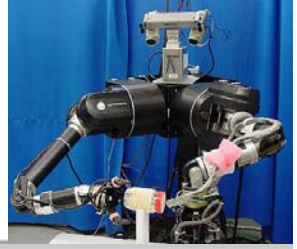
Einführung in Visual Computing

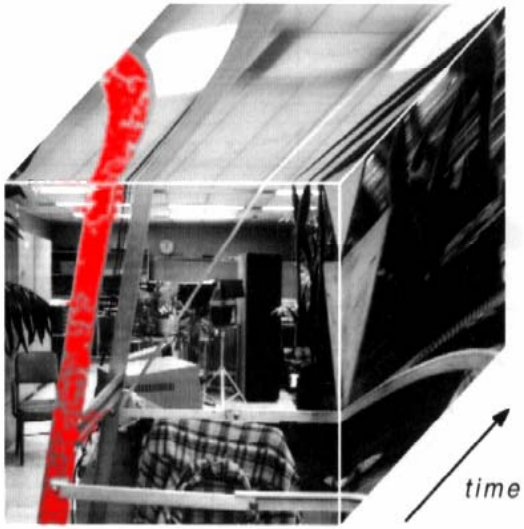
Unit 18: Stereo and Motion

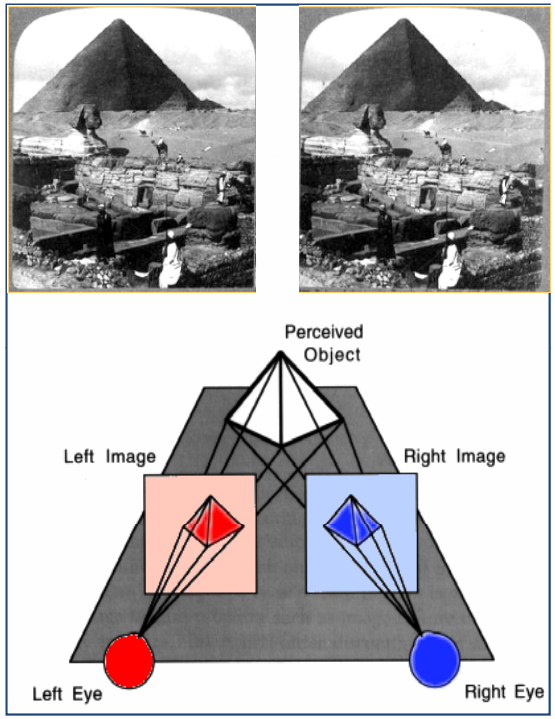


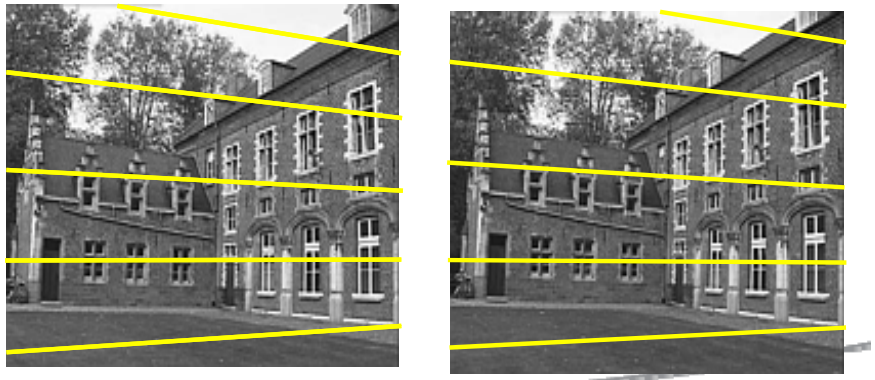
http://www.caa.tuwien.ac.at/cvl/teaching/sommersemester/evc

Content:

- Introduction to Stereo Vision
- Stereo Geometry
- Epipolar Lines
- Correspondence Problem
- Area-Based Stereo Matching
- Feature Based Stereo Matching
- Structure from Motion
  - Motion Field
  - Motion Vector
  - Optical Flow

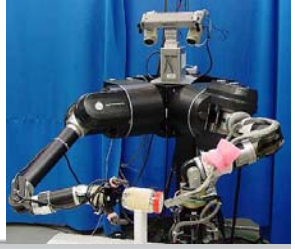






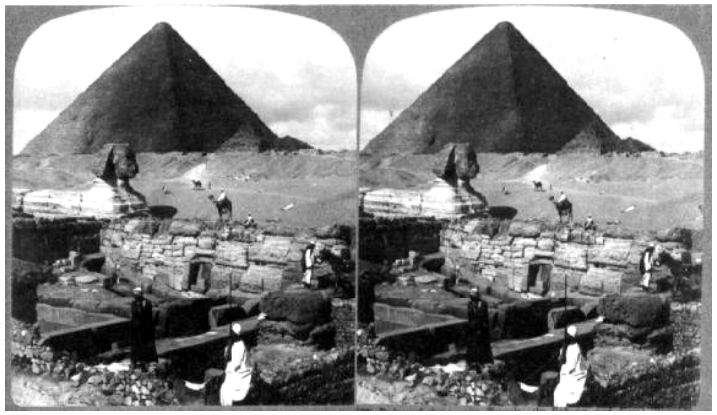
Stereo Vision

Stereo Vision

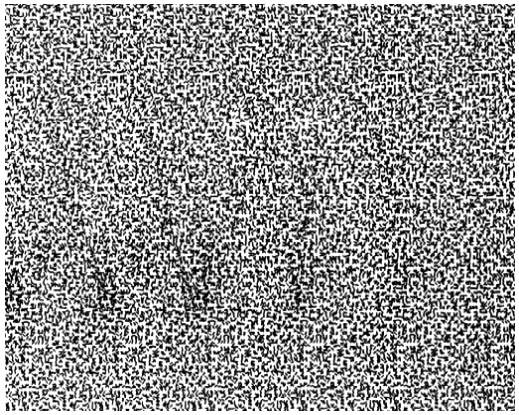


2 dim + 2 dim + geometry => 3 dim

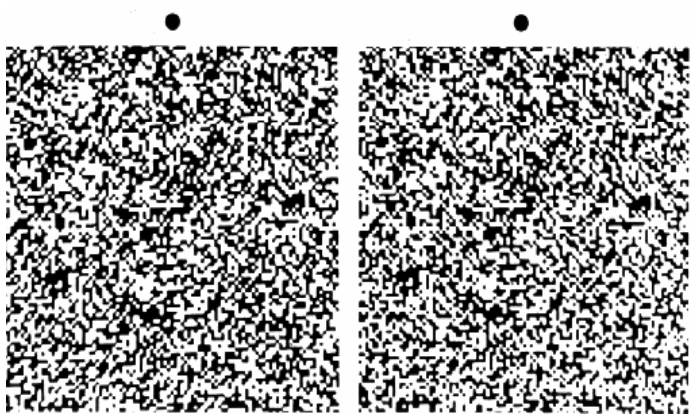
- Correct 3d information using 2d images:
  - 2 or more images taken from different geometric positions plus geometric calibration of camera
  - Tries to imitate human visual system
  - Is also used in the entertainment industry



Stereo photographs

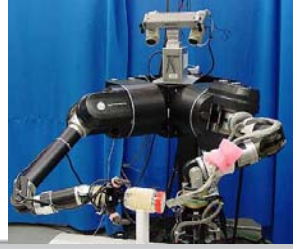


Auto stereogram

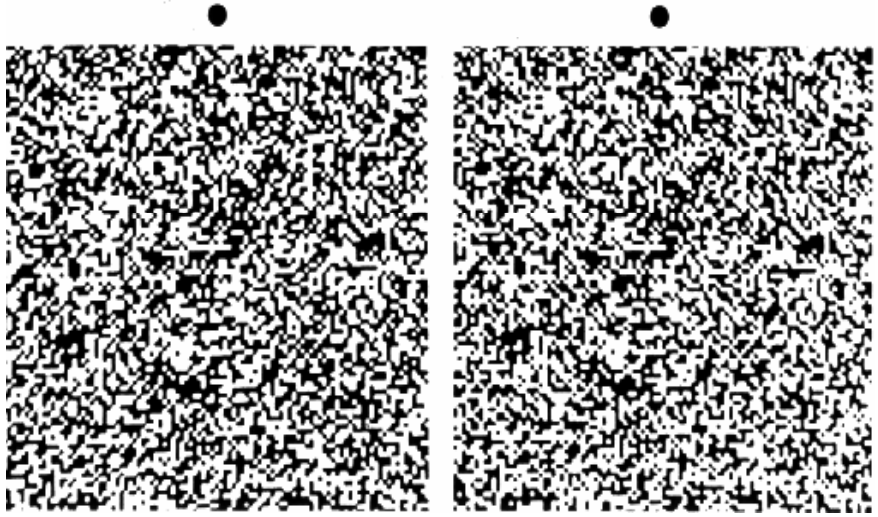


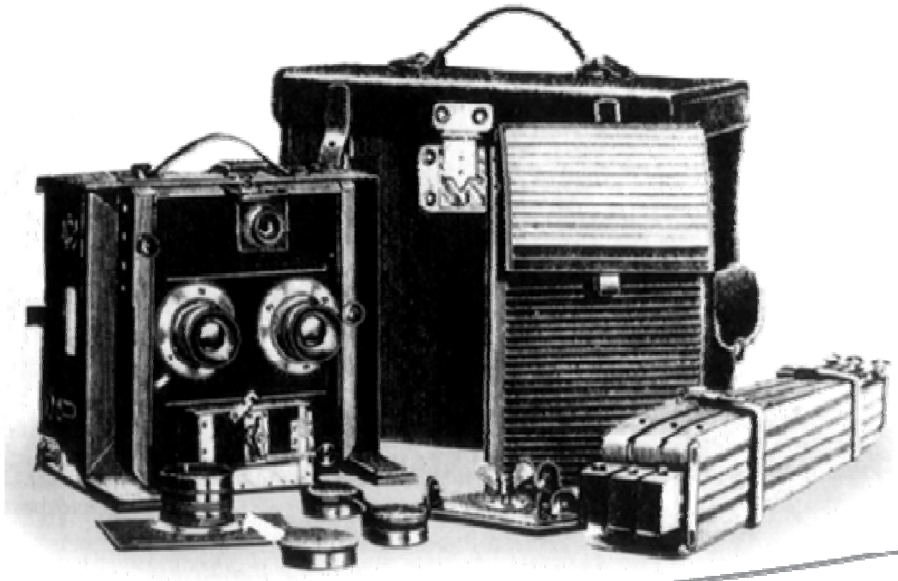
Random Dot Stereogram

Stereo Vision



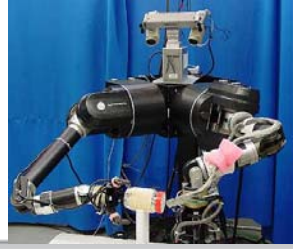
- 3D vision is a natural product of our visual perception:
  - Learned capability
  - Is learned in the first month in life: Hand - eye problem
  - Stereo emerges in the brain => proof: Random Dot Stereograms
  - Also in entertainment industry and art
  - Around 1890 first stereo camera



