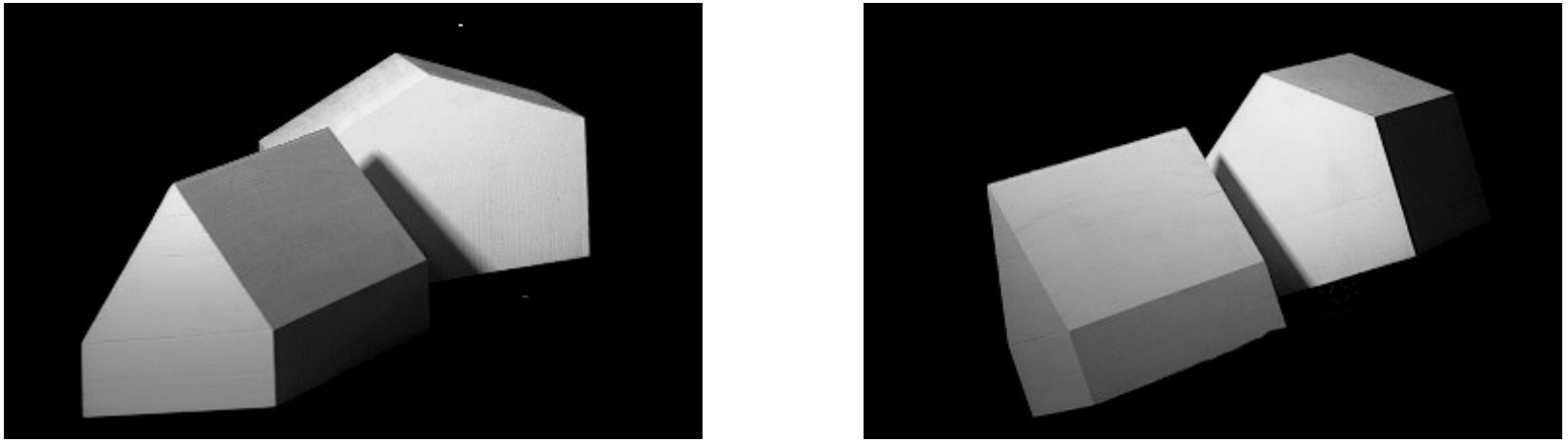


Rectified image pair.
The epipolar lines are horizontally aligned.

Image rectification



Original stereo pair



Rectified stereo pair

OpenCV stereo -
Calibration, Rectification, and Correspondence

- **Mat cv::findFundamentalMat()**
  - the main parameters are 2 arrays of **N** corresponding points in the stereo pair
  - 4 different methods can be used to calculate F:
    - 7-point algorithm => **N**=7 (*method extremely sensitive to outliers)*
    - 8-point algorithm => **N**≥8 (*method extremely sensitive to outliers, even with* N>8*)*
    - RANSAC algorithm => **N**≥8
    - LMedS algorithm => **N**≥8
  - The function calculates the fundamental matrix using one of four methods listed above and returns the found fundamental matrix.
    - Normally just one matrix is found.
    - But in case of the 7-point algorithm, the function may return up to 3 solutions (9×3 matrix that stores all 3 matrices sequentially).
  - The calculated fundamental matrix may be passed further to **computeCorrespondEpilines()** that finds the epipolar lines corresponding to the specified points.
  - It can also be passed to **stereoRectifyUncalibrated()** to compute the rectification transformation.
- **Mat cv::findEssentialMat()**

Stereo Calibration, Rectification, and Correspondence Code Example
(Learning OpenCV 3; A.Kaehler & G.Bradski, p.752)
https://github.com/oreillymedia/Learning-OpenCV-3_examples/blob/master/example_19-03.cpp

OpenCV stereo -
Calibration, Rectification, and Correspondence

- Stereo calibration is the process of computing the geometrical relationship between the two cameras in space.
  - Stereo calibration depends on finding the rotation matrix **R** and translation vector **T** between the two cameras, as depicted in a previous slide.
  - Both **R** and **T** are calculated by the function **cv::stereoCalibrate()**, which is similar to the function **cv::calibrateCamera()** except that we now have two cameras and our new function can compute (or make use of any prior computation of) the camera, distortion, essential, or fundamental matrices
  - To be clear on what stereo calibration gives you: the rotation matrix will put the right camera in the same plane as the left camera; this renders the two image planes parallel but not row-aligned (we'll see how row-alignment is accomplished in the section "Stereo Rectification")
- **double cv::stereoCalibrate()**
  - main inputs:
    - the calibration pattern points.
    - projections of the calibration pattern points, observed by the 1st camera
    - projections of the calibration pattern points, observed by the 2nd camera
  - computes:
    - intrinsic camera matrices for both cameras
    - lens distortion coefficients for both cameras
    - rotation matrix between the 1st and the 2nd camera coordinate systems
    - translation vector between the coordinate systems of the cameras
    - essential and fundamental matrices
  - returns the final value of the re-projection error

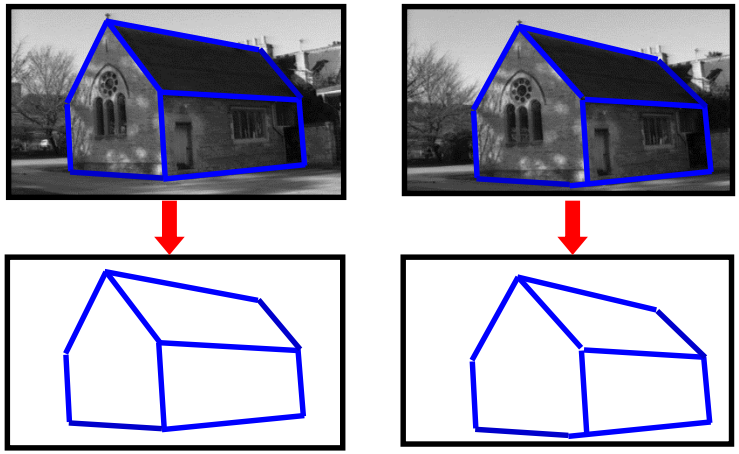
OpenCV stereo -
Calibration, Rectification, and Correspondence

- **void cv::computeCorrespondEpilines()**
  - computes, for a list of points in one image, the epipolar lines in the other image
- **void cv::stereoRectify()**
  - computes the rectification maps using the calibrated (Bouguet) method
- **bool cv::stereoRectifyUncalibrated()**
  - computes the rectification maps using the uncalibrated (Hartley) method
- **void cv::remap()**
  - computes the rectified images
- **class cv::StereoSGBM()**
  - compute the disparity maps
- **void cv::perspectiveTransform()**
  - transforms a set of points in an image to another set of points in an image,
- **void cv::reprojectImageTo3D()**
  - takes a single-channel disparity image and transforms each pixel's (x, y) coordinates along with that pixel's disparity - i.e., the vector (x, y, d) – to the corresponding three-dimensional point (X/W, Y/W, Z/W) by using the 4 × 4 reprojection matrix Q.