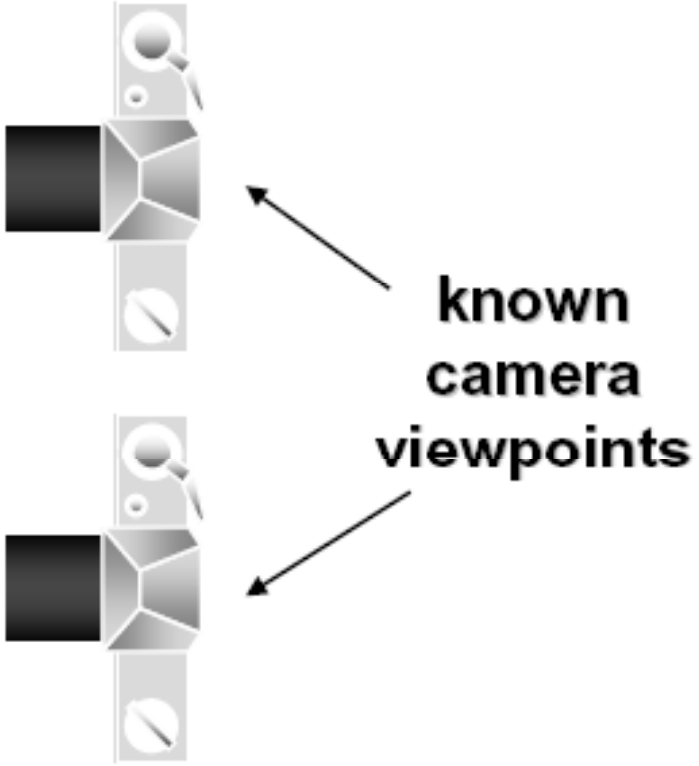


Mark Twain at Pool Table", no date, UCR Museum of Photography

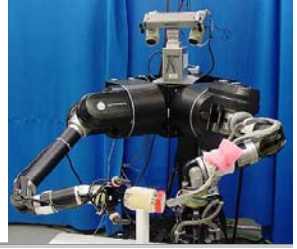
Stereo Reconstruction



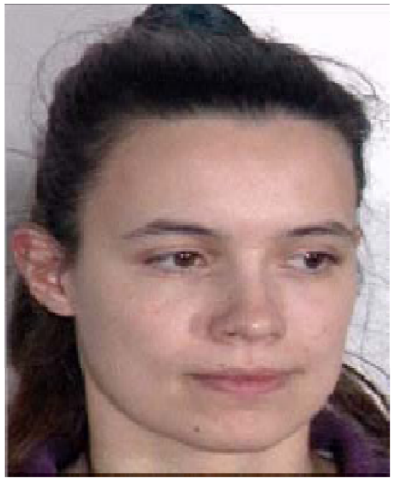
- Shape (3D) from two (or more) images

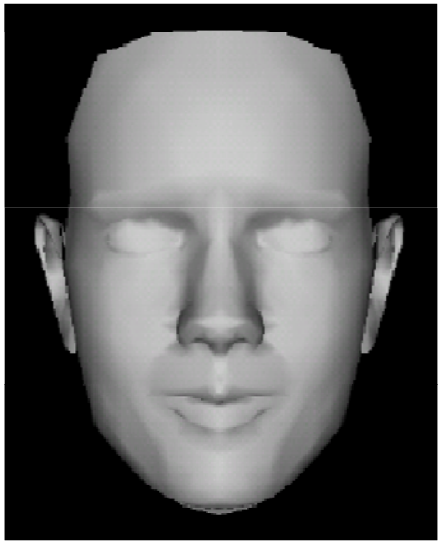


Example



images



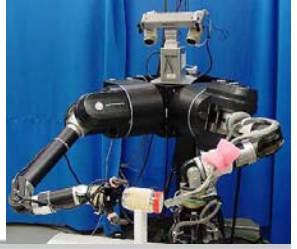


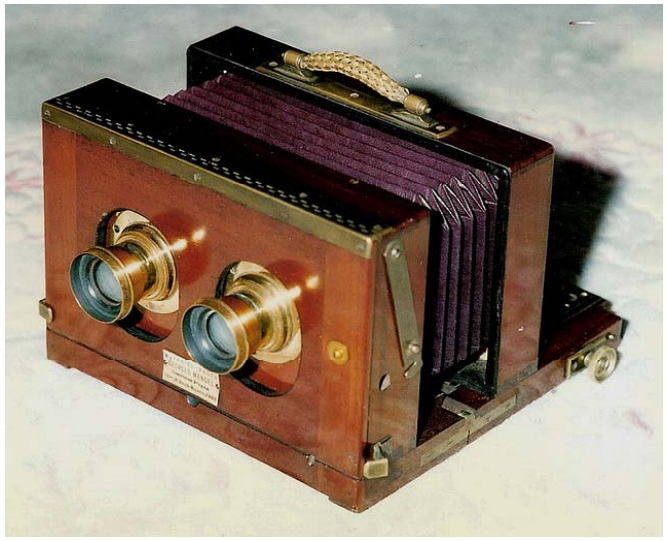
shape

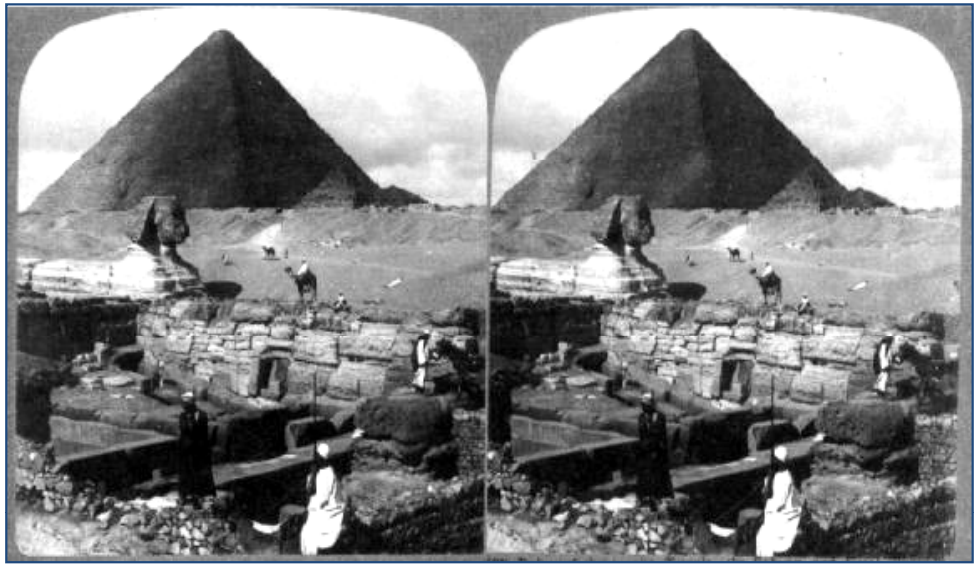
surface
reflectance

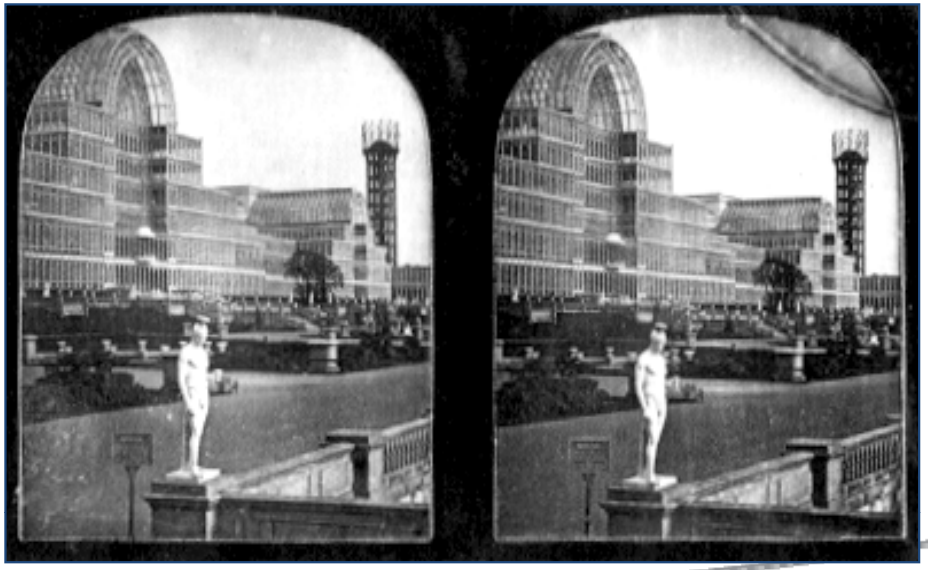
Stereoscopy

Stereo Camera: 1890

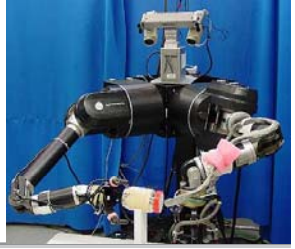




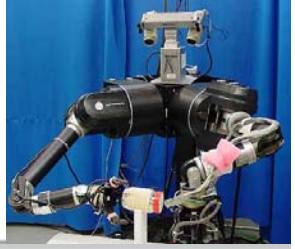




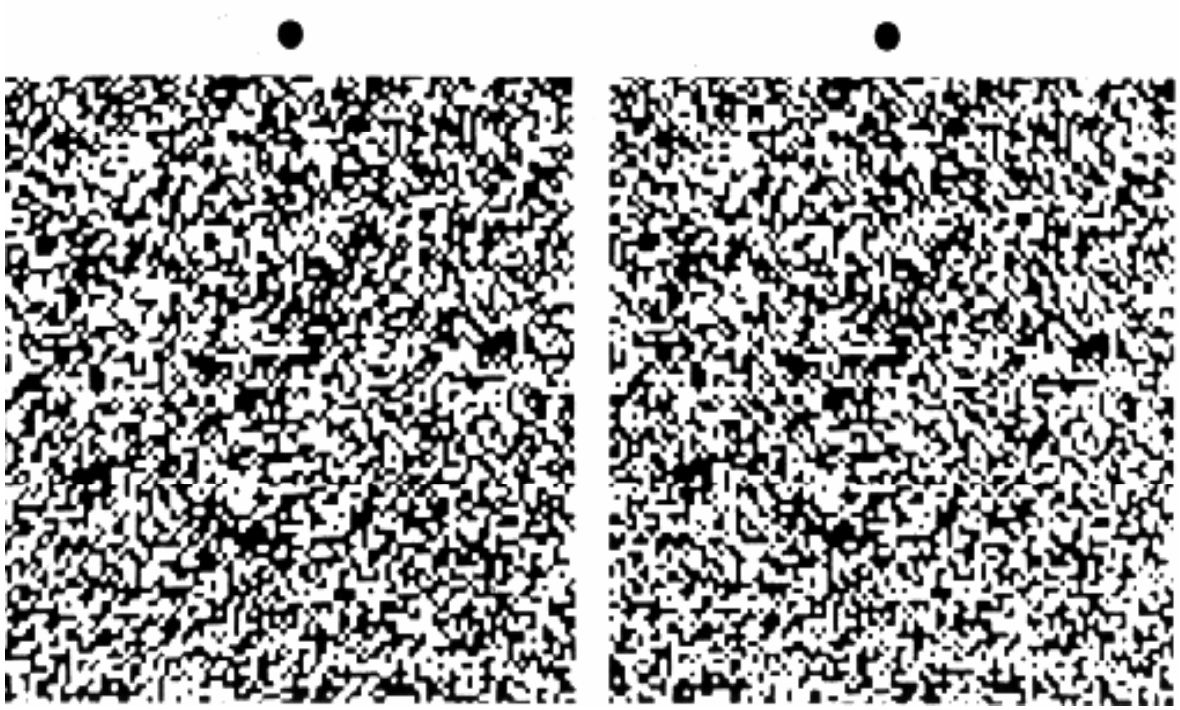
Stereo Camera: 120 years later



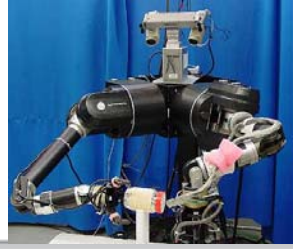
Random Dot Stereograms



- 1959: Evidence that human brain generates 3D impressions by first random dot stereogram [Julez59]
  - Revolutionized psychology of perception
  - „modern“ from time to time

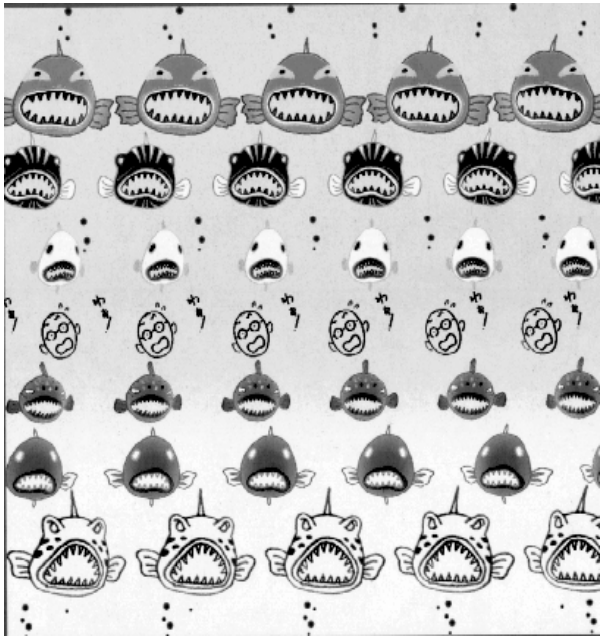


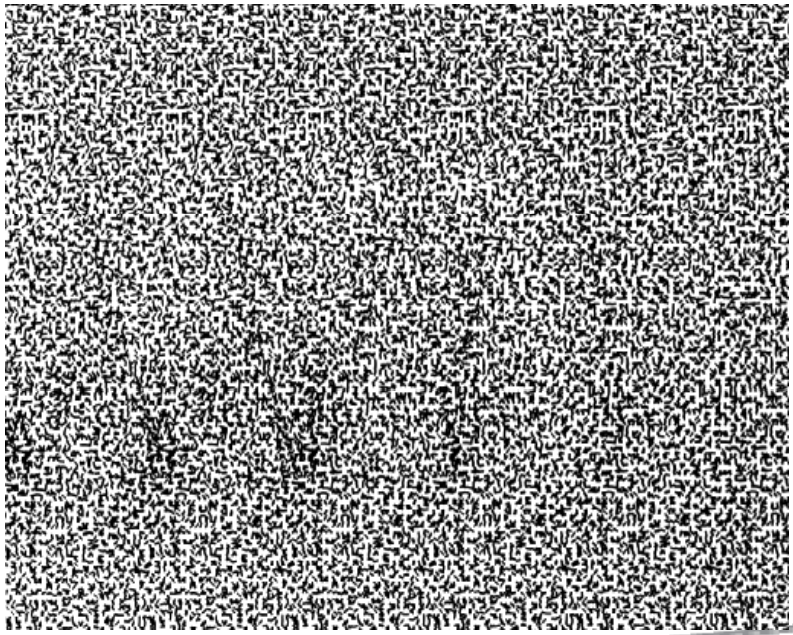
Autostereograms



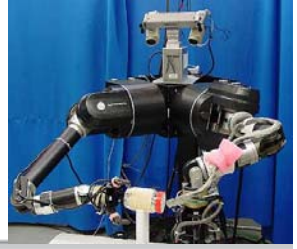
For some people Random Dot Stereograms are complicated to view: Autostereogramme [Tyler79]:

- Also called single image Stereogram
- Form of Art
- Is formed by repetition of patterns in specific intervals



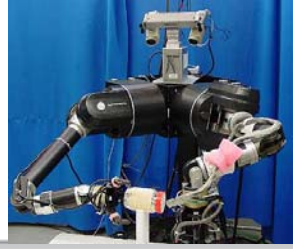


Stereoscopes

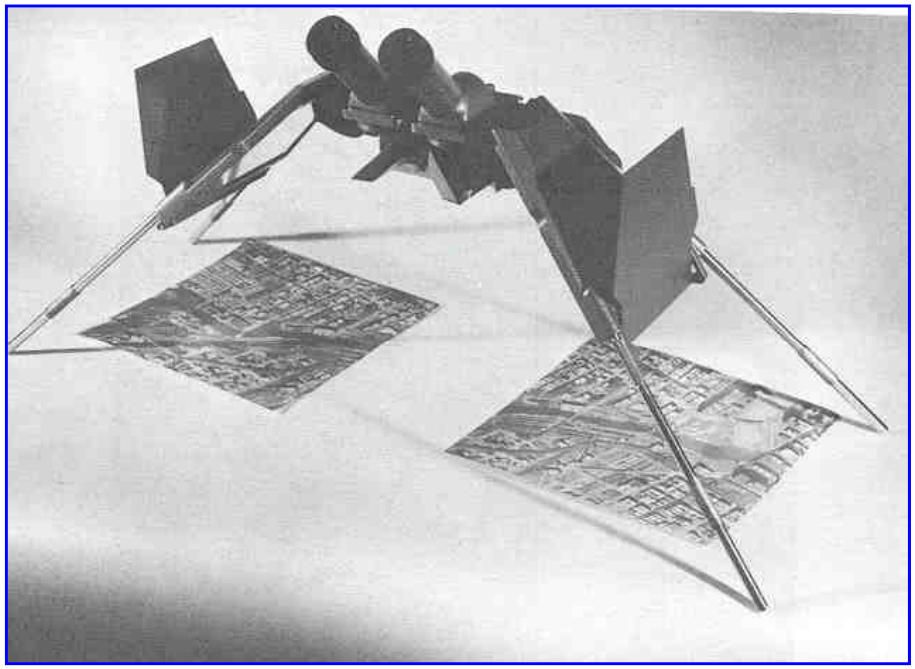


- A binocular optical instrument that helps us view two properly oriented photos to
- obtain a 3-dimensional model.
- Lens (pocket) stereoscope
  - Simplest
  - Least expensive
  - Small
  - 2-4 x magnification

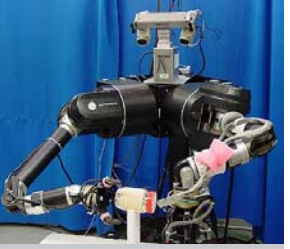
Stereoscopes

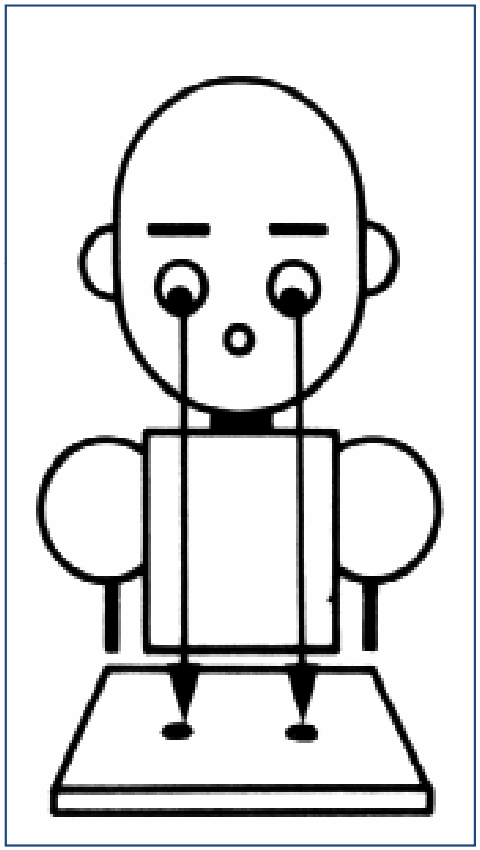


- Mirror stereoscope
  - Photos can be placed separately for viewing



Stereograms – Parallel Technique

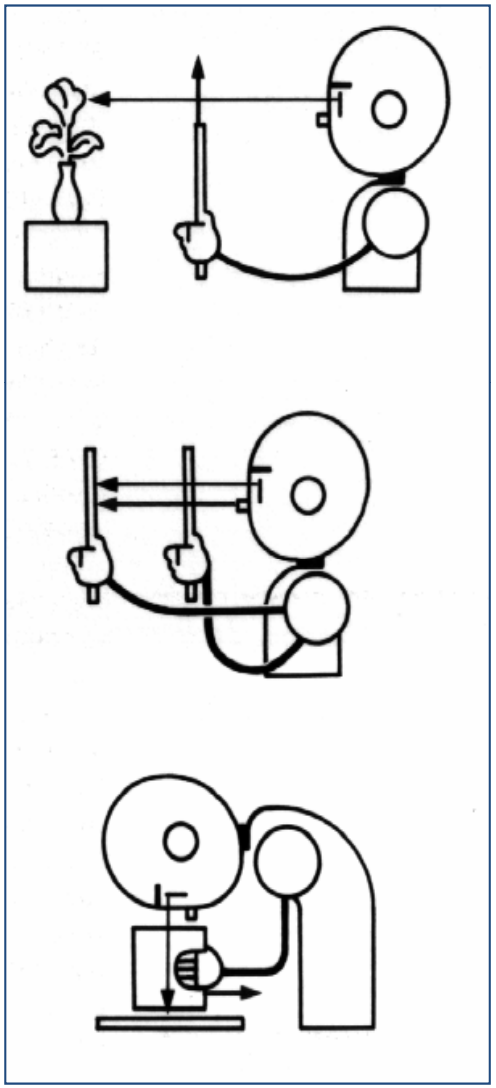


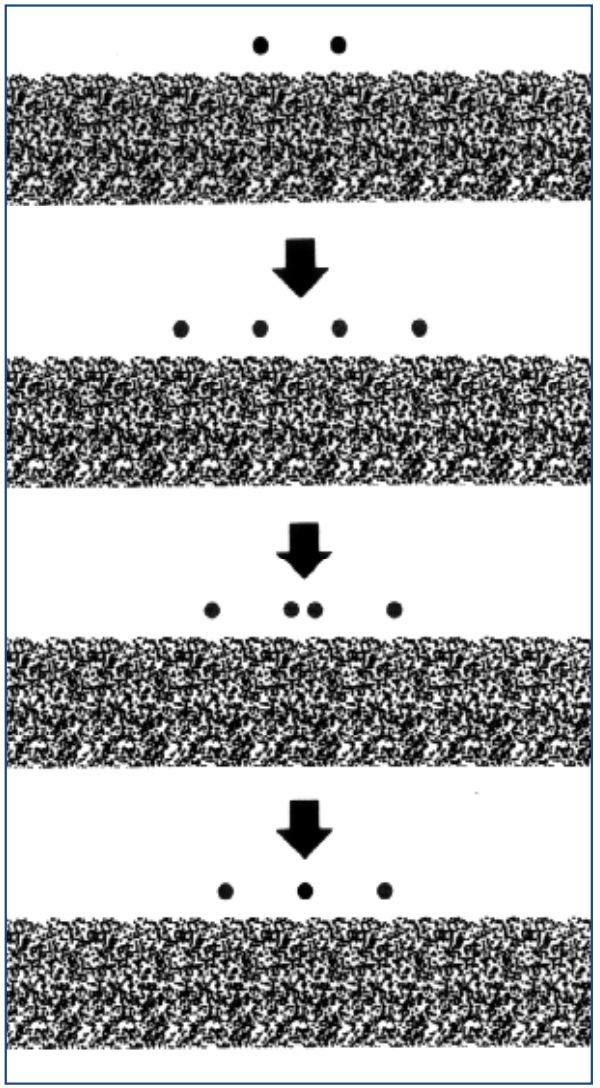


Insertion-technique

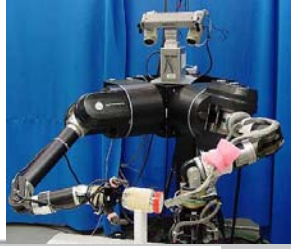
Tip of nose-technique

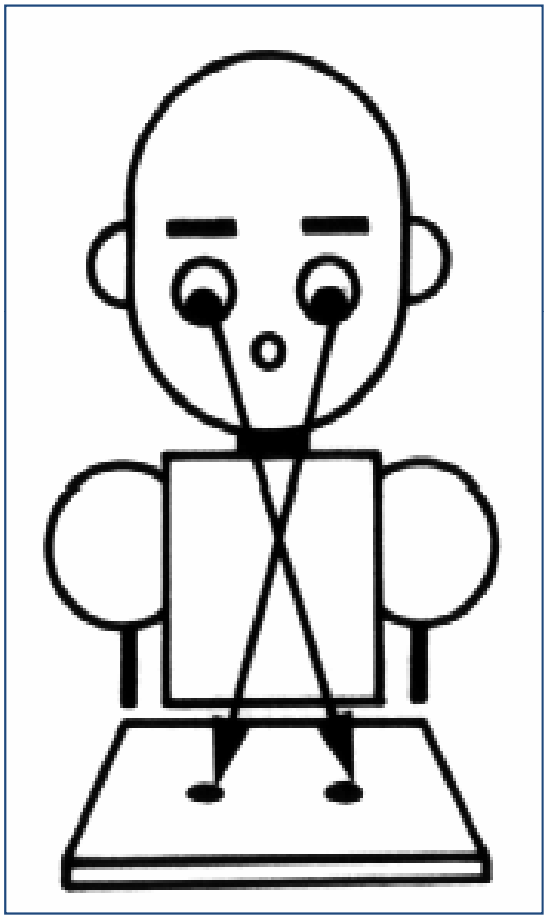
Separation-technique

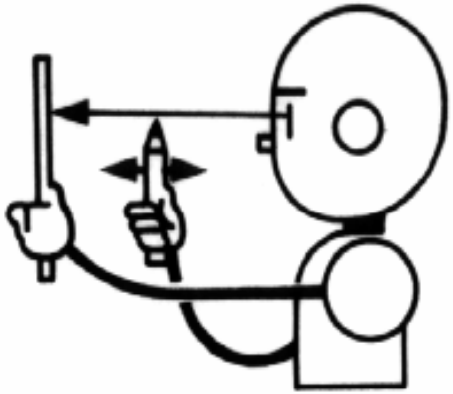




Stereograms – Cross-eye Technique

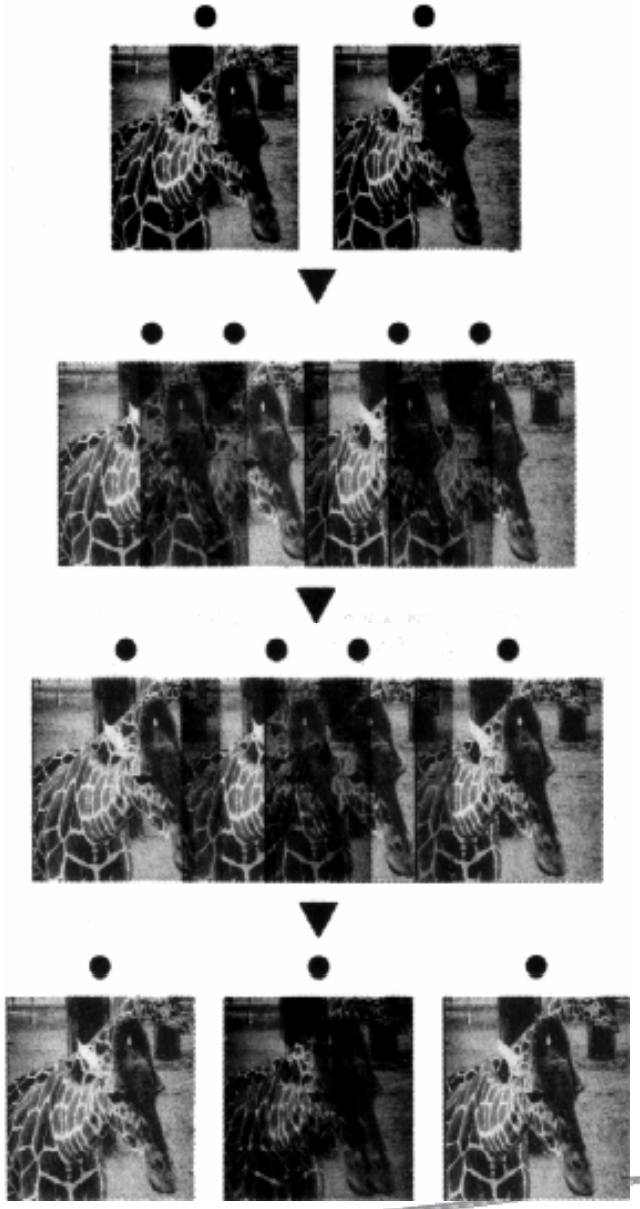






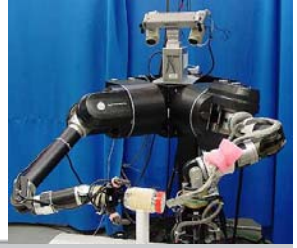
Pencil technique

Ring technique

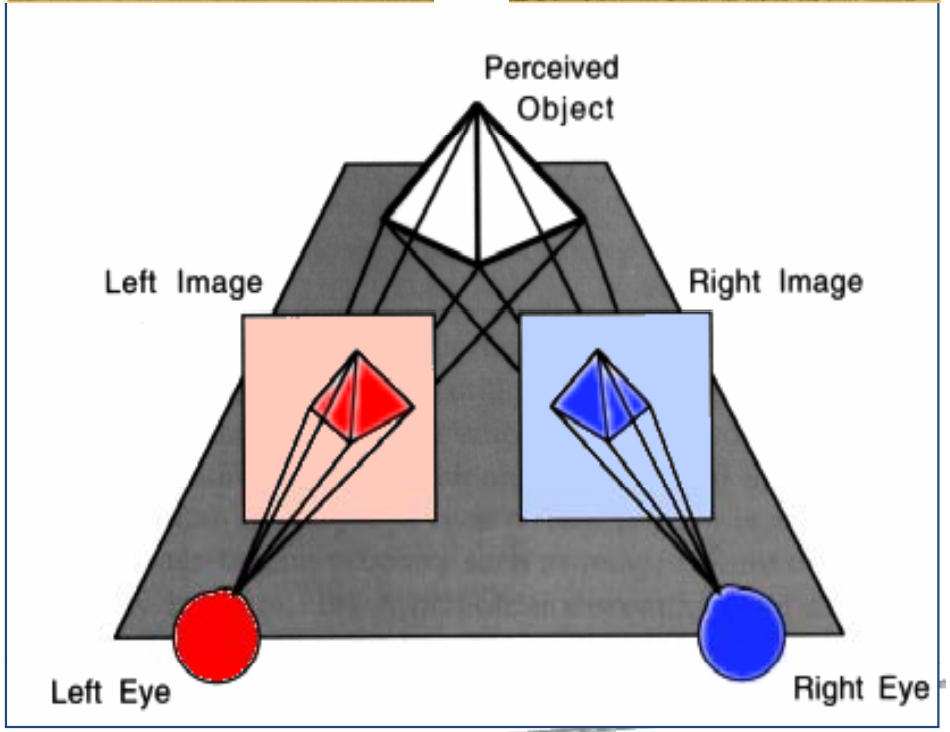


Stereo Analysis

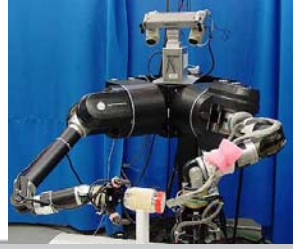
Stereo Analysis



- Using two views of an object from two different positions enables to reconstruct 3d shape of the object out of the geometric differences of the two 2d projections:
  - two 2d images
  - geometric relation between cameras

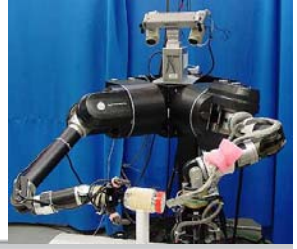


Challenges

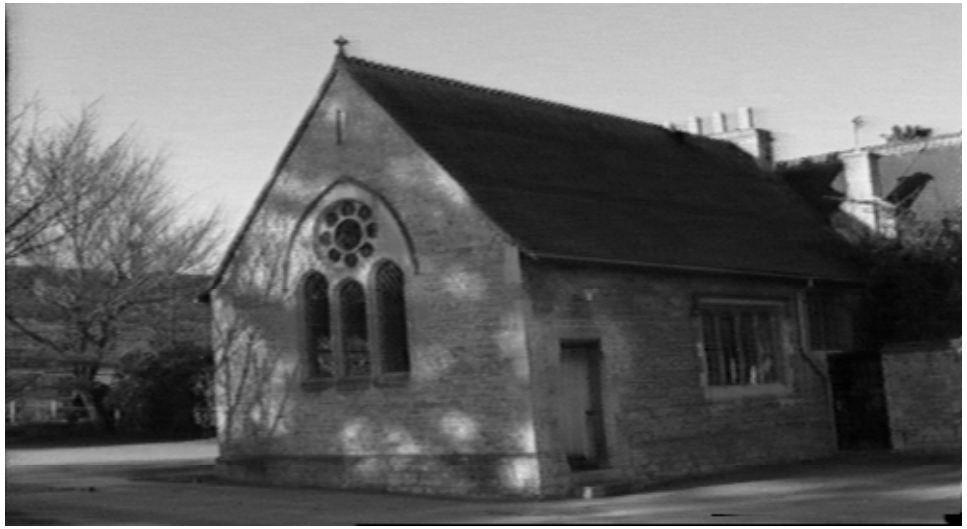


- Ill-posed inverse problem
  - Recover 3-D structure from 2-D information
- Difficulties
  - Uniform regions
  - Half-occluded pixels

The Objective

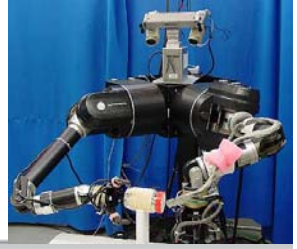


- **Given** two images of a scene acquired by known cameras compute the **3D position of the scene** (structure recovery)



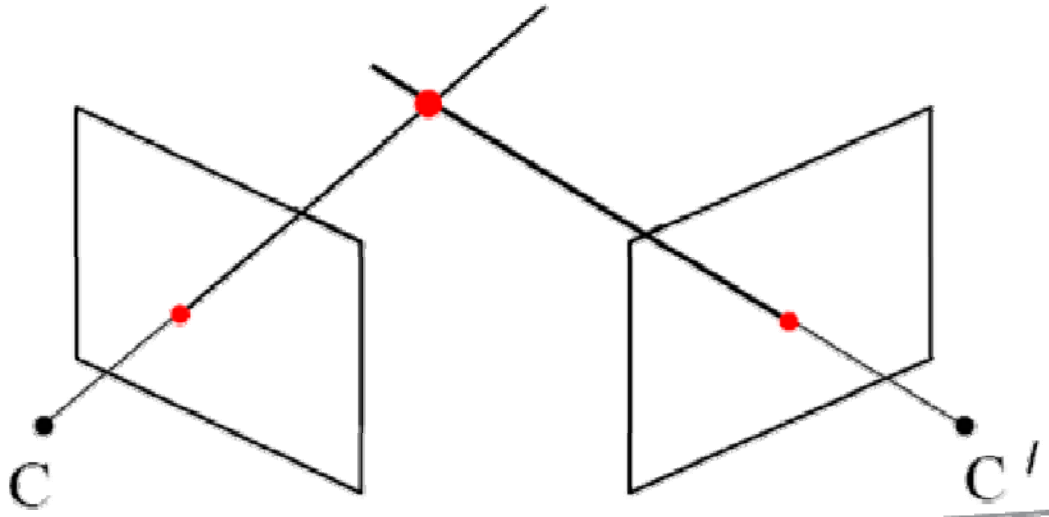
- Basic principle: triangulate from corresponding image points
  - Determine 3D point at intersection of two back-projected rays

The Solution

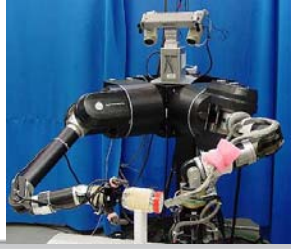


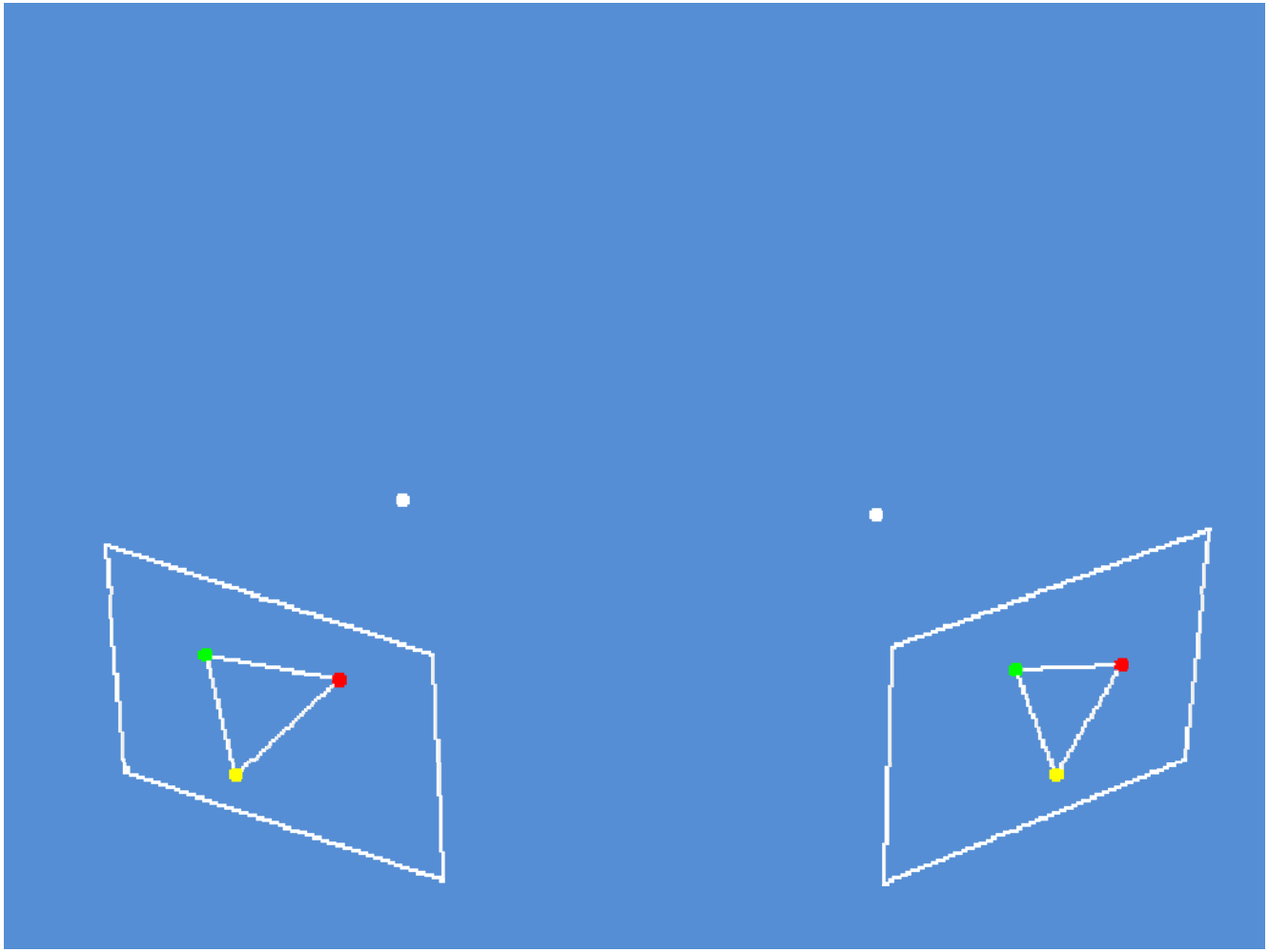
- Corresponding points are images of the same scene point