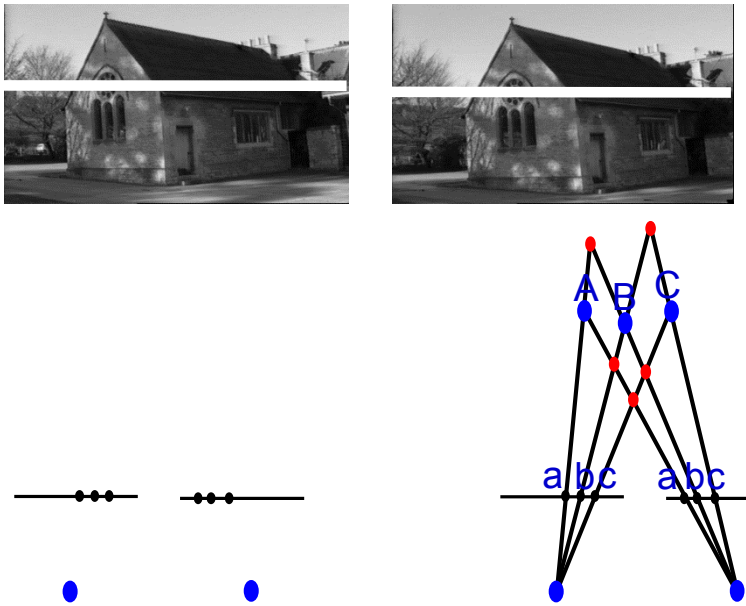
Stereo Calibration, Rectification, and Correspondence Code Example
(Learning OpenCV 3; A.Kaehler & G.Bradski, p.752)
https://github.com/oreillymedia/Learning-OpenCV-3_examples/blob/master/example_19-03.cpp

Correspondence problem Algorithms

Top-down



Bottom-up



- Group model (house, windows, etc) independently in each image
- Match points (vertices) between images

Correspondence problem Algorithms

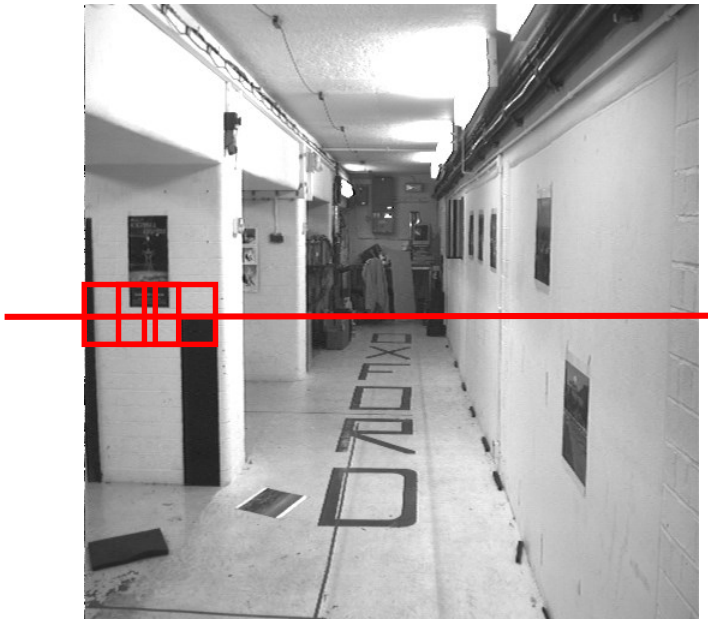
- Algorithms may be classified into two types:
  - Correlation-based
    - compute a correspondence at every pixel (dense)
  - Feature-based
    - compute correspondences only for feature points (sparse)
- Which method to use?
  - Correlation-based:
    - Dense maps, good for surface reconstruction
    - Require textured images
    - Sensitive to illumination variations
    - Inadequate for very different viewpoints
  - Feature-based:
    - Must find features first
    - Sparse maps, good for navigation

Dense correspondence algorithm

- Parallel camera example
- Search problem (geometric constraint):
  - for each point in the left image,
    the corresponding point in the right image lies on the epipolar line (1D ambiguity)
- Disambiguating assumption (photometric constraint):
  - the intensity neighbourhood of corresponding points are similar across images
- Measure similarity of neighbourhood intensity by some similarity measure

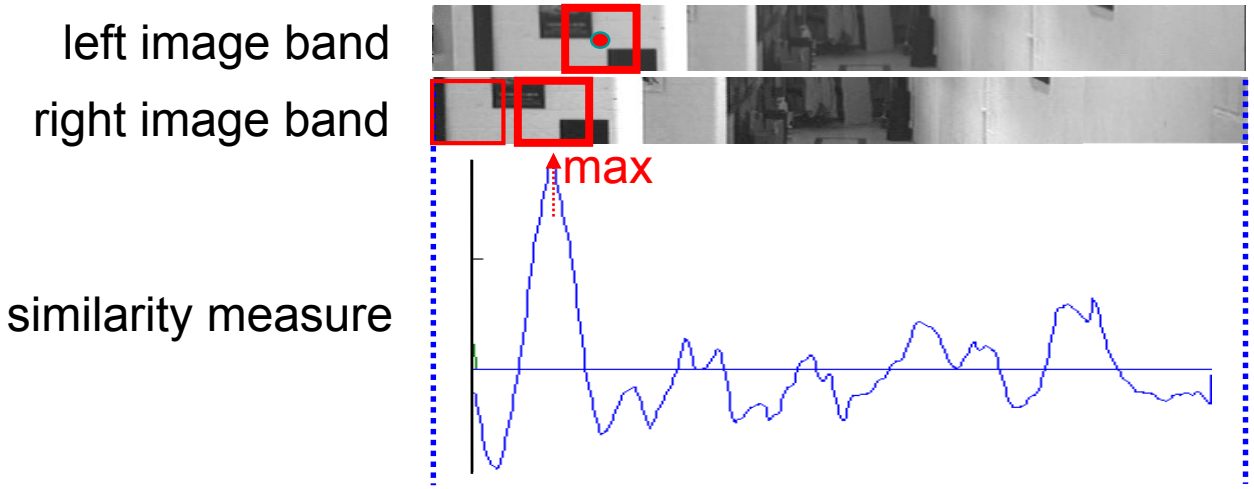
Dense correspondence algorithm

- Searching for the corresponding point …



adapted from Zisserman & Lazebnik

Dense correspondence algorithm



adapted from Zisserman & Lazebnik

Similarity measures

Sum of Square Differences

$$SSD(m,n) = \sum_x \sum_y \left(p(x,y) - r(x-m, y-n)\right)^2$$

Sum of Absolute Differences

$$SAD(m,n) = \sum_x \sum_y \left|p(x,y) - r(x-m, y-n)\right|$$

Sum of Absolute Differences Normalized

$$SADN(m,n) = \sum_x \sum_y \left|\left[p(x,y) - \overline{p(x,y)}\right] - \left[r(x-m, y-n) - \bar{r}\right]\right|$$

- Properties:
  - perfect matching => SSD=0, SAD=0, SADN=0
  - SSD e SAD are sensitive to illumination variations
  - All the measures are sensitive to contrast variations

Similarity measures

Correlation

$$C(m,n) = \sum_x \sum_y p(x,y)\, r(x-m,\, y-n)$$

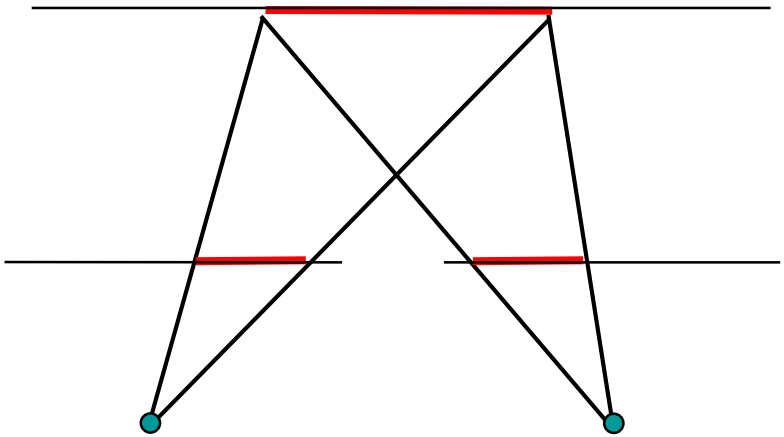
Normalized Cross-correlation

$$CN(m,n) = \frac{\sum_x \sum_y p(x,y)\, r(x-m,\, y-n)}{\left(\sum_x \sum_y p(x,y)^2\right)^{1/2} \left(\sum_x \sum_y r(x-m,\, y-n)^2\right)^{1/2}}$$