Triangulation

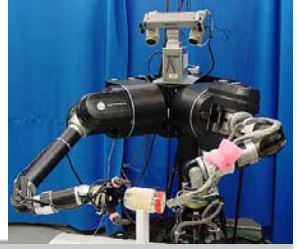


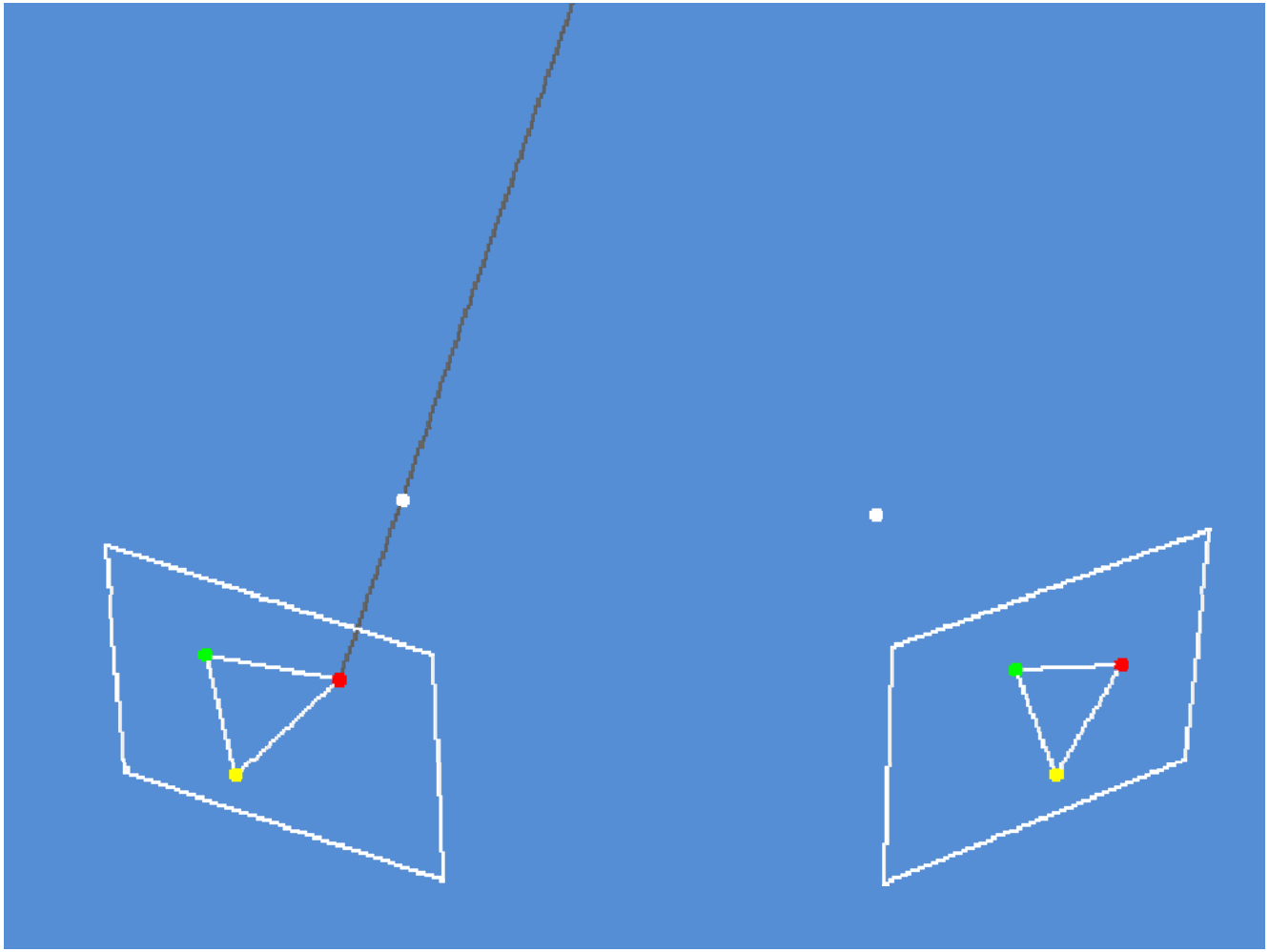
Fundamentals of Stereo Vision



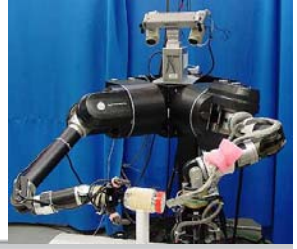


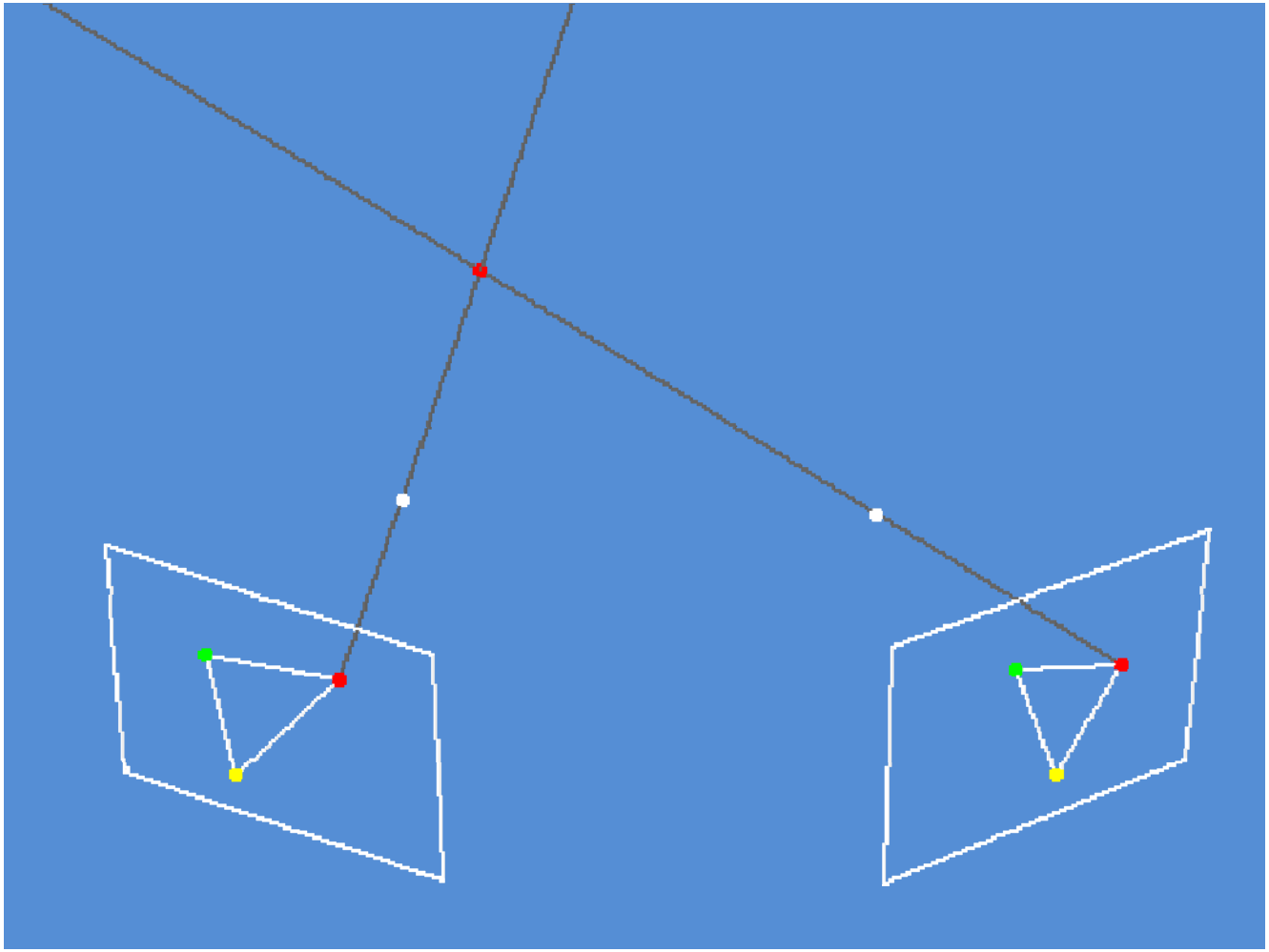
Fundamentals of Stereo Vision



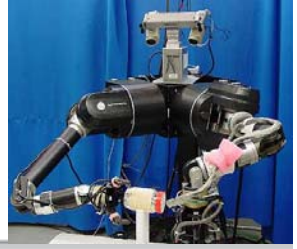


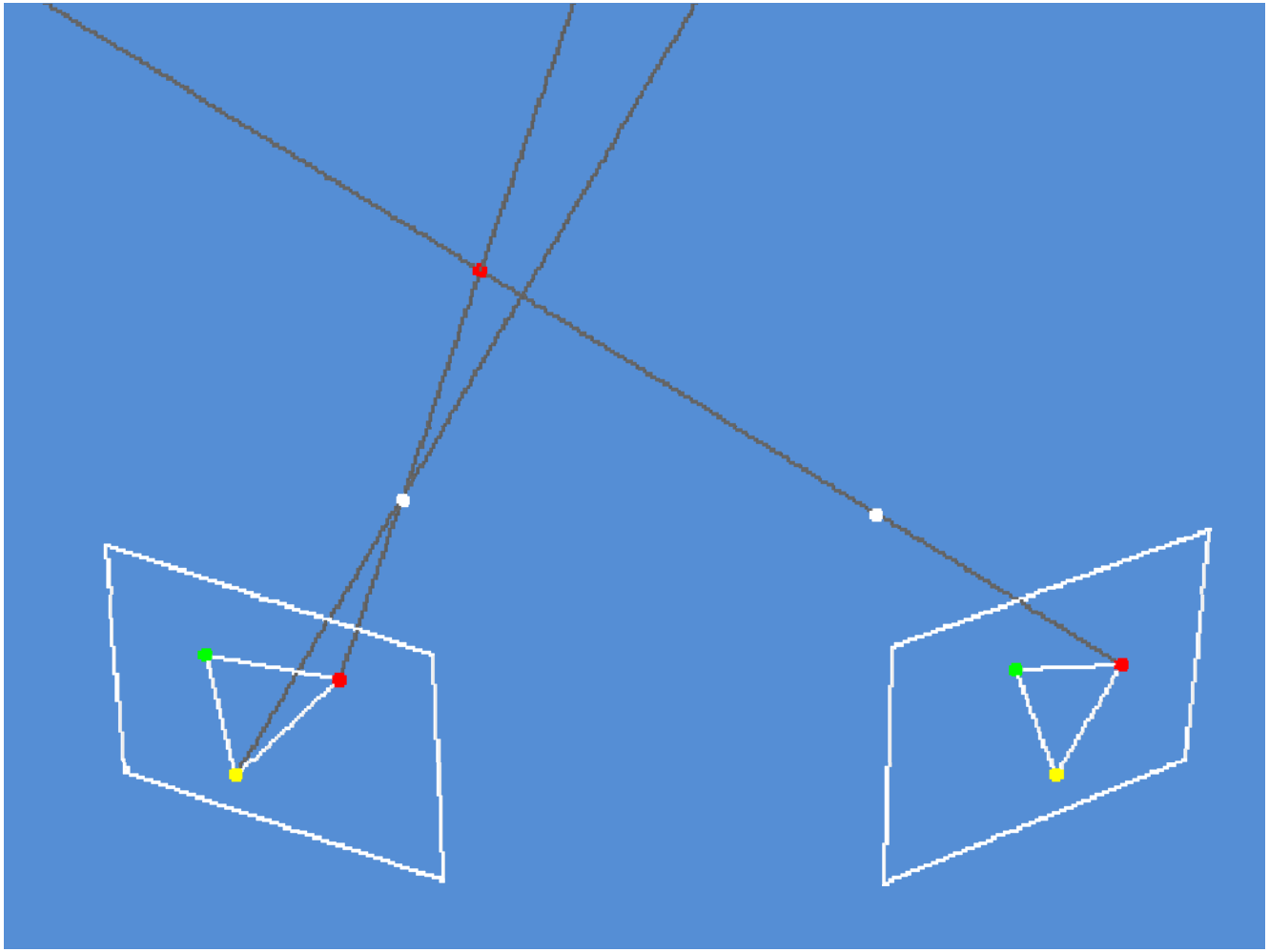
Fundamentals of Stereo Vision



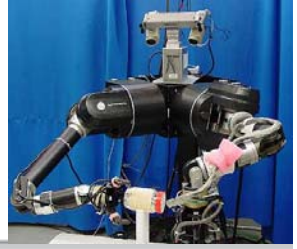


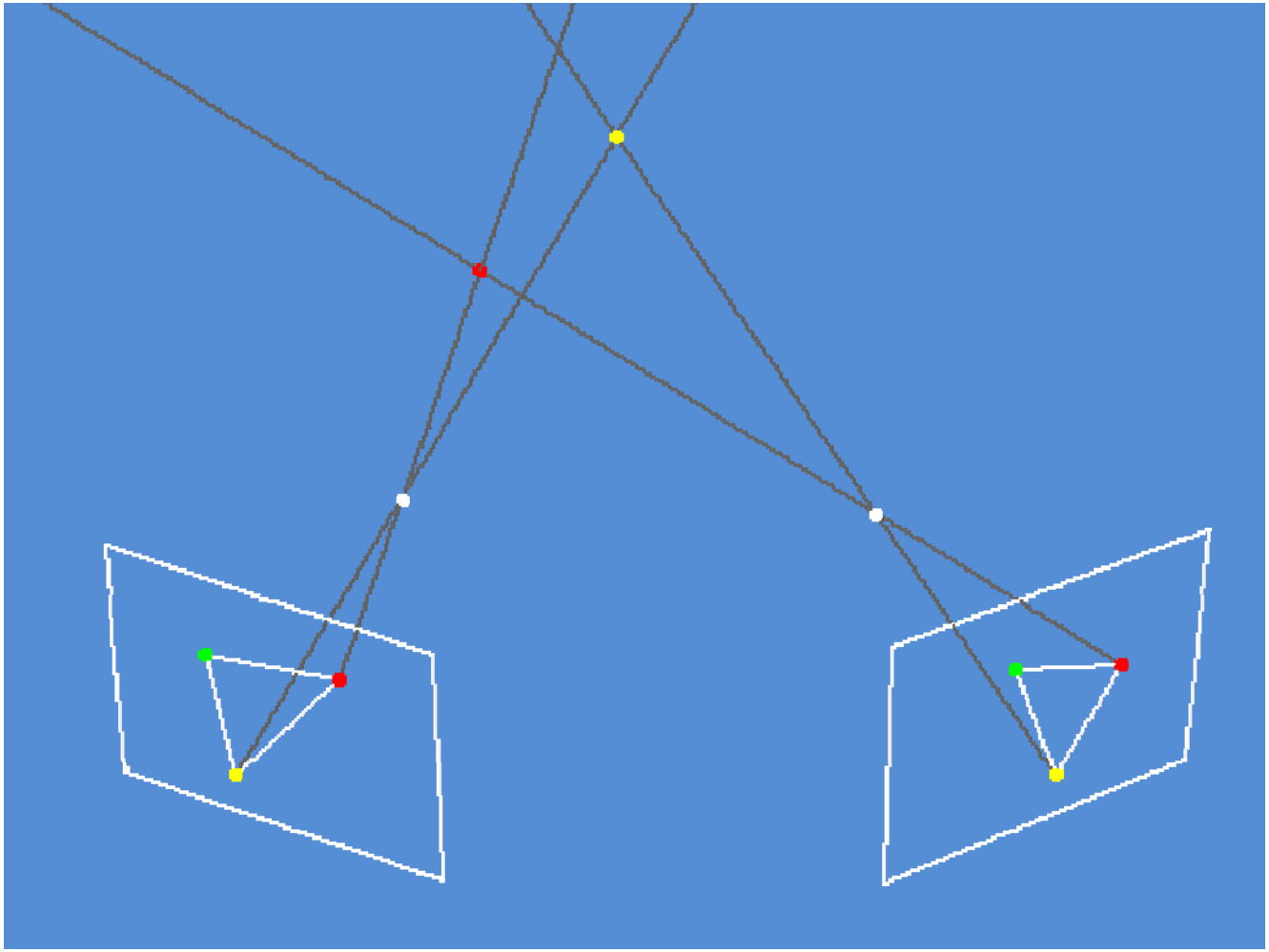
Fundamentals of Stereo Vision



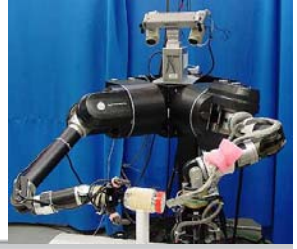


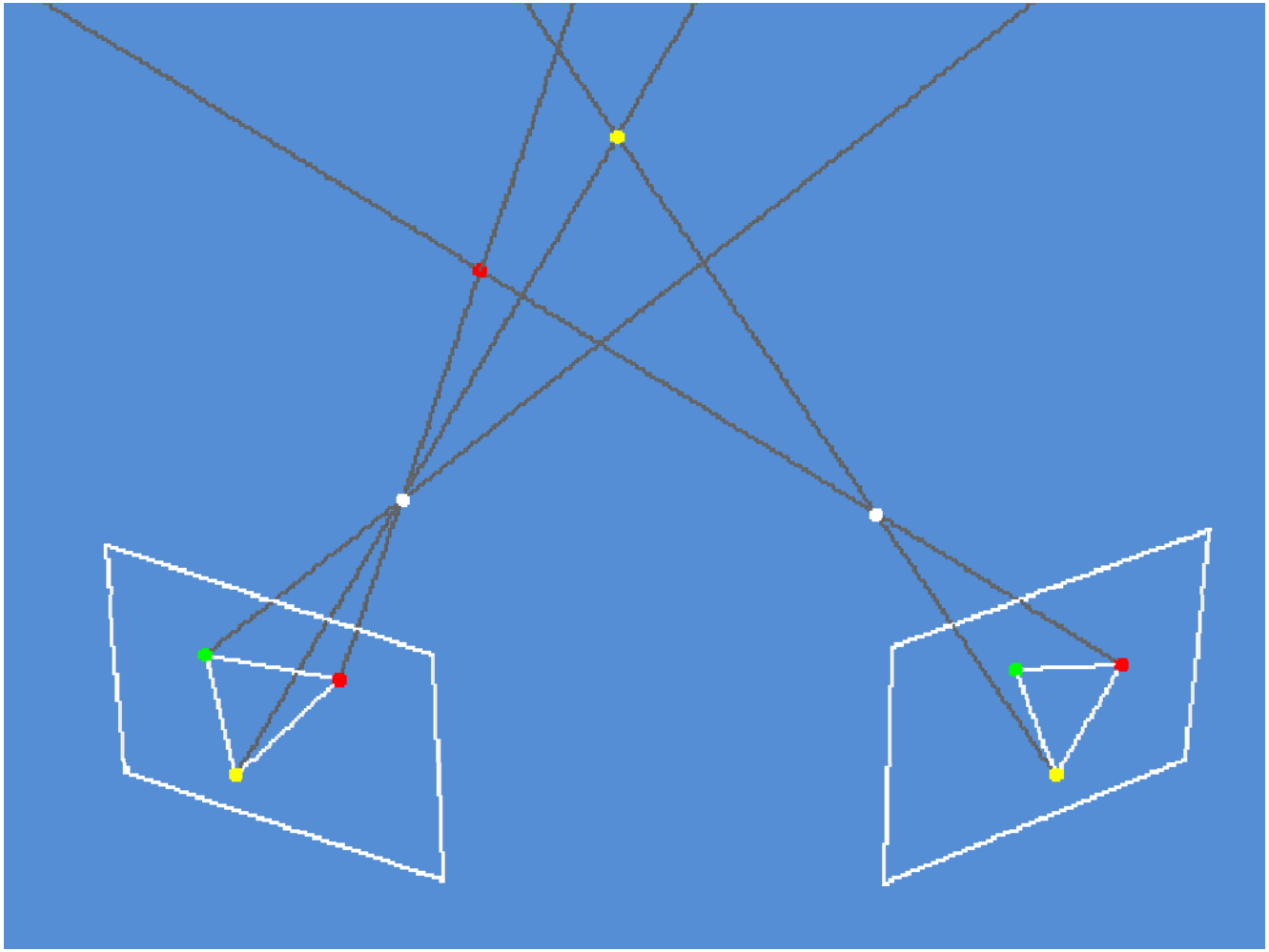
Fundamentals of Stereo Vision



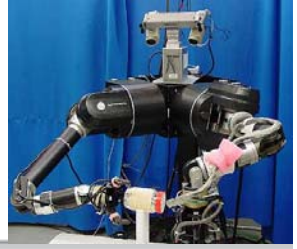


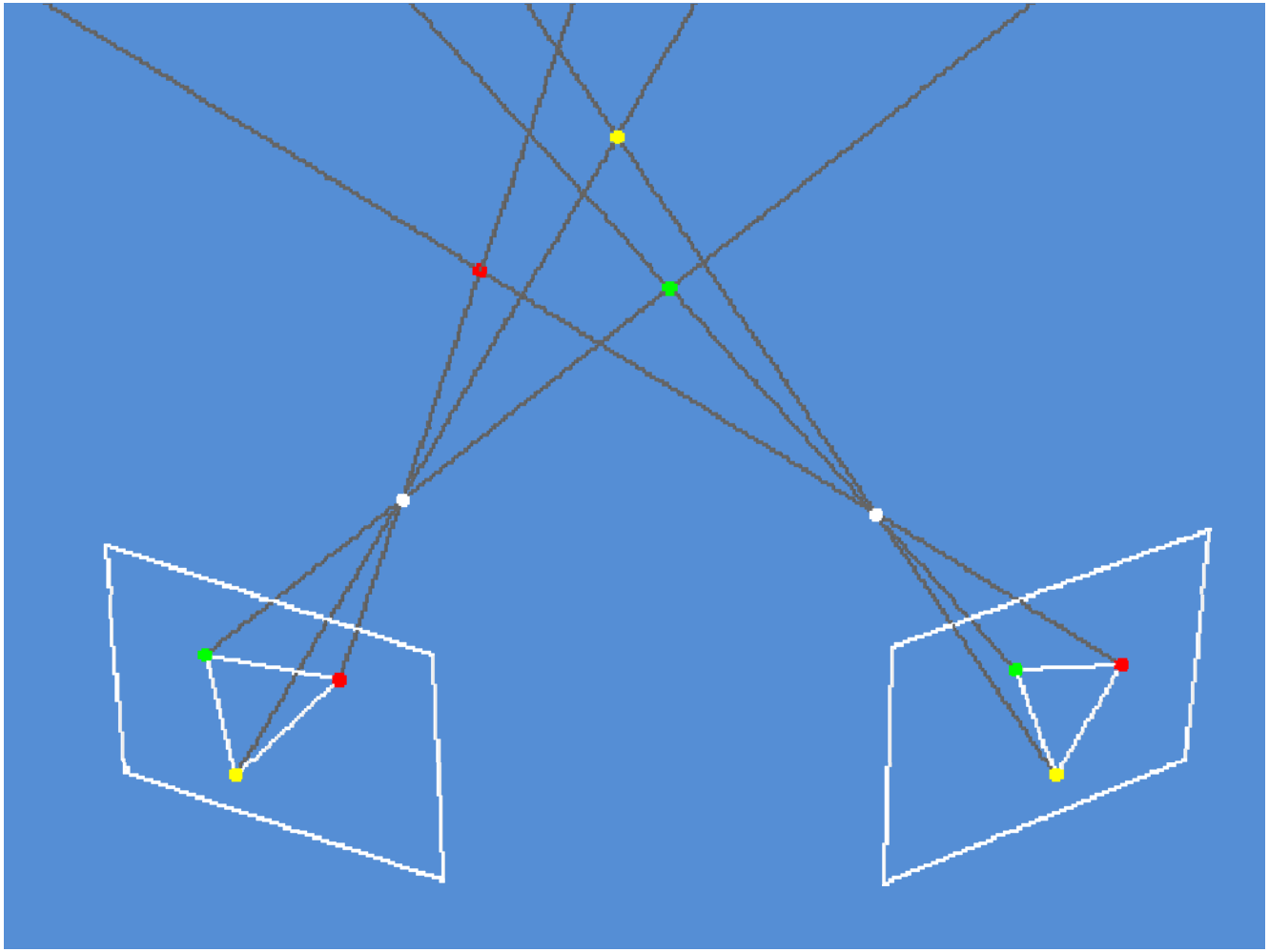
Fundamentals of Stereo Vision



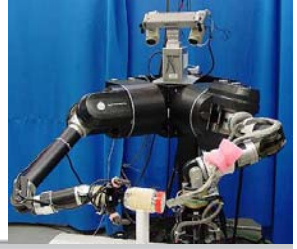


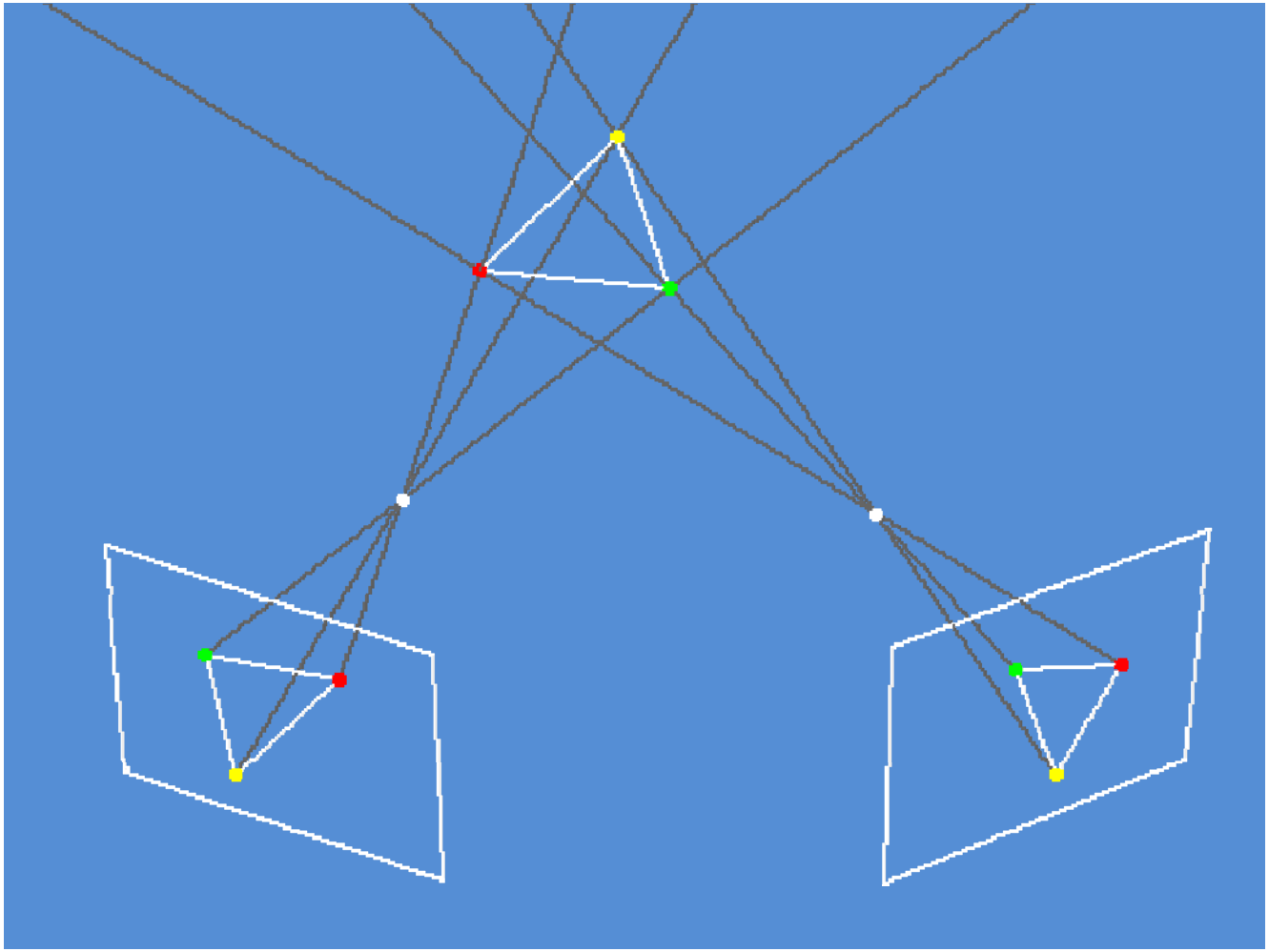
Fundamentals of Stereo Vision



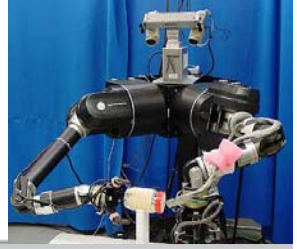


Fundamentals of Stereo Vision

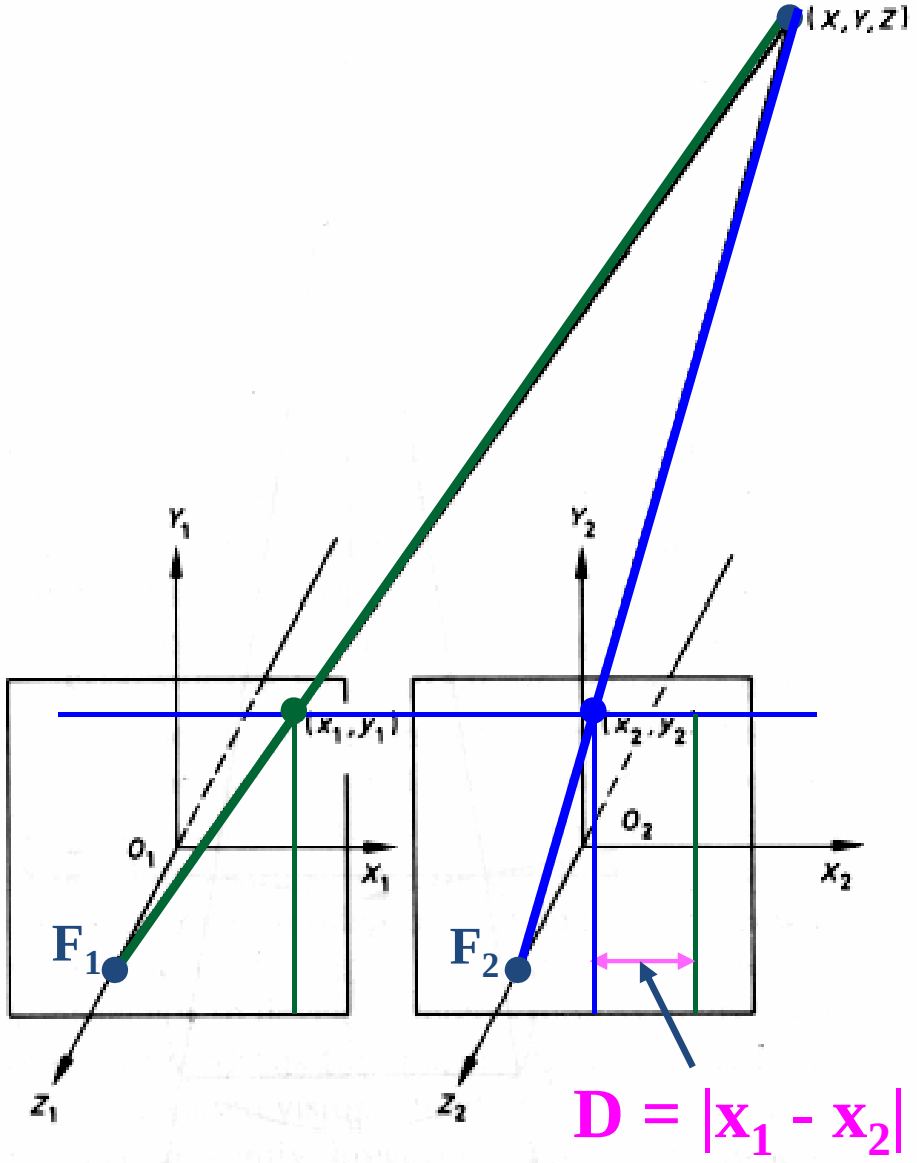




Stereo Geometry

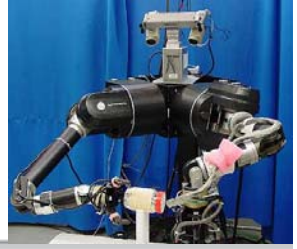


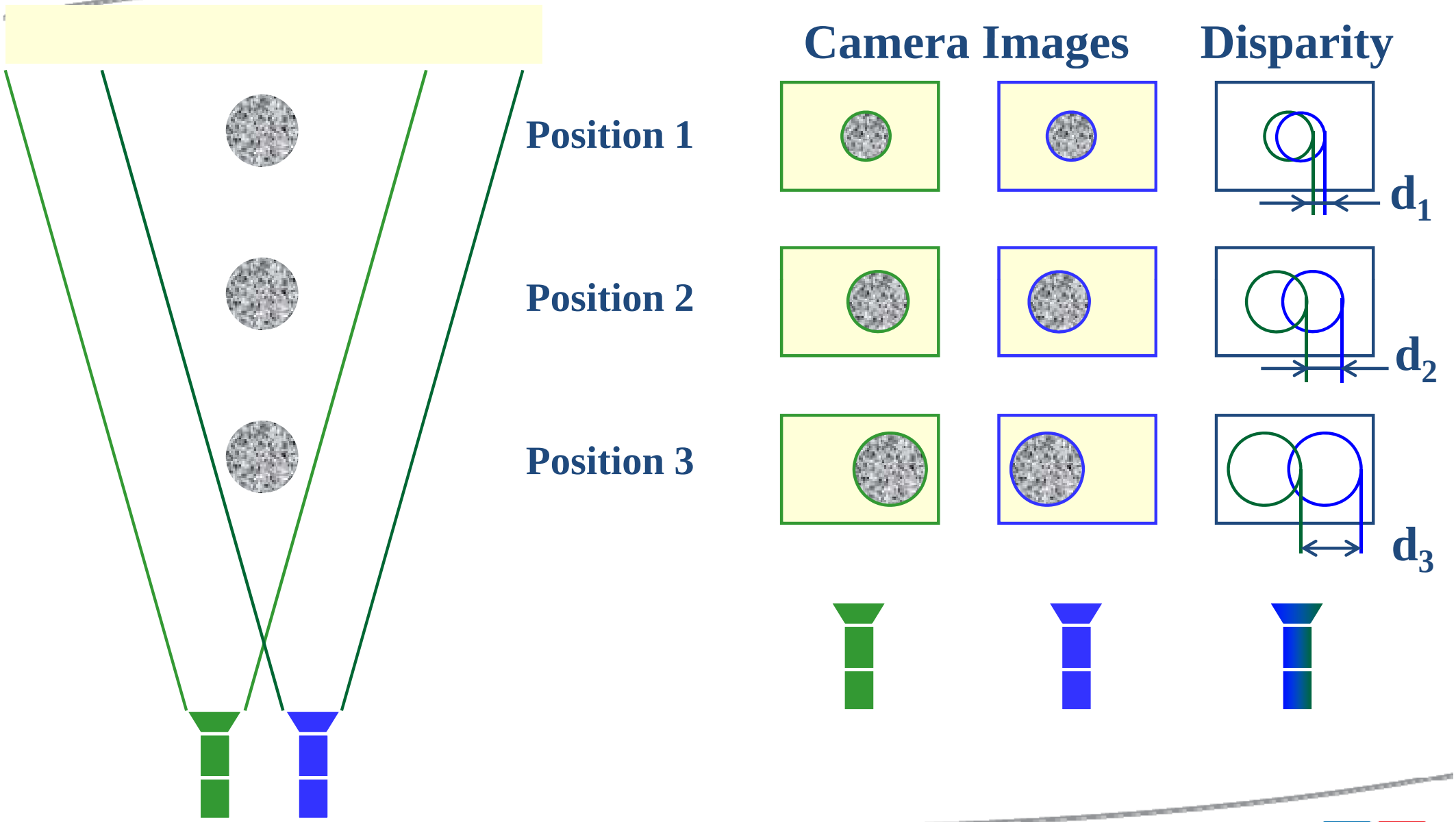
- Perspective **projections** of object points
- Specific case: both image planes are parallel => **normal case**
- Using **geometric** position of images planes and object **projection** depth information is computed



Disparity

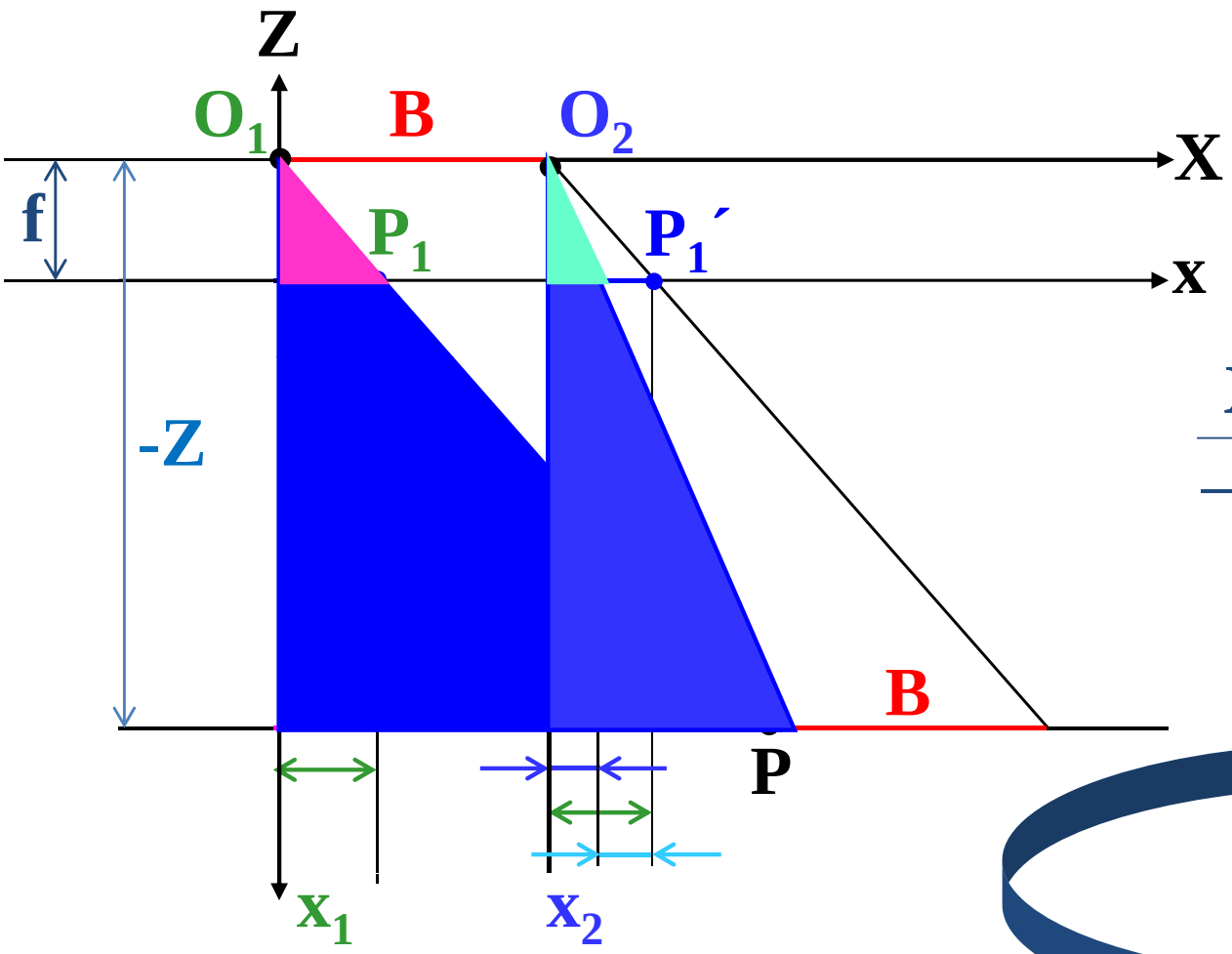
Disparity




Camera Images
Disparity
Position 1
Position 2
Position 3
d_1
d_2
d_3

Stereo Analysis

F: center of projection
X: world coordinate
x: image coordinate
f: focal length
Z: distance to center of projection
B: basis, distance between centers of projection



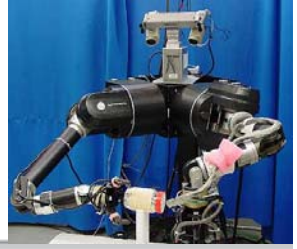
$$\frac{X}{-Z} = \frac{x_1}{f} \qquad \frac{X-B}{-Z} = \frac{x_2}{f}$$

$$X = -Z\frac{x_1}{f} \qquad X = B - Z\frac{x_2}{f}$$

$$-Z = \frac{f\,B}{x_1 - x_2}$$

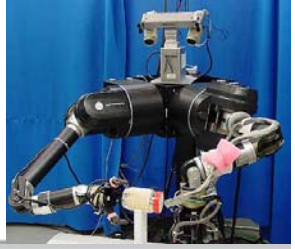
Epipolar Geometry

Epipolar Geometry

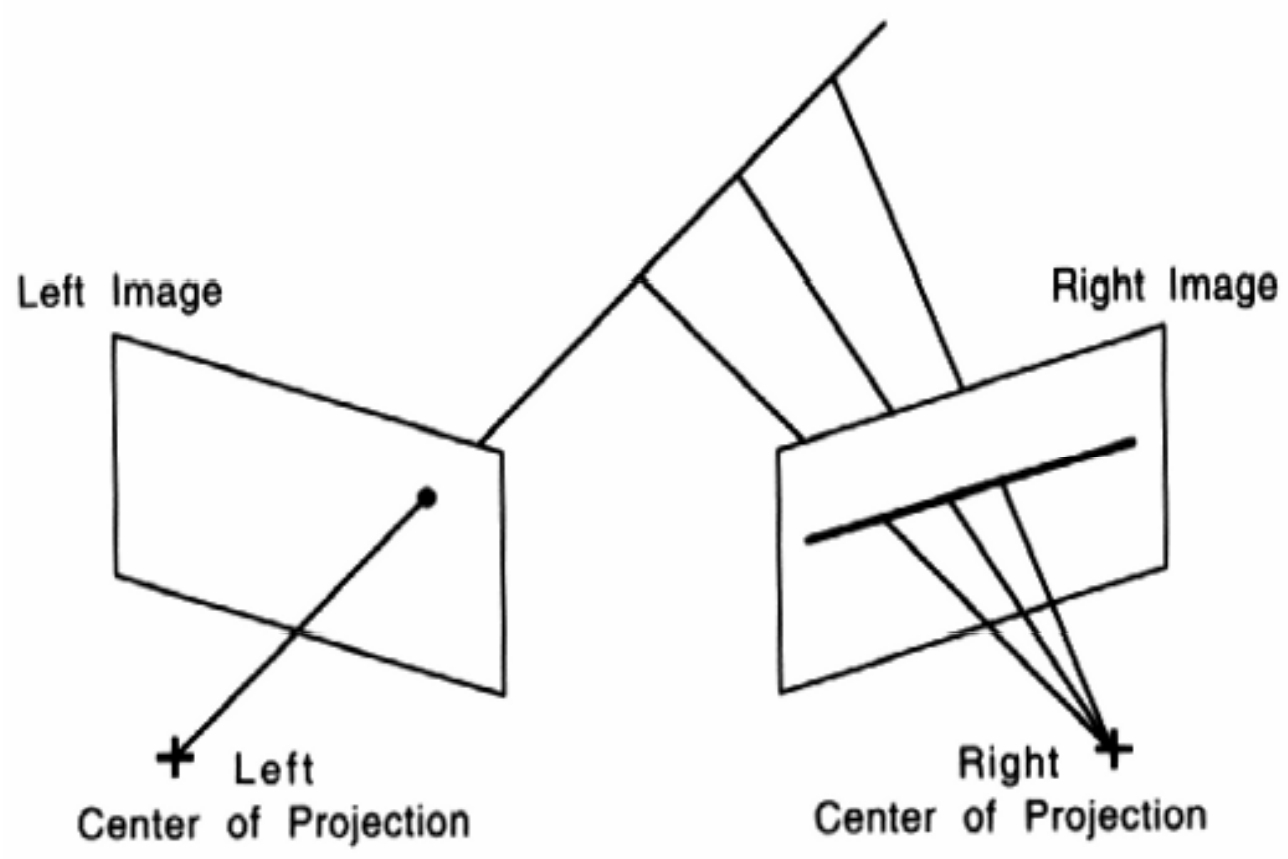


- Epipolar constraint
  - Reduces correspondence problem to 1D search along an epipolar line