- Properties:
  - perfect matching => C maximum, CN=1

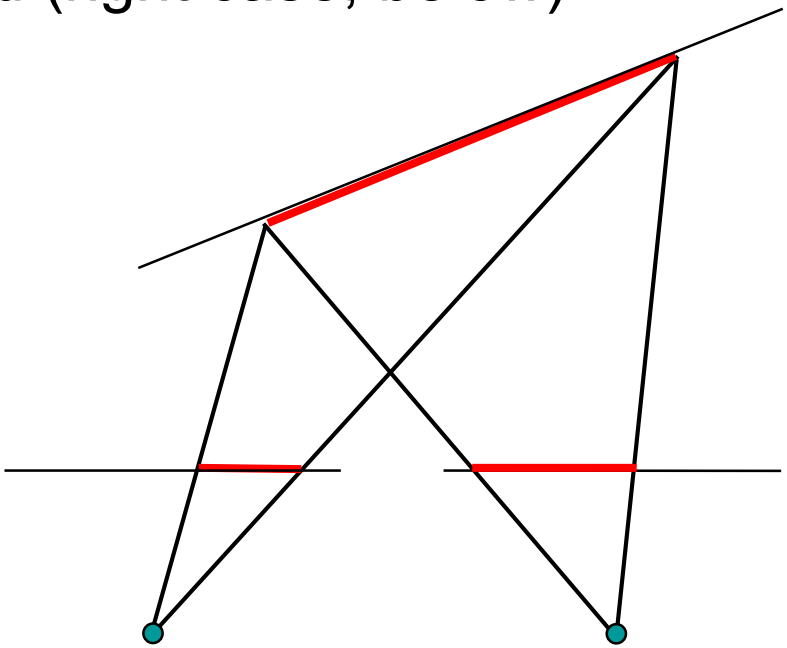
Correlation problems

- When the surfaces have no features.
- When the surfaces are slanted (right case, below)



fronto-parallel surface

imaged length is the same



slanting surface

imaged lengths differ

Constraining search for correspondence

- The ambiguity of correspondence search may be reduced by several (partly heuristic) constraints:
  - Epipolar constraint
    - reduces search space from 2D to 1D
  - Uniqueness constraint
    - a pixel in one image can correspond to only one pixel in another image
  - Ordering constraint
    - corresponding points lie in the same order on the epipolar line
  - Disparity smoothness constraint
    - disparity varies only slowly almost everywhere in the image
  - Disparity limit constraint
    - disparity must be smaller than a limit
  - Similarity/compatibility constraint
    - similar grey levels if matching intensities,
      similar contrast and orientation when matching edges, etc.
  - … (other)

- But …
  - there are several situations in which some of these constraints may be violated …
    - ex: at scene depth discontinuities, with transparent objects, …

Multi-view stereo & Structure from motion

- Multi-view stereo:
  (A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms, Seitz & al.)
  - The goal is to reconstruct a complete 3D object model from a collection of images taken from known camera viewpoints.
  - Several algorithms may be used for this purpose. Similar in spirit to the binocular stereo, they differentiate from each other taking into account: the scene representation, photoconsistency measure, visibility model, shape prior, reconstruction algorithm, and initialization requirements.

- Simultaneous Localization and Mapping (SLAM)
  - SLAM techniques build a map of an unknown environment and localize the sensor in the map with a strong focus on real-time operation.
  - Visual SLAM solutions:
    - single camera
    - stereo pair of cameras (ex: LSD-SLAM)
    - RGB-D camera (ex: KinectFusion)

Passive stereo problems

- Matching
  - scenes without features
  - feature matching
- Sparse data
  - $\Rightarrow$ interpolation
- Partial occlusion vs. accuracy
- Accuracy: is highly dependent upon
  - the viewing geometry (baseline, optical axis angle)
  - resolution of the image pair
  - accuracy of feature detection
  - algorithm robustness for matching
  - interpolation of data
  - …
  - surface reflection properties

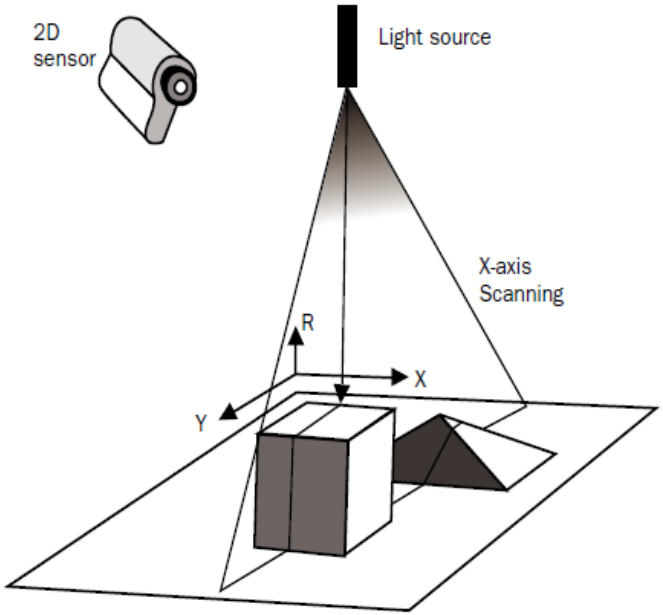
Active stereo / Structured light techniques

- Active stereo
  - When the scene has no (/ few) features
  - Illuminating the scene with a strong source of light (a point, a line of light or other kind of pattern) which can be observed by both cameras.
  - Known corresponding points provided in each image.
  - Triangulation can be done either
    - between the 2 acquired images (like in passive stereo)
    - between each acquired image and projected pattern
  - Depth maps can then be produced by sweeping the light source across the whole scene.
  - Laser light source typically employed
  - Active stereo can only be applied in controlled environments -- industrial applications.

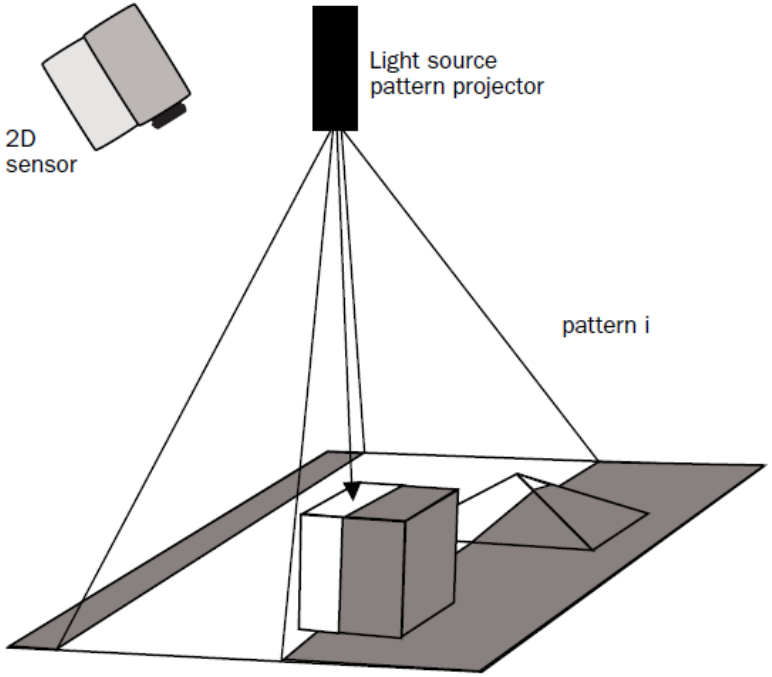
Structured-light techniques

- Basic principle
  - triangulation
- Also named “active-stereo”
  (not the same as “active-vision”)
- Structured-light types
  - one / multiple ray of light
  - one / multiple planes of light
- Advantages over passive stereo
  - scenes without features
  - correspondence problem easily solved
  - dense 3D data acquisition

Structured-light techniques

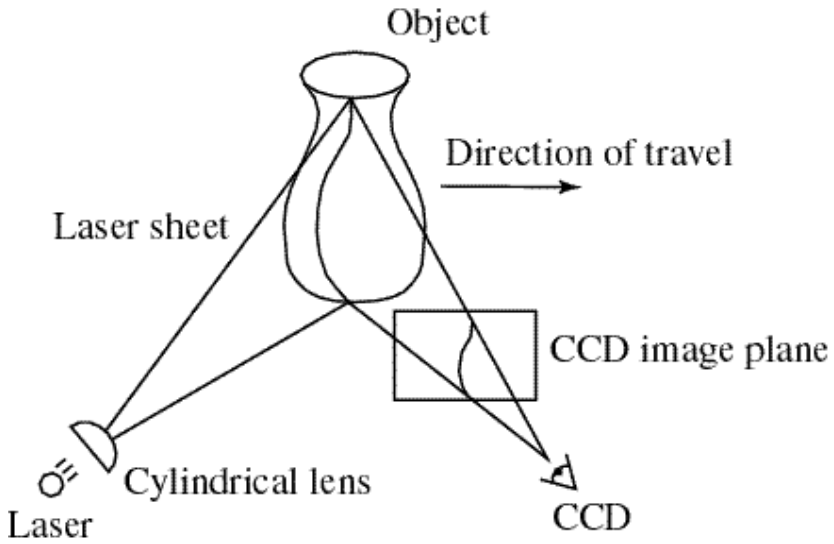


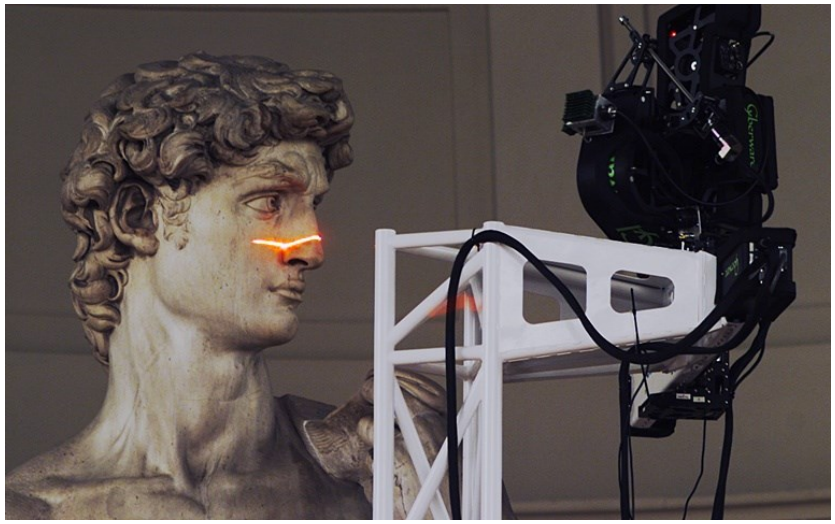
Sheet-of-light range imaging.
1-axis scanning is required
for complete range image acquisition.



Range imaging with gray coded light.
No scanning is required.

Laser scanning



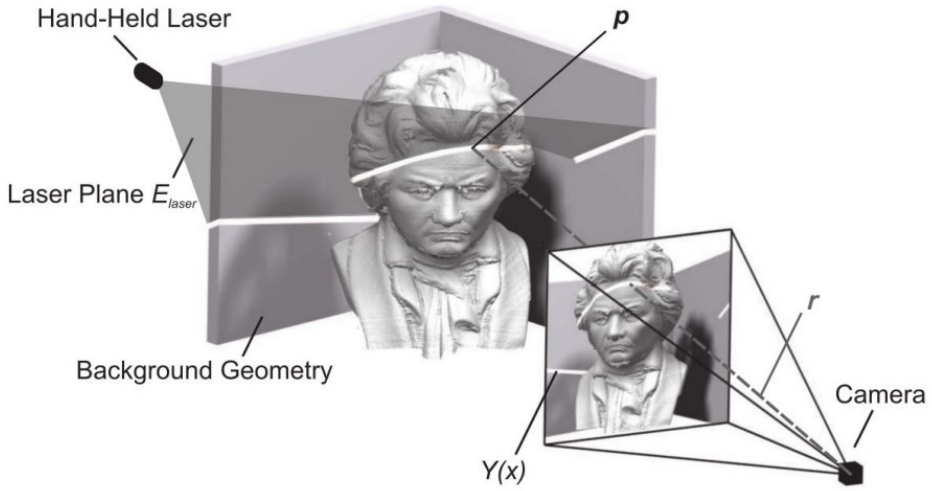


Digital Michelangelo Project, Levoy et al.
https://accademia.stanford.edu/mich/

- Optical triangulation
  - Project a single stripe of laser light
  - Scan it across the surface of the object
  - This is a very precise version of structured light scanning

Structured-light techniques

- Winkelbach et al. proposed a self-calibration method for a hand-held laser line projector by placing the object in front of a corner with two known planes.
- The principle of the approach is depicted in the figure.
- The system became popular as the David scanner and was acquired by Hewlett-Packard (HP).