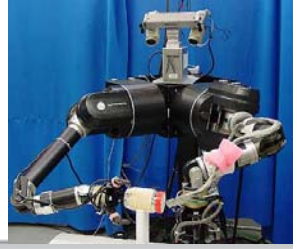
Epipolar Line



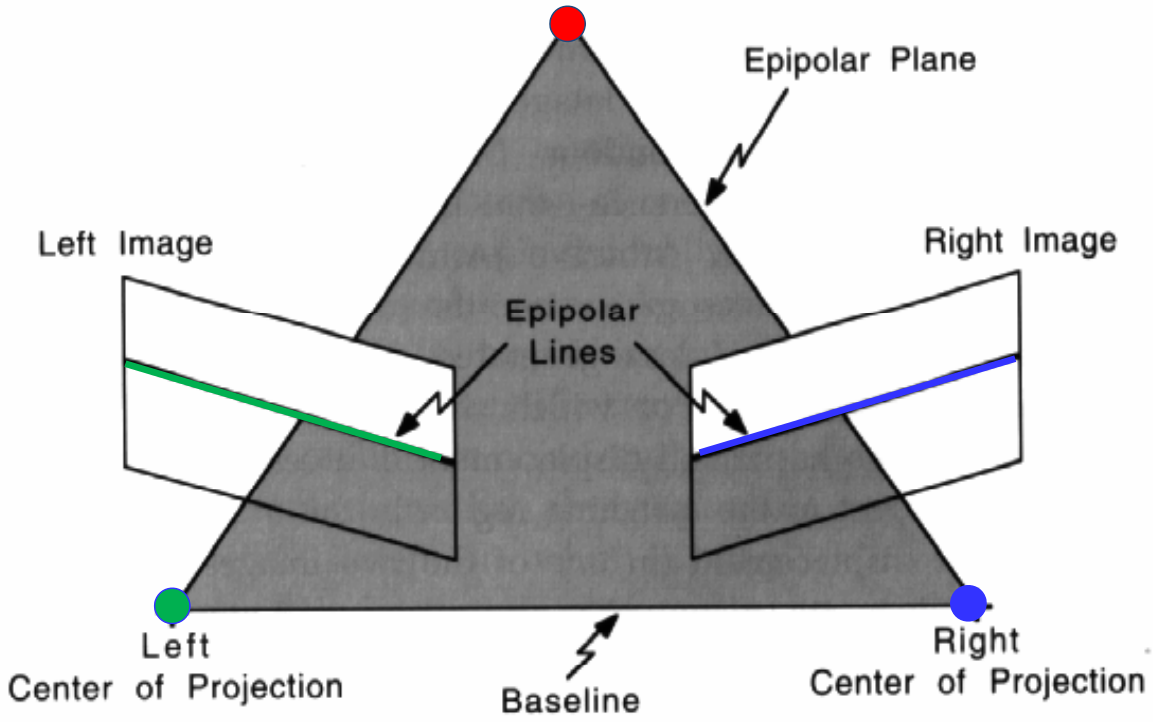
- Epipolar Constraint: Each point of the left image can lie only on a specific line in the right image: the Epipolar Line



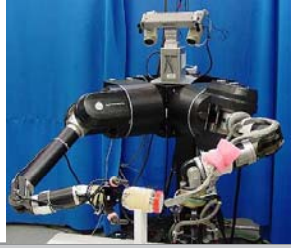
Epipolar Geometry

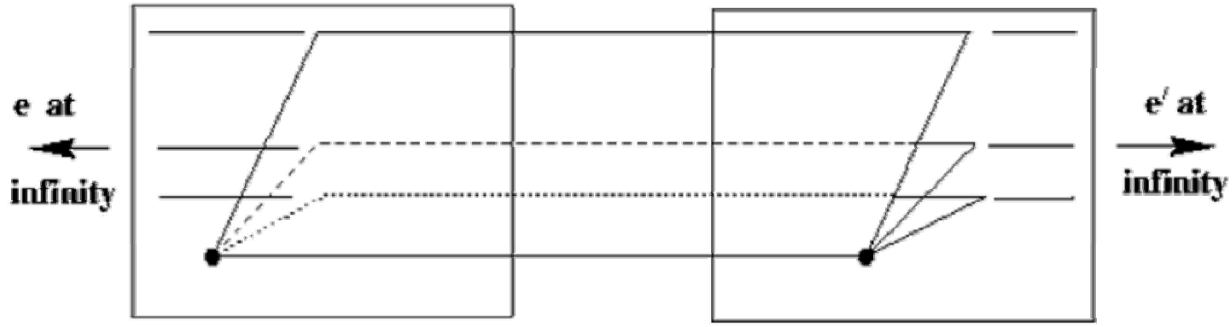


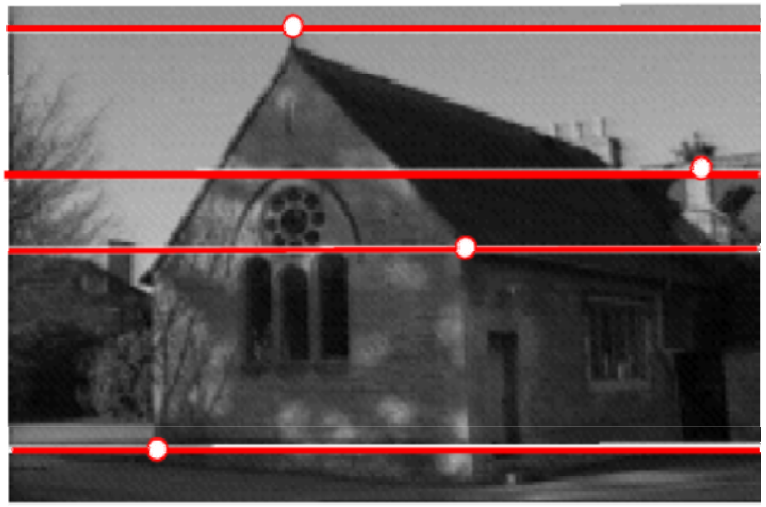
- Epipolar geometry is a consequence of the coplanarity of the camera centers and scene point
- The camera centers, corresponding points and scene point lie in a single plane, known as the epipolar plane



Epipolar Geometry

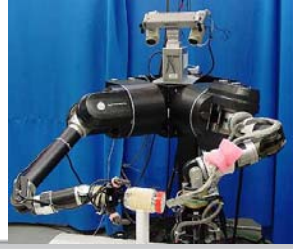




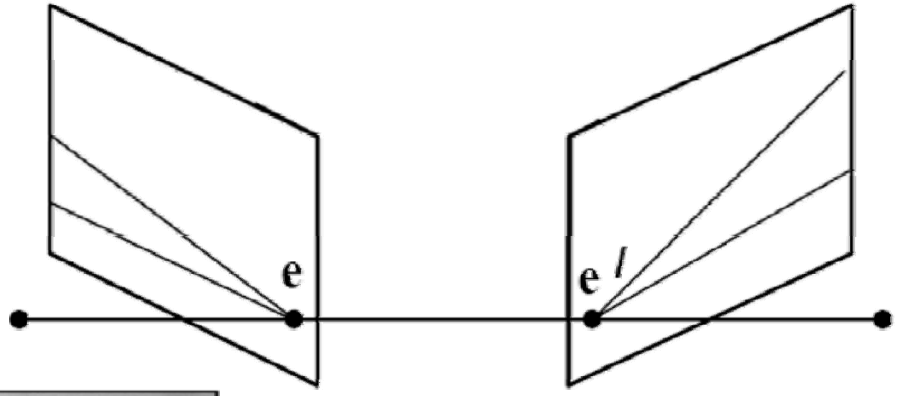


- Epipolar geometry depends only on the relative pose (position and orientation) and internal parameters of the two cameras, i.e. the position of the camera centers and image planes.
- It does not depend on the scene structure (3D points external to the camera).

Epipolar Geometry

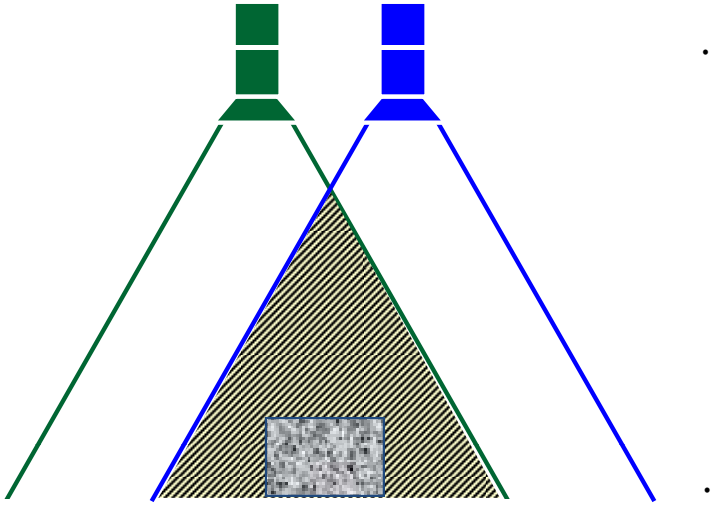


- Note, epipolar lines are in general not parallel

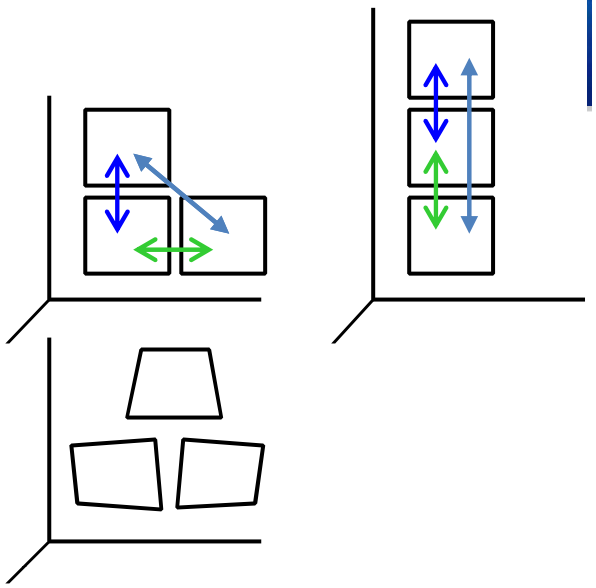


Camera Setup

Static Stereo:

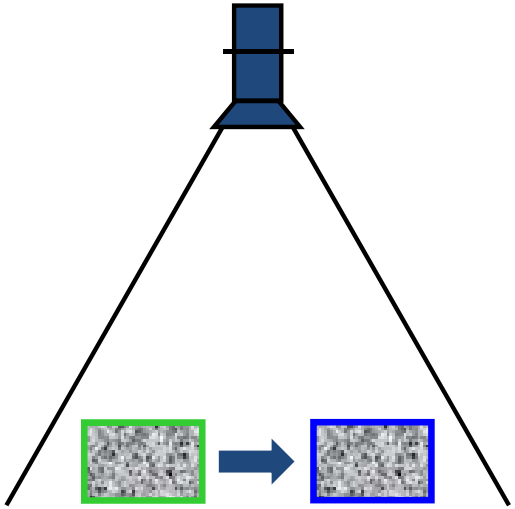


Binocular Stereo

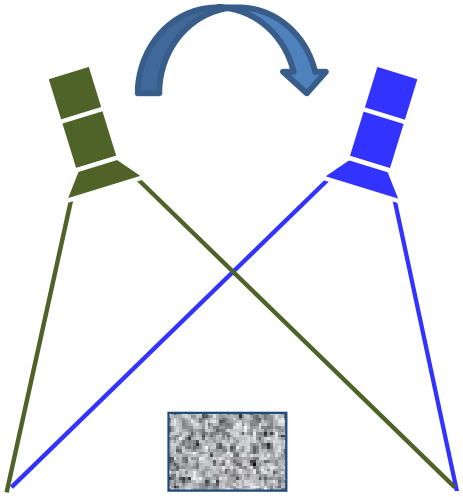


Trinocular Stereo

Dynamic Stereo:

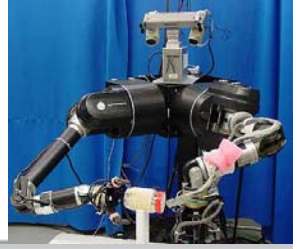


Moving Object

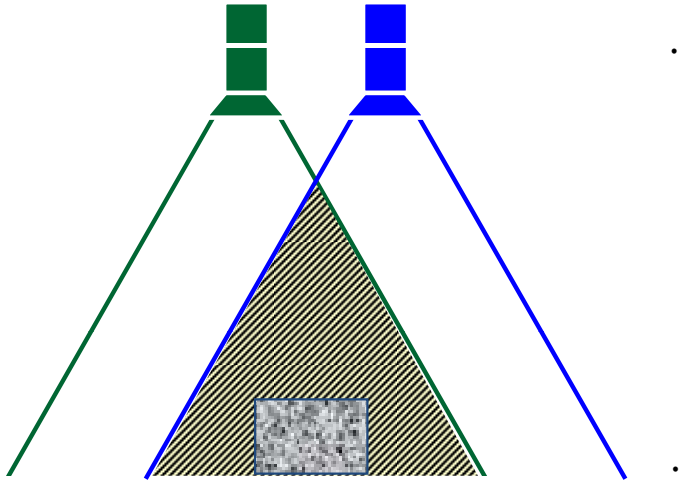


Moving Camera

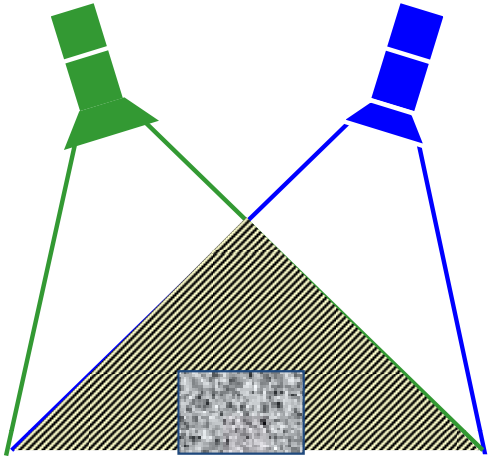
Choice of Camera Setup



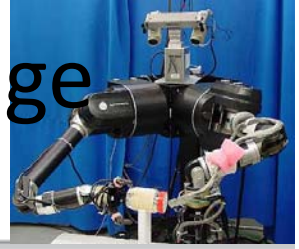
- Baseline
  - distance between cameras (focal points)



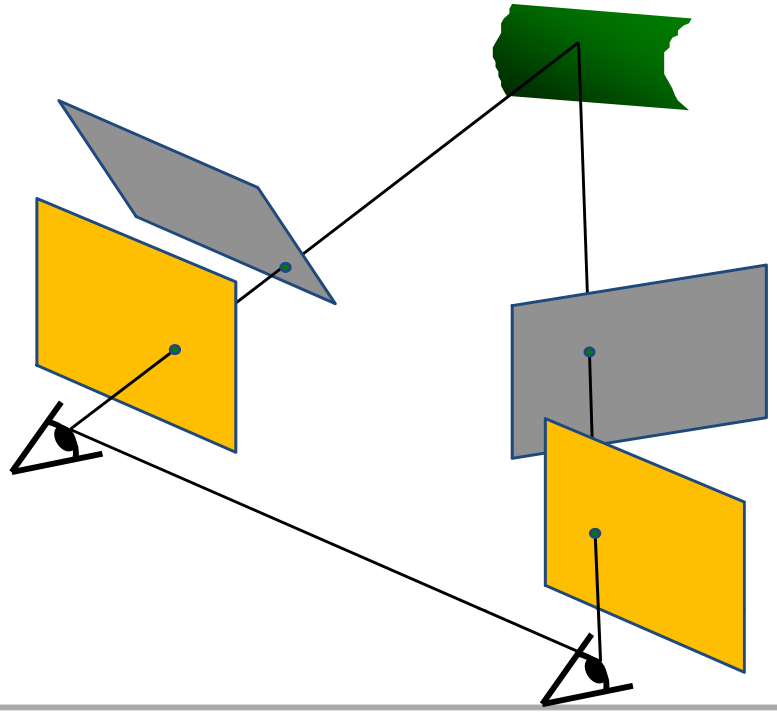
- Trade-off
  - Small baseline: Matching easier
  - Large baseline: Depth precision better



We can always achieve this geometry with image rectification



- Image Re-projection
  - Re-project image planes onto common plane parallel to line between optical centers
  - Notice, only focal point of camera really matters