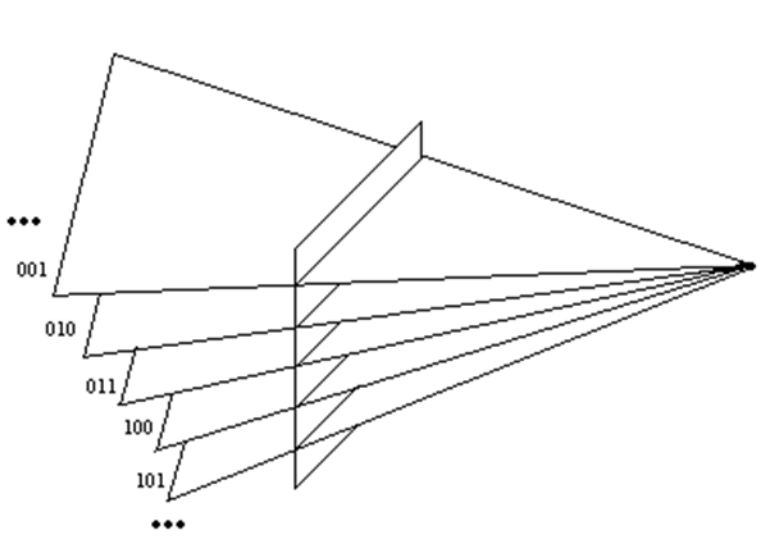


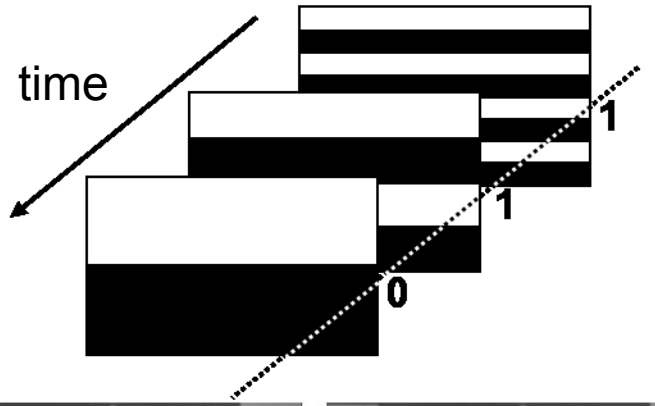
The principle of the Davidscanner

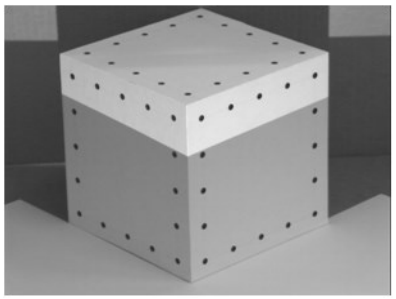
Simon Winkelbach, Sven Molkenstruck, Friedrich M. Wahl,
Low-cost laser range scanner and fast surface registration approach.
Lecture Notes in Computer Science, 4174:718, 2006.

Coded structured-light system

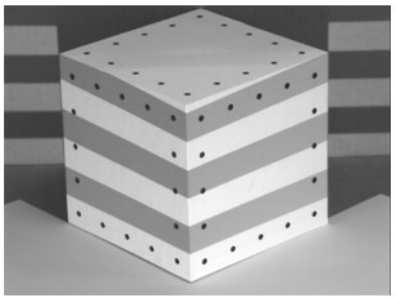
- Example with 8 light/shadow planes



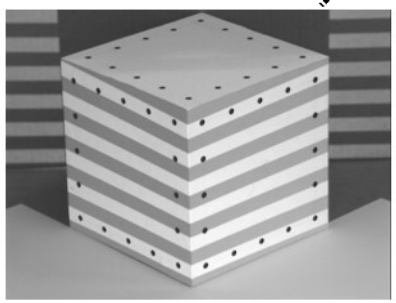




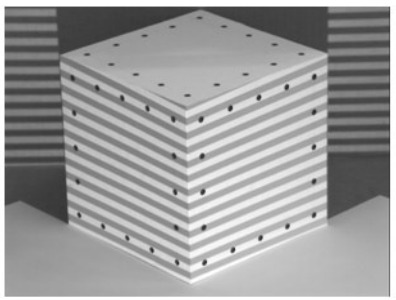
bit plane 7



bit plane 4



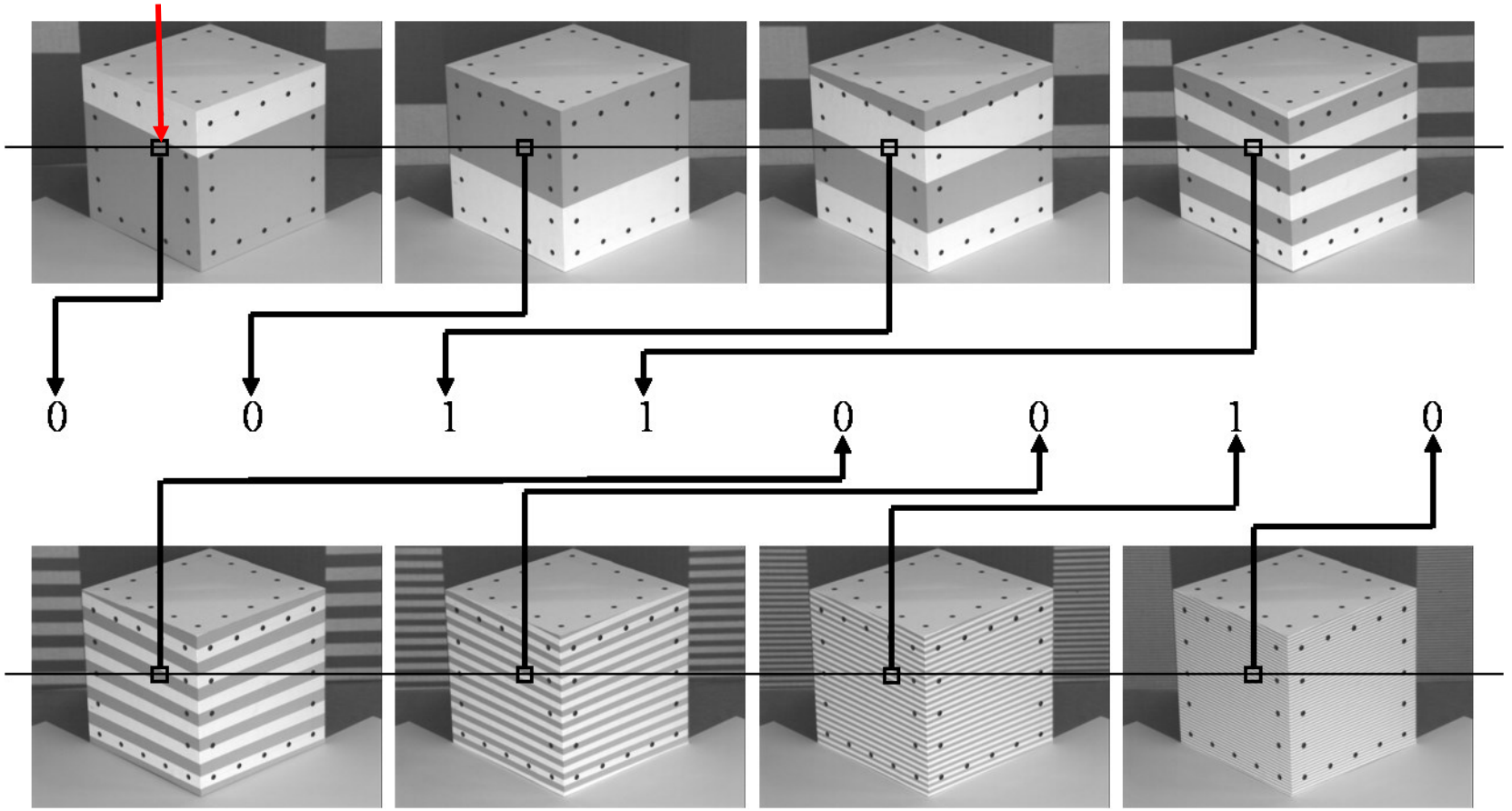
bit plane 3



bit plane 2

Coded structured-light system

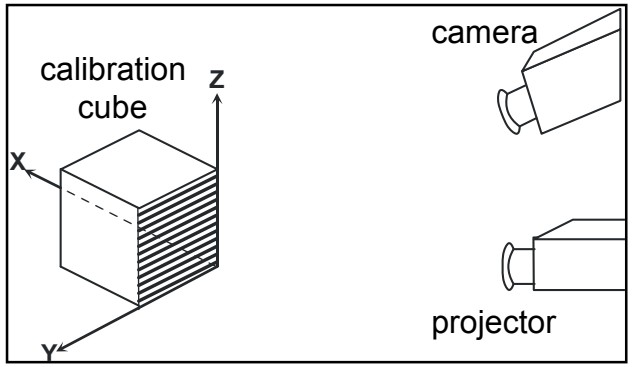
- Determination of the code of the light/shadow plane that "illuminates" pixel (i_c, j_c)



Coded structured-light system

- Camera model

$$\begin{bmatrix} w_c i_c \\ w_c j_c \\ w_c \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} & C_{14} \\ C_{21} & C_{22} & C_{23} & C_{24} \\ C_{31} & C_{32} & C_{33} & C_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



camera & projector calibration

- Projector model

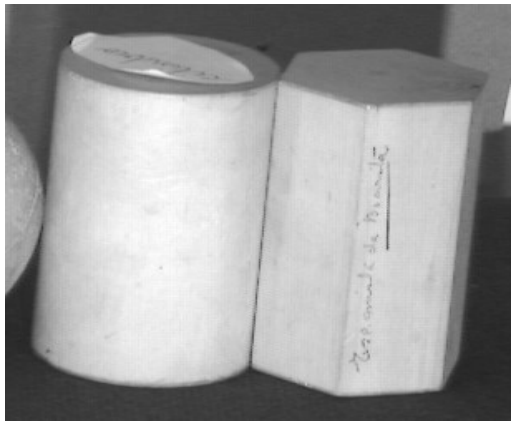
$$\begin{bmatrix} w_p i_p \\ -- \\ w_p \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ -- & -- & -- & -- \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

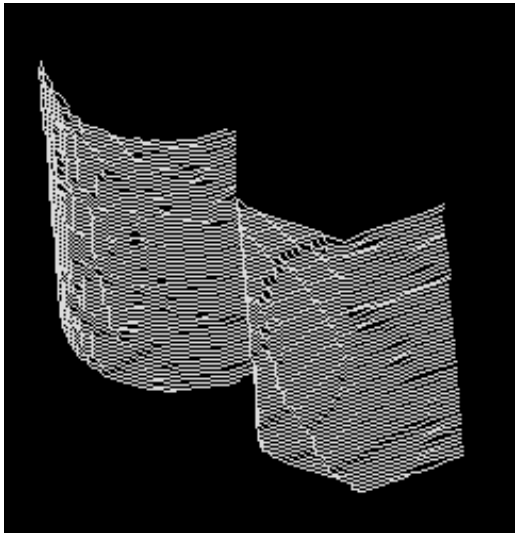
$$\begin{cases} (i_c C_{31} - C_{11}) \cdot x + (i_c C_{32} - C_{12}) \cdot y + (i_c C_{33} - C_{13}) \cdot z = C_{14} - i_c C_{34} \\ (j_c C_{31} - C_{21}) \cdot x + (j_c C_{32} - C_{22}) \cdot y + (j_c C_{33} - C_{23}) \cdot z = C_{24} - j_c C_{34} \\ (i_p P_{31} - P_{11}) \cdot x + (i_p P_{32} - P_{12}) \cdot y + (i_p P_{33} - P_{13}) \cdot z = P_{14} - i_p P_{34} \end{cases}$$

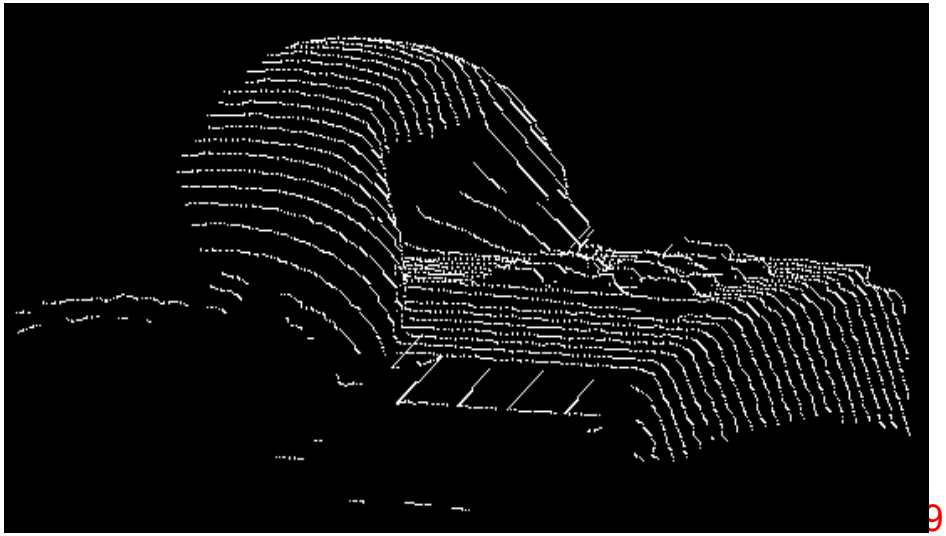
- Determines the equation of light/shadow plane given its code, i_p
- Solve this equation to determine the 3D position of the point imaged at (i_c, j_c), which is "illuminated" by plane i_p