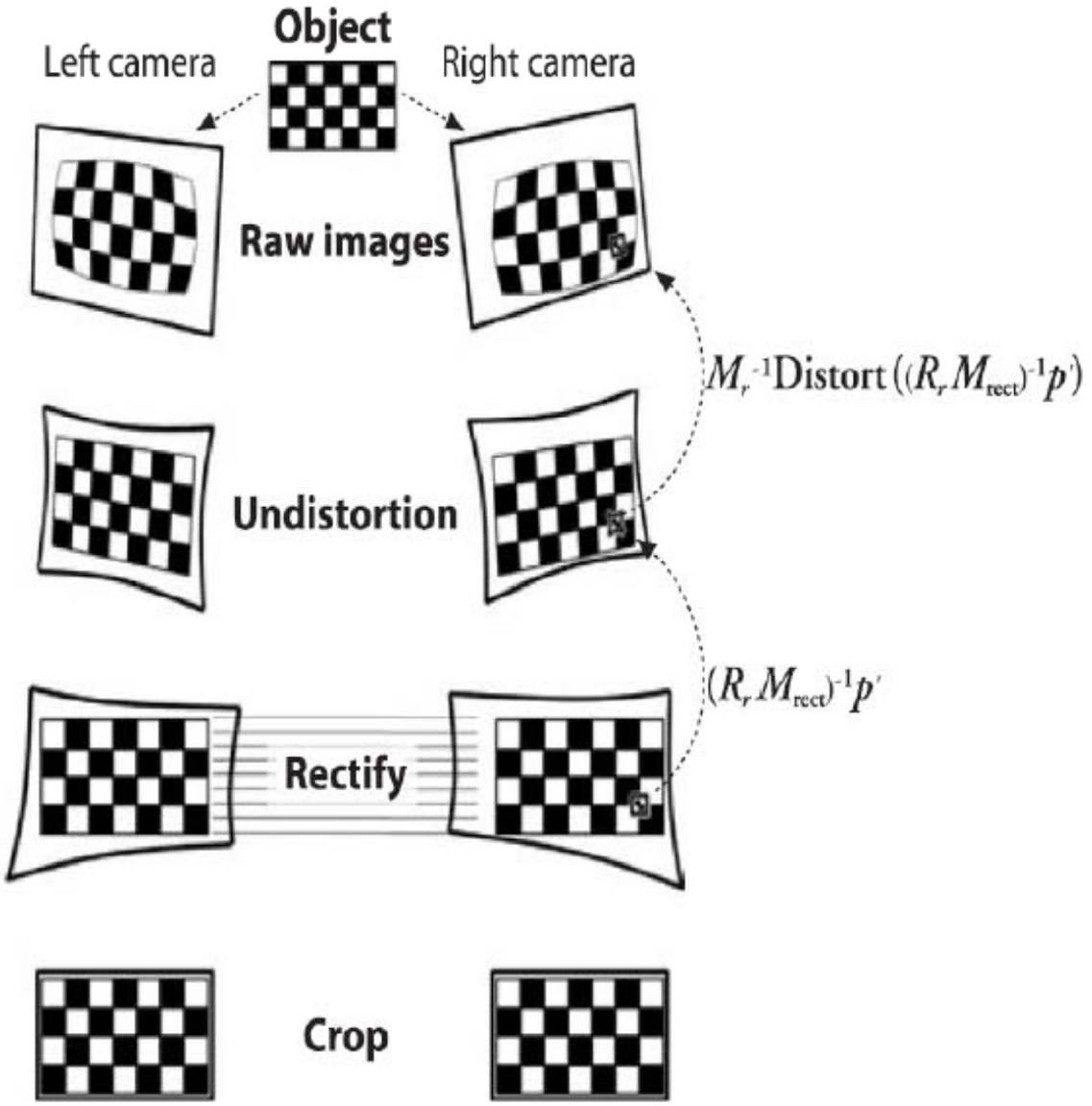
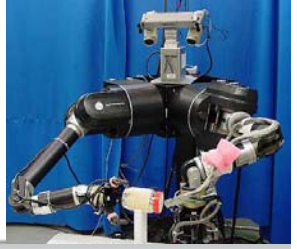
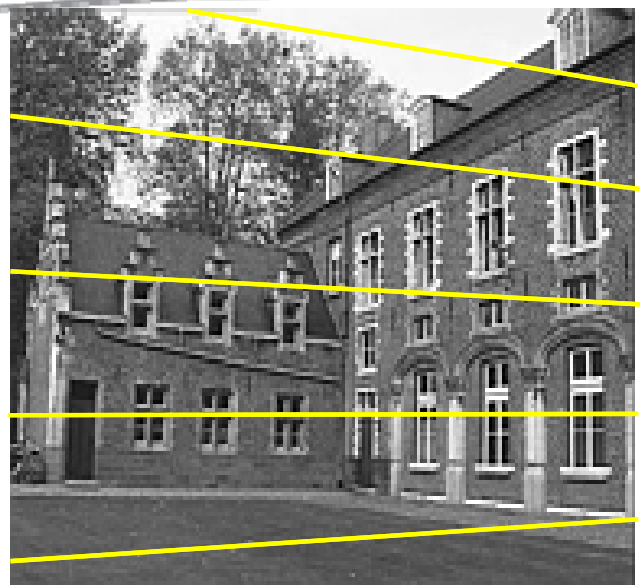
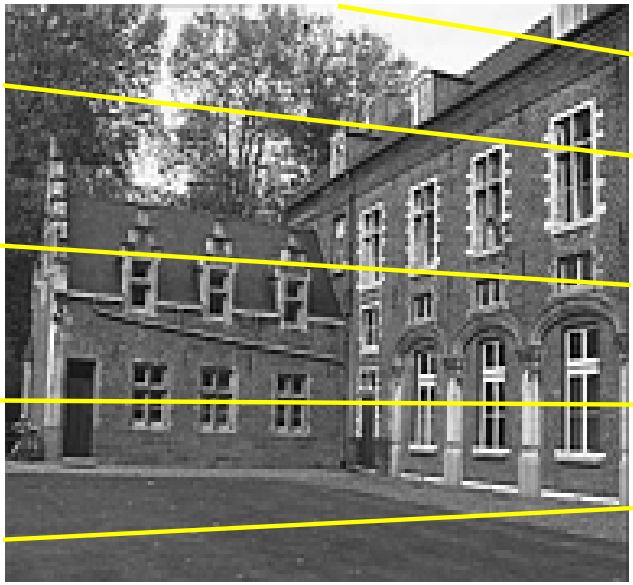
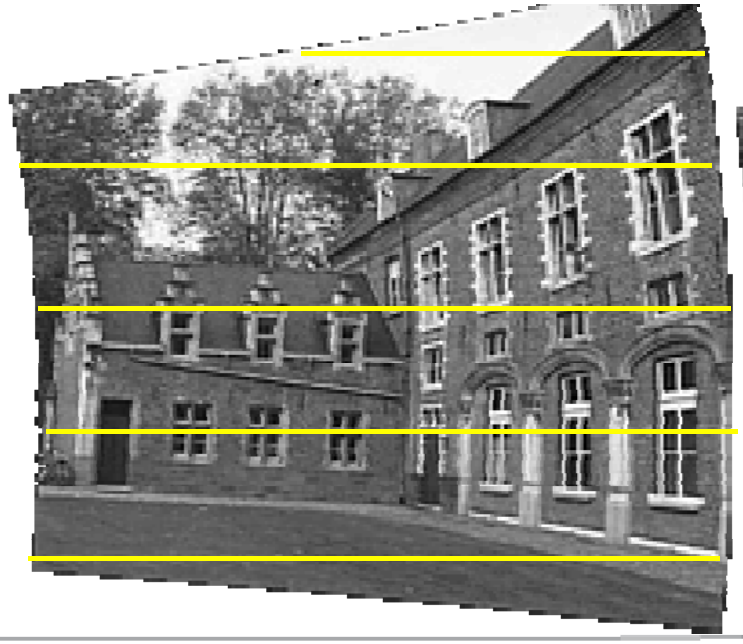


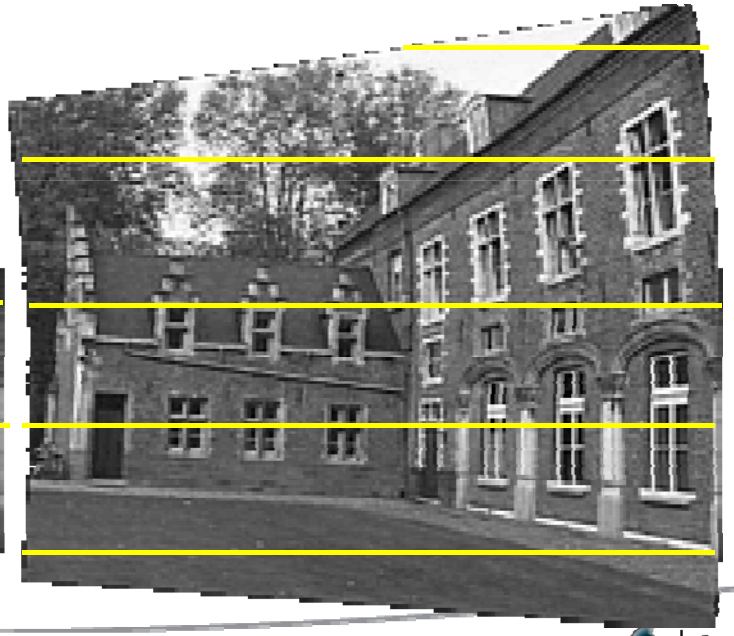
Image Rectification



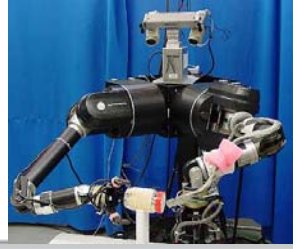




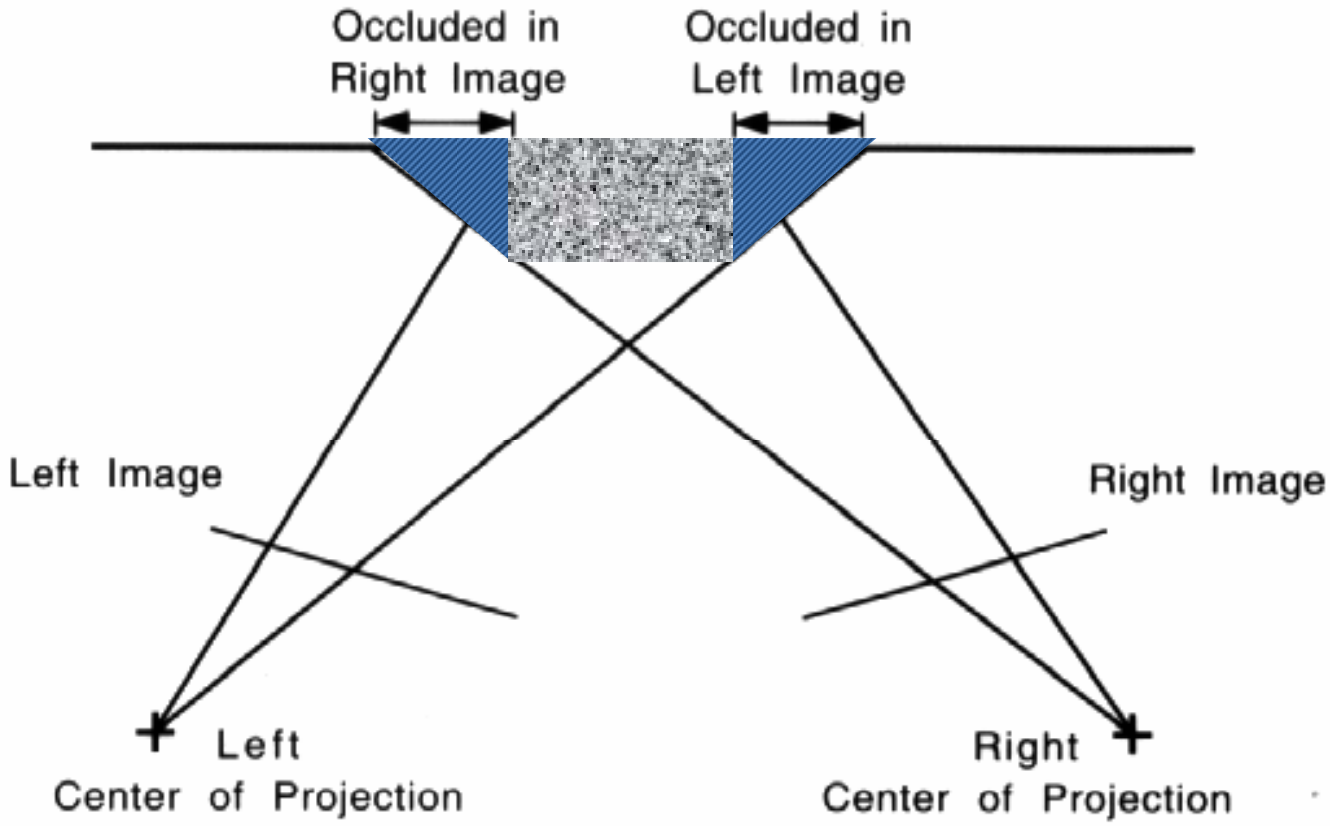




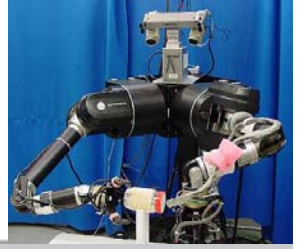
Occlusions



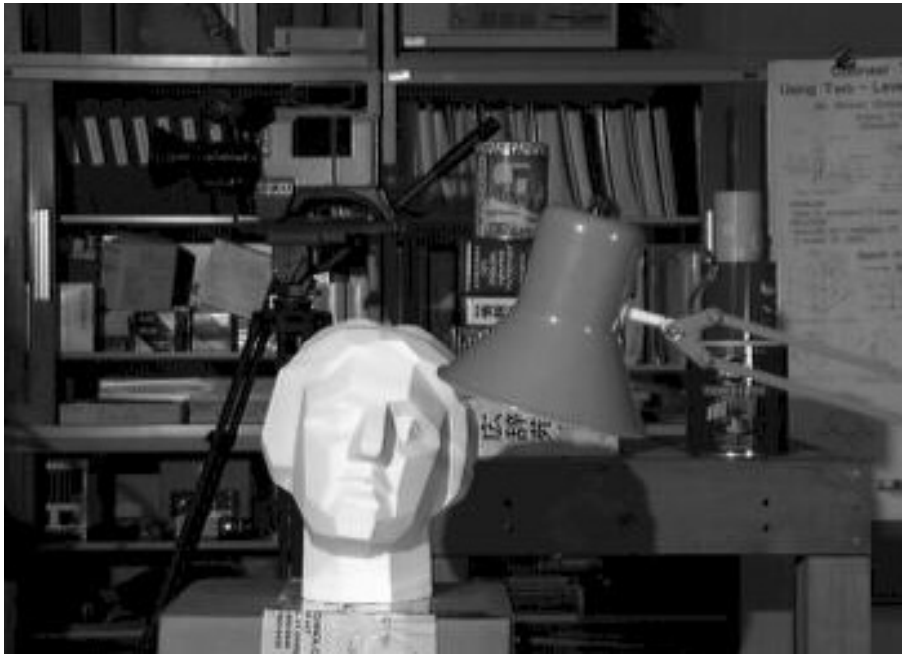
- View dependent
- Occluded points cannot be computed

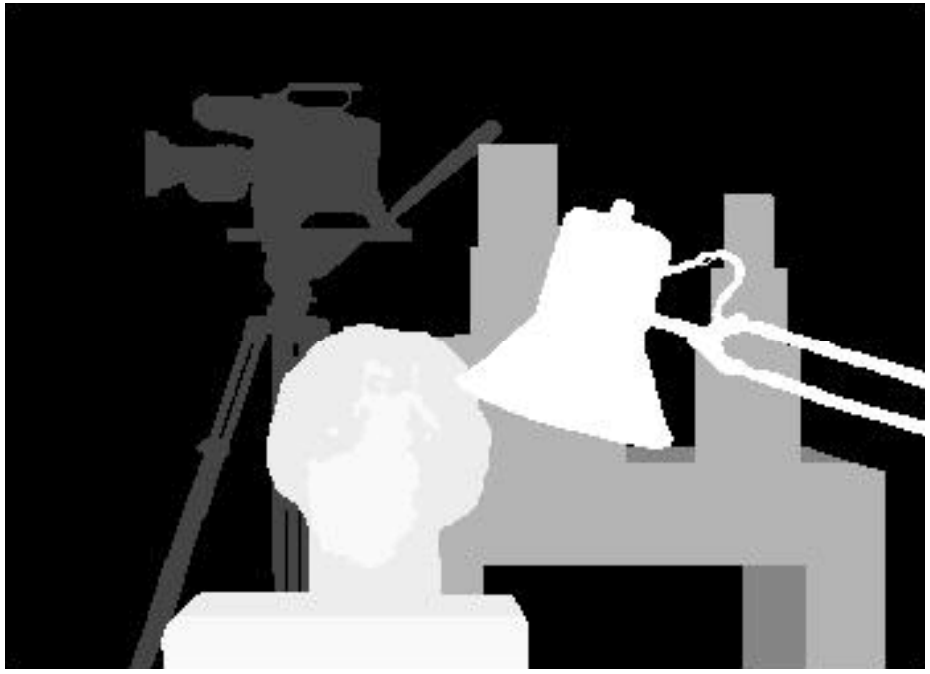


Goal: Disparity Map

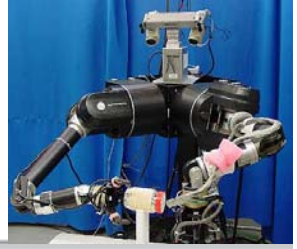


- Disparity:
  - The horizontal displacement between corresponding points
  - Closely related to scene depth



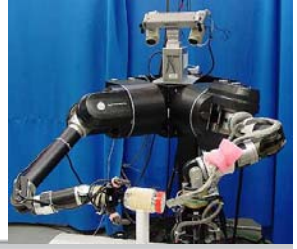


Prerequisites

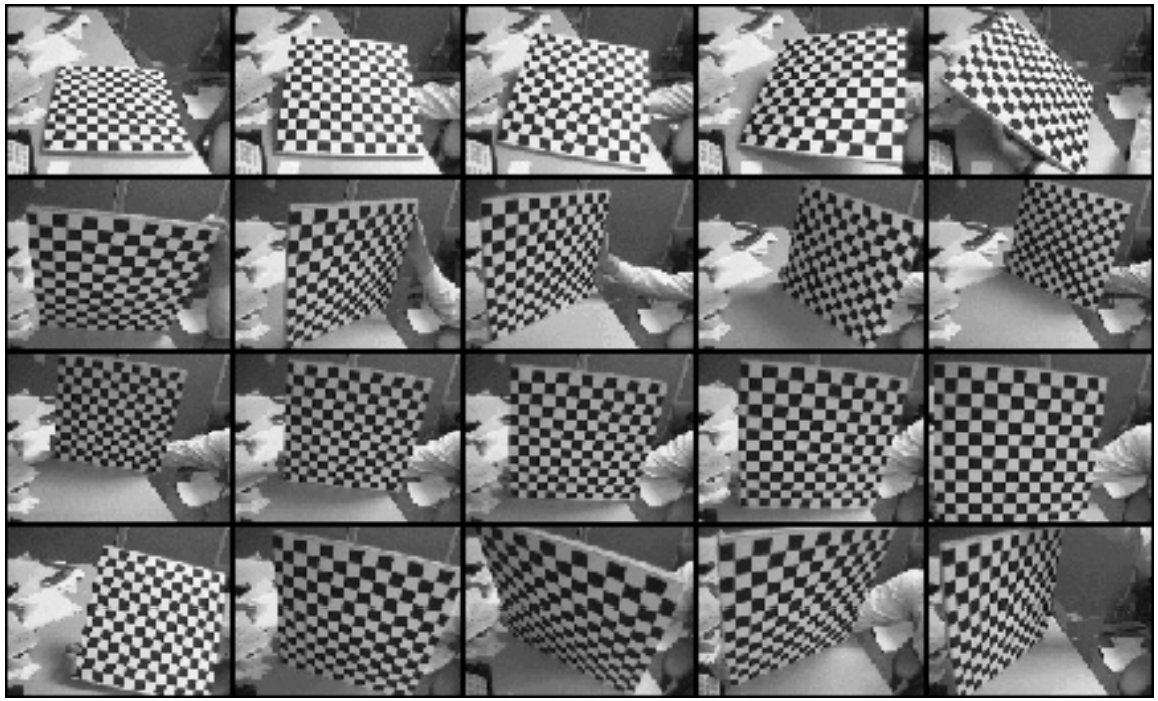


- Camera model parameters must be known:
  - **External** parameters:
    - Positions, orientations
  - **Internal** parameters:
    - Focal length, image center, distortion, etc..