Coded structured-light system

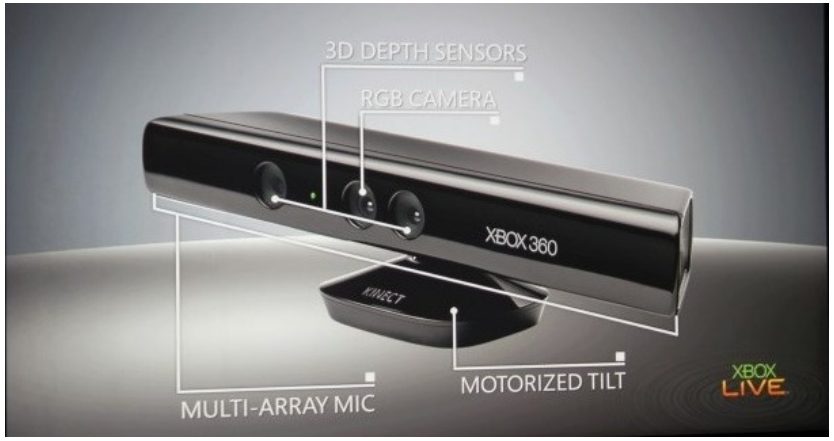
- Acquired data

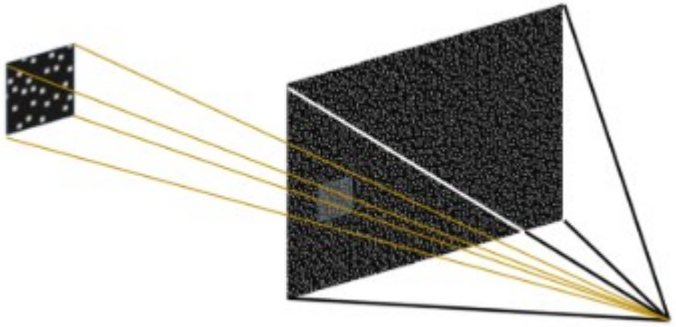




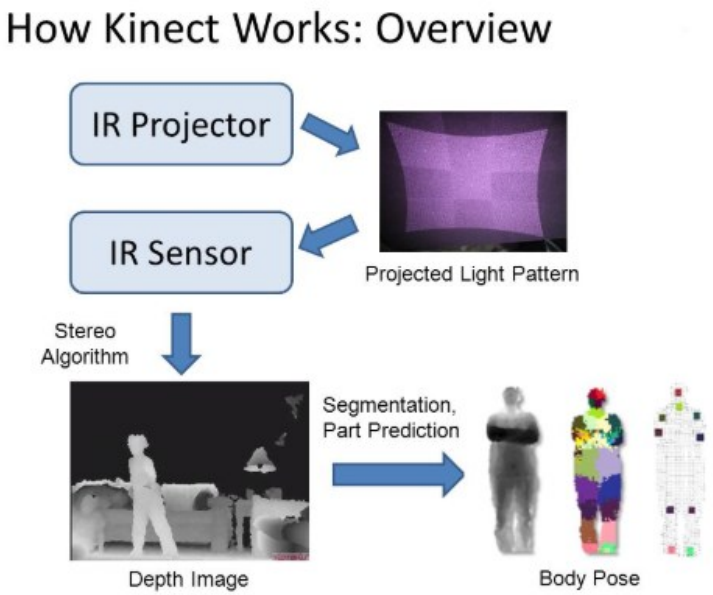


Microsoft Kinect





Pseudo-random infrared dot pattern projected by Kinect



Kinect projects dots which are glyphs

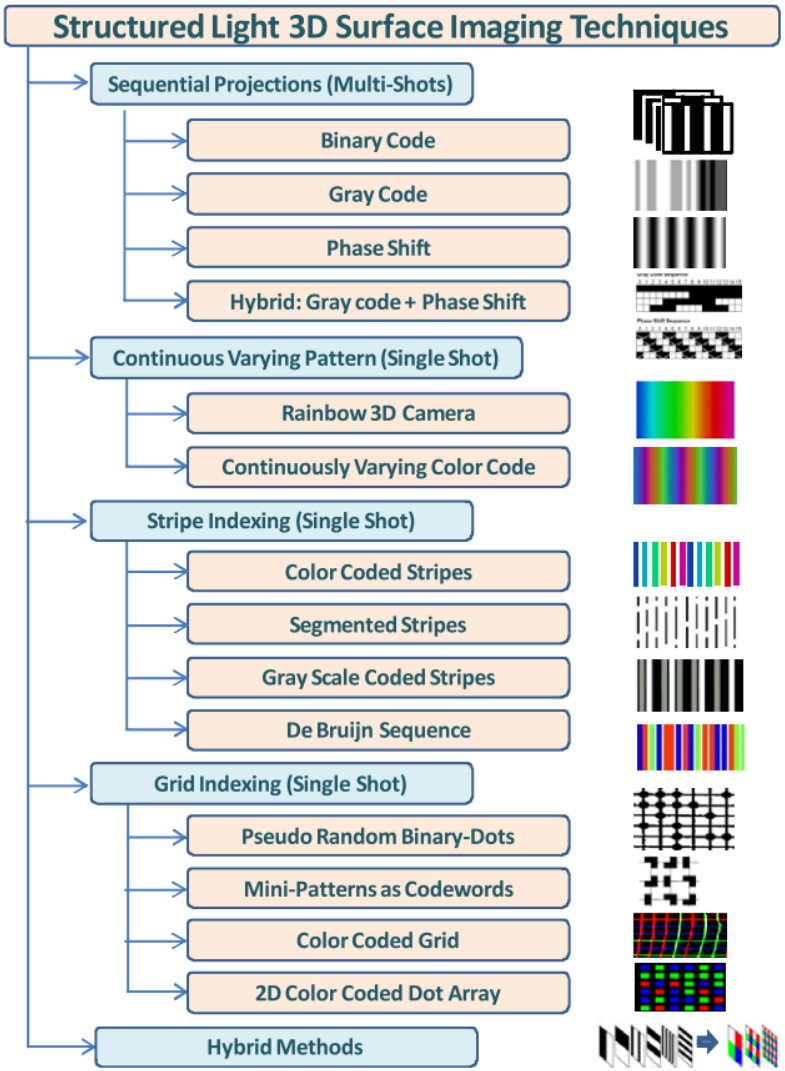
Limitations of Kinect

- Not that accurate
  unless you do more complex calibrations
- It was designed to interpret motions,
  not to build accurate 3d models or
  measure objects
- Frequency of infrared projector similar to sun
- So Kinect can not be used close to a window or be taken outdoors in bright sunlight
- Multiple Kinects interfere with each other

… but

- For Human Computer Interaction,
  Kinect is a big breakthrough;
  inexpensive and useful

Structured-light techniques



Classification framework of
structured-light surface 3D imaging techniques.

Jason Geng, “Structured-light 3D surface imaging: a tutorial”, Advances in Optics and Photonics 3, 128–160 (2011)

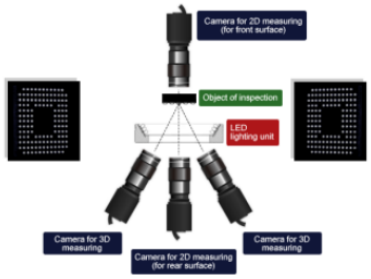
3-D vision sensor technology comparisons

	Stereoscopic Vision	Structured Light		Time of Flight
		Fixed Pattern	Programmable Pattern	
Depth Accuracy	mm to cm *Difficulty with smooth surface*	mm to cm	µm to mm *Variable patterns & different light sources improves accuracy*	mm to cm *Depends on resolution of sensor*
Scanning Speed	Medium *Limited by software complexity*	Fast *Limited by camera speed*	Fast/Medium *Limited by camera speed*	Fast *Limited by sensor speed*
Distance Range	Mid range	Very short to mid range *Depends on illumination power*	Very short to mid range *Depends on illumination power*	Short to long range *Depends on laser power & modulation*
Low Light Performance	Weak	Good	Good	Good
Outdoor Performance	Good	Weak/Fair *Depends on illumination power*	Weak/Fair *Depends on illumination power*	Fair *Depends on illumination power*
Software Complexity	High	Low/Middle	Middle/High	Low
Material Cost	Low	Middle	Middle/High	Middle

Michael Brading, Kenneth Salsman,Manjunath Somayaji,Brian Dipert,Tim Droz,Daniël Van Nieuwenhove,Pedro Gelabert. 2013. "3-D Sensors Bring Depth Discernment to Embedded Vision Designs." Embedded Vision Alliance. http://www.embeddedvision.com/platinum-members/embedded-vision-alliance/embedded-visiontraining/documents/pages/3d-sensors-depth-discernment.

https://www.edge-ai-vision.com/2013/07/3-d-sensors-bring-depth-discernment-to-embedded-vision-designs/

Methods for 3D Scanning and Measurement

	STEREO VISION	3D TOF	3D DLP
Key Principles	Stereo disparity	TOF	Structured light
Typical Application	Broad range – ADAS, industrial, consumer, UAV	RGBZ, RGBD camera Industrial, consumer, robotics	Industrial inspection, metrology
Example Systems			
Depth Accuracy	mm to cm Difficult with smooth surface	mm to cm Variable patterns and different light sources improve accuracy	µm to cm Depends on resolution of sensor
Scanning Speed	Medium Limited by software complexity	Medium to fast Limited by camera speed	Fast Limited by image acquisition speed
Distance Range	Mid-range	Short to long range	Very short to mid-range
Low Light Performance	Weak	Good	Light source dependent
Outdoor Performance	Good	Fair Depends on illumination power	Weak to fair Depends on illumination power
Software Complexity	High	Low	Medium to high

3D Machine Vision Reference Design Based on AM572x With DLP Structured Light, Texas Instruments

Methods for 3D Scanning and Measurement

	STEREO VISION	3D TOF	3D DLP
Pros	• Widely used across various applications • Wide range of software and hardware components available • Can be easily implemented on a mobile processor	• Better spatial resolution than stereo vision • Can design light sources for the specific scenarios and field of view • Can be used day or night, rain or shine • Lower power for applications that requires an Always-ON vision sensor, similar to Siri® or Cortana® in the voice side • The computing is significantly simpler than stereo vision	• Can identify anomalies in a flat surface • Can design light source to optimize reflection for the targeted objects • No interference • Allows projection of multiple patterns on the same object to extract features • Allows creation of complex patterns • Allows adaptive pattern generation
Cons	• Not usable in dark environment or adverse weather conditions • Objects required to have identifiable geometric features • May give erroneous depth if background and object mix colors	• Interference between multiple units • Needed lenses for both the sensor and light sources • Power of the light sources usually exceeds processor power significantly	• Specialized system typically targeted for inspection of defects, shapes, size
Material Cost	Low	Middle	Middle to high

3D Machine Vision Reference Design Based on AM572x With DLP Structured Light, Texas Instruments

Range/Depth cameras

Intel REALSENSE cameras

L515 D455 D435I / D435 D415 SR305

	L515	D455	D435I/D435	D415	SR305
Depth Technology	LiDAR	Active IR Stereo	Active IR Stereo	Active IR Stereo	Coded Light
Depth Accuracy	~5 mm to ~14 mm thru 9 m2	<2% at 4 m2	<2% at 2 m2	<2% at 2 m2	—
Ideal Range	.25 m to 9 m3	.6 m to 6 m	.3 m to 3 m	.5 m to 3 m	—
Use Environment	Indoor	Indoor/Outdoor	Indoor/Outdoor	Indoor/Outdoor	Indoor

https://www.intel.com/content/www/us/en/architecture-and-technology/realsense-overview.html

IDS cameras

IDS NXT rio IDS NXT rome Ensenso XR Ensenso X Ensenso N uEye ACP

https://en.ids-imaging.com/store/products/cameras.html

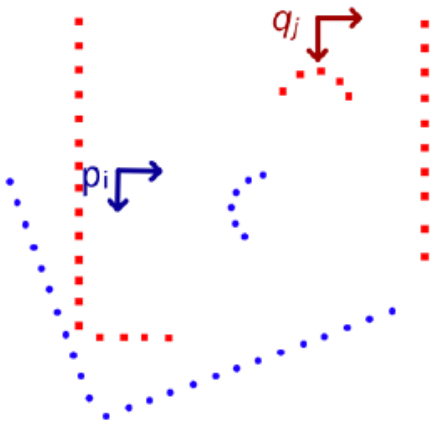
NOTE: the mention of any company or the use of a photo of any company's product is not an endorsement for that company or their product.

Aligning depth/range images

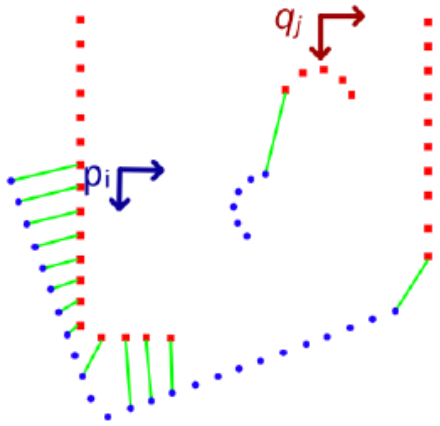
- A single range scan is not sufficient to describe a complex surface
- Need techniques to register multiple range images
- => Iterative Closest Point (ICP) algorithm and others

B. Curless and M. Levoy, A Volumetric Method for Building Complex Models from Range Images, SIGGRAPH 1996

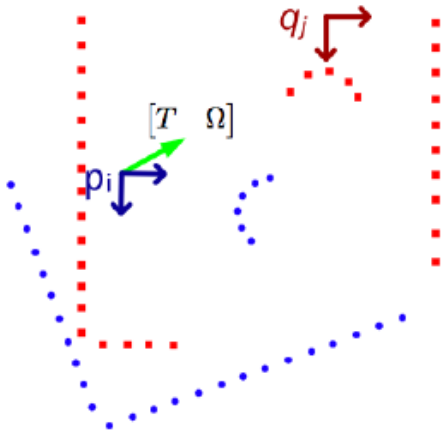
Iterative Closest Point – in 2D



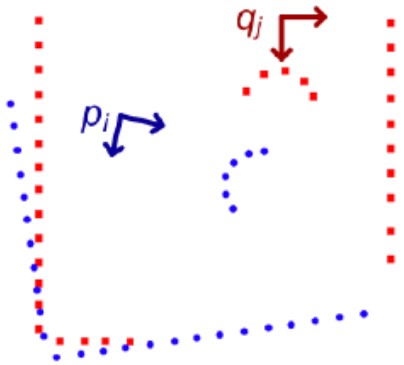
1- point clouds



2- transformation minimizing error



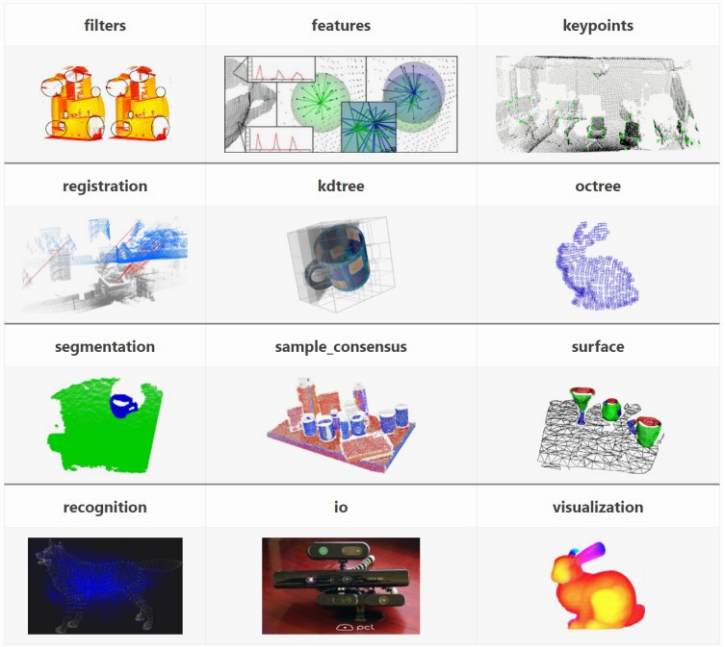
3- move data points according to transformation



4- 1st iteration result

Point Cloud Library (PCL)

- ”The Point Cloud Library (or PCL) is a large scale, open project for 2D/3D image and point cloud processing.
- The PCL framework contains numerous state-of-the art algorithms including:
  - filtering,
  - feature estimation,
  - surface reconstruction,
  - registration,
  - model fitting and
  - segmentation.
- These algorithms can be used, for example,
  - to filter outliers from noisy data,
  - stitch 3D point clouds together,
  - segment relevant parts of a scene,
  - extract keypoints and compute descriptors to recognize objects in the world based on their geometric appearance, and
  - create surfaces from point clouds and visualize them – to name a few.



- The PCL Visualization library is meant to integrate PCL with VTK by providing a comprehensive visualization layer for n-D point cloud structures.

https://pointclouds.org/
https://pointclouds.org/assets/pdf/pcl_icra2011.pdf