Prerequisites



- Camera calibration



Correspondence Problem

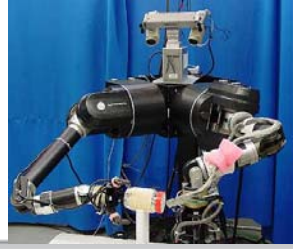
Correspondence Analysis

Left Stereo Image

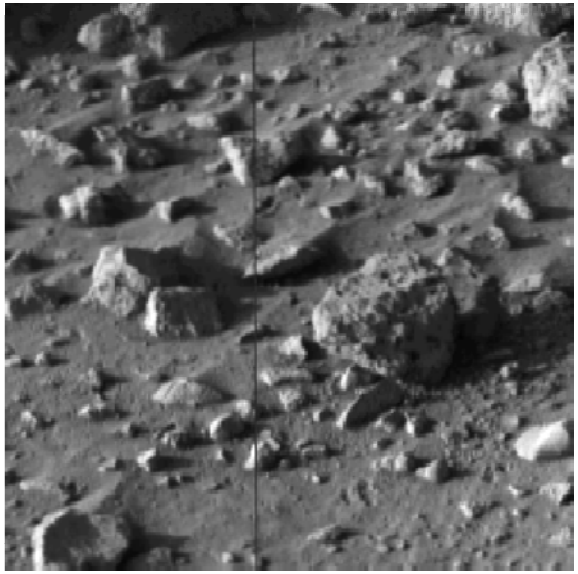
Right Stereo Image

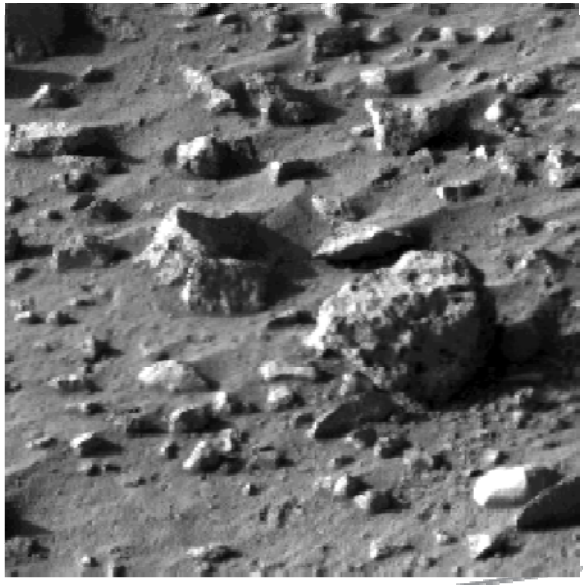
- Area Based
  - Comparison of **intensity values** in the left and right image
  - Correspondence due to the **similarity** between the **intensity values**
  - Correspondence **for each pixel**
- Feature Based
  - Comparison of **features** in left and right image
  - Correspondence on the basis of **selected characteristics** of features (edge orientation, edge length, gradient, etc.)
  - Correspondence only for **selected pixels**
  - **more accurate** because of sub-pixel positioning of features

Matching Challenges

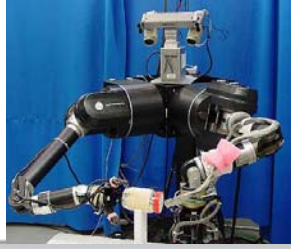


- Scene elements do not always look the same in the two images
  - Camera-related problems
    - Image noise, differing gain, contrast, etc..
  - Viewpoint-related problems:
    - Perspective distortions
    - Occlusions
    - Specular reflections

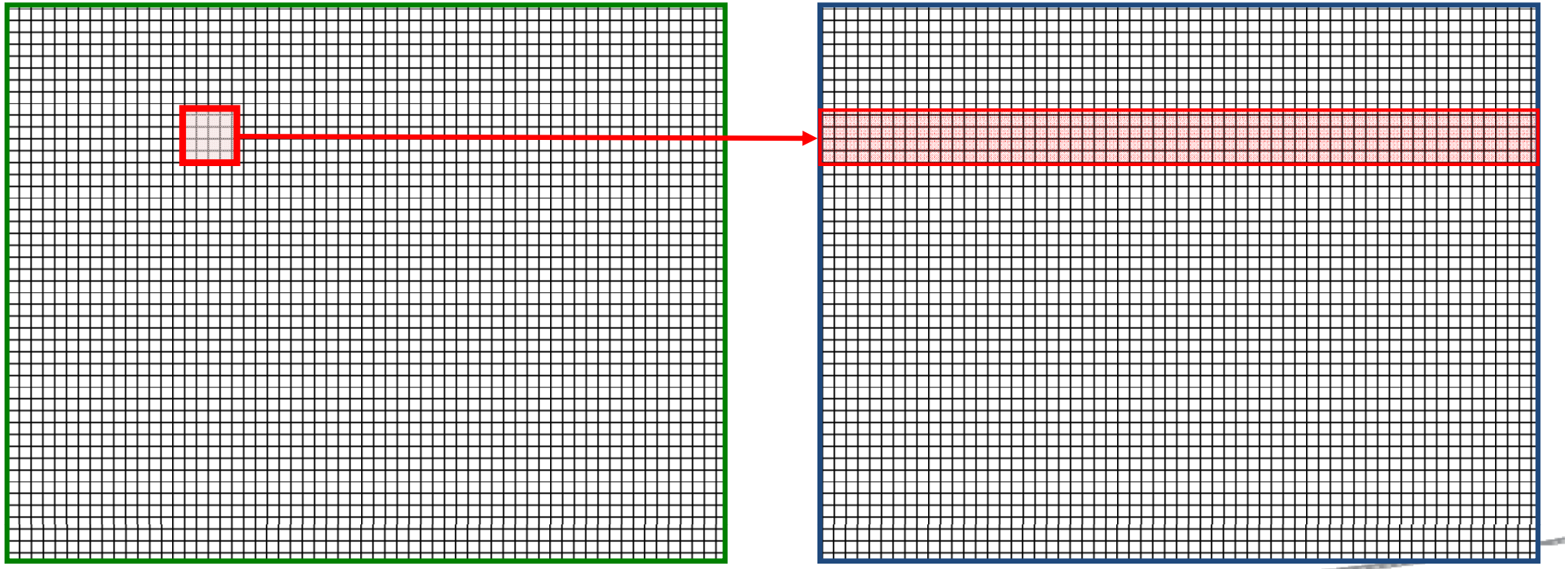




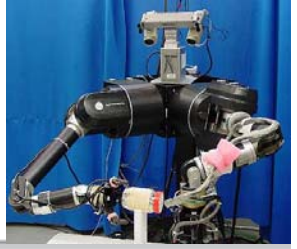
Area-based Matching



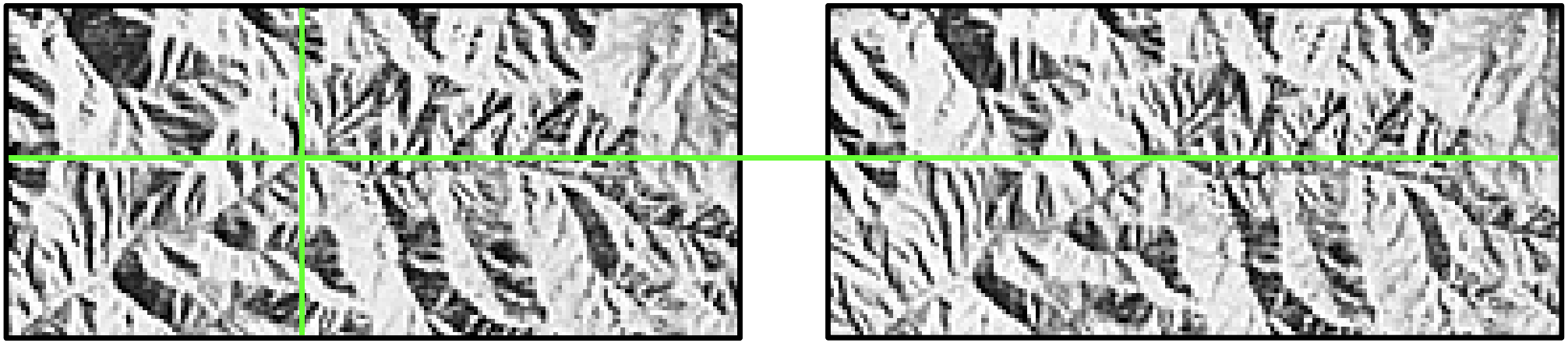
- Observation window in the left image fix
- for each position in the right image the correlation function is calculated
- Window will be “slided" from left to right across the image
- is performed for each position in the left image



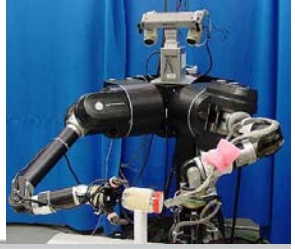
Area-based Matching



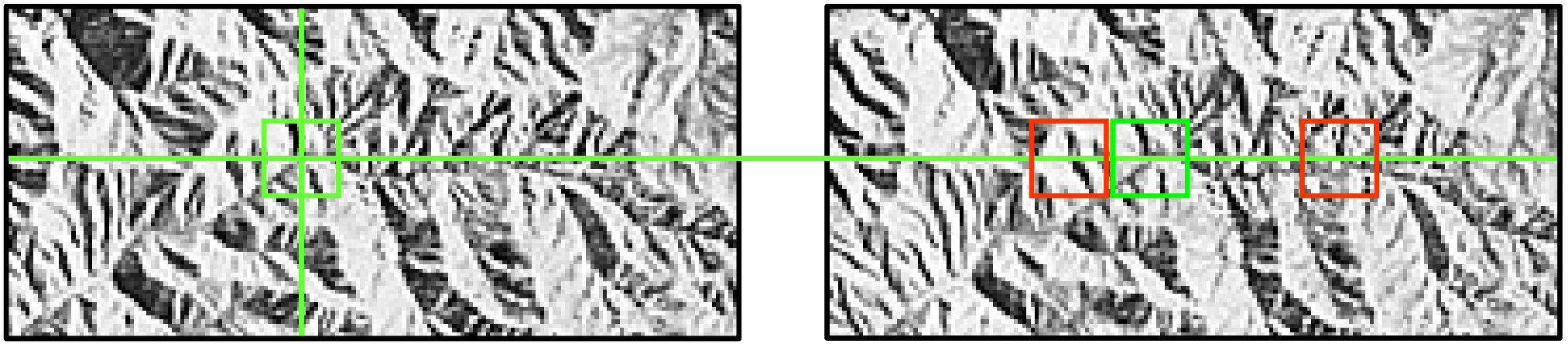
- Finding pixel-to-pixel correspondences
  - For each pixel in the left image, search for the most similar pixel in the right image



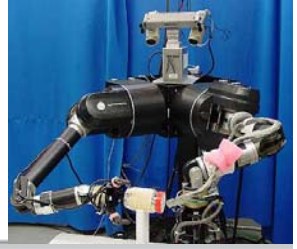
Area-based Matching



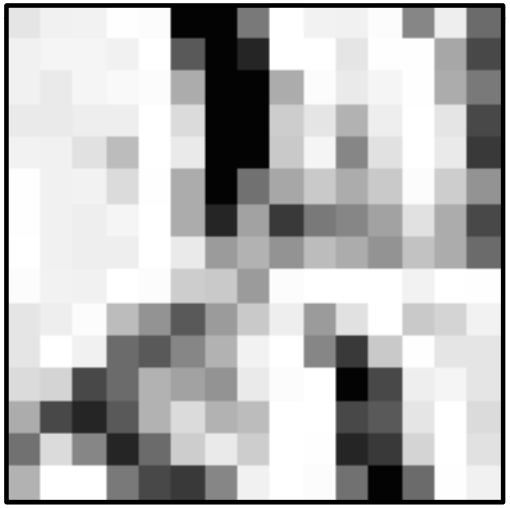
- Finding pixel-to-pixel correspondences
  - For each pixel in the left image, search for the most similar pixel in the right image
  - Using neighborhood windows

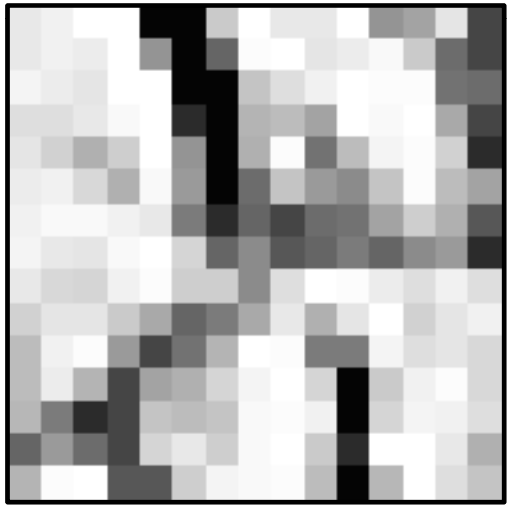


Area-based Matching



- Similarity measures for two windows
  - SAD (sum of absolute differences)
  - SSD (sum of squared differences)
  - CC (cross-correlation)
  - …

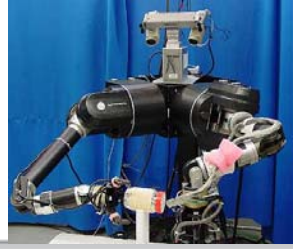




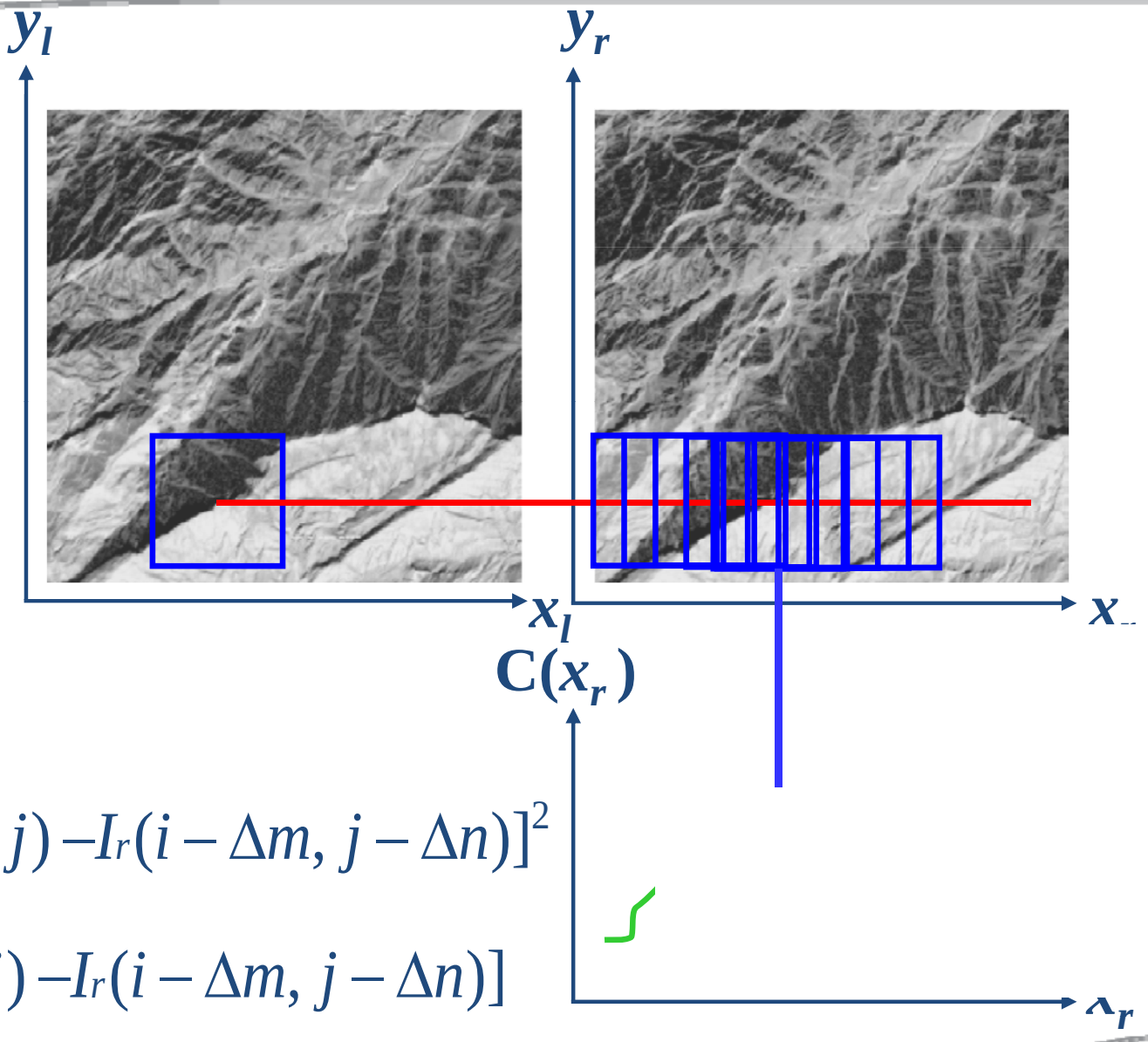
$$SSD(\Delta m, \Delta n) \hat{=} \sum_{i,j \in R} [I_l(i,j) - I_r(i-\Delta m, j-\Delta n)]^2$$

$$CC(\Delta m, \Delta n) \hat{=} \sum_{i,j \in R} [I_l(i,j) - I_r(i-\Delta m, j-\Delta n)]$$

Area-based Matching



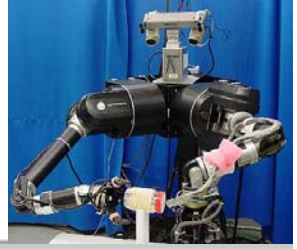
- Correspondence due to **similarity** between the gray values in the left and right image

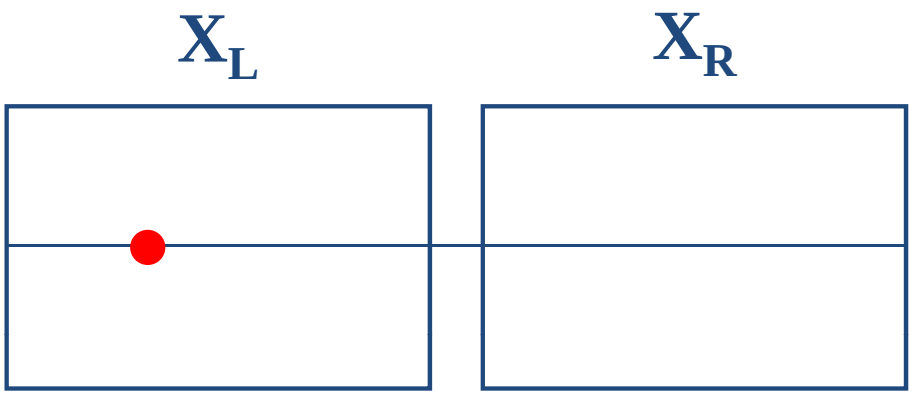


$$SSD(\Delta m, \Delta n) \hat{=} \sum_{i,j \in R} [I_l(i,j) - I_r(i-\Delta m, j-\Delta n)]^2$$

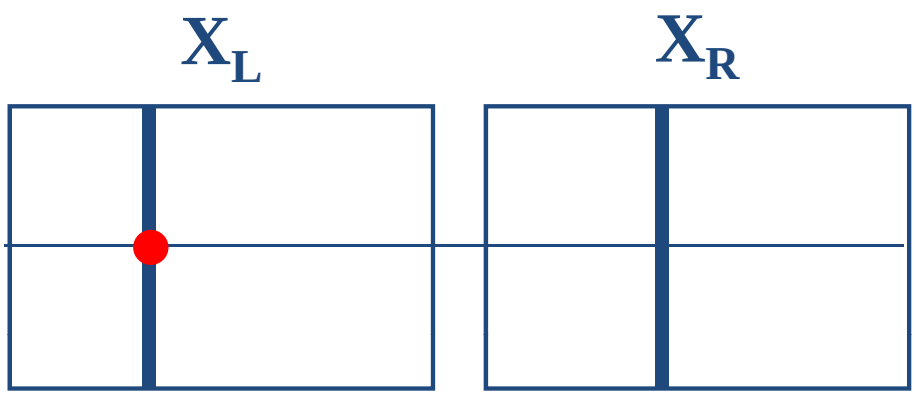
$$CC(\Delta m, \Delta n) \hat{=} \sum_{i,j \in R} [I_l(i,j) - I_r(i-\Delta m, j-\Delta n)]$$

Correspondence Problem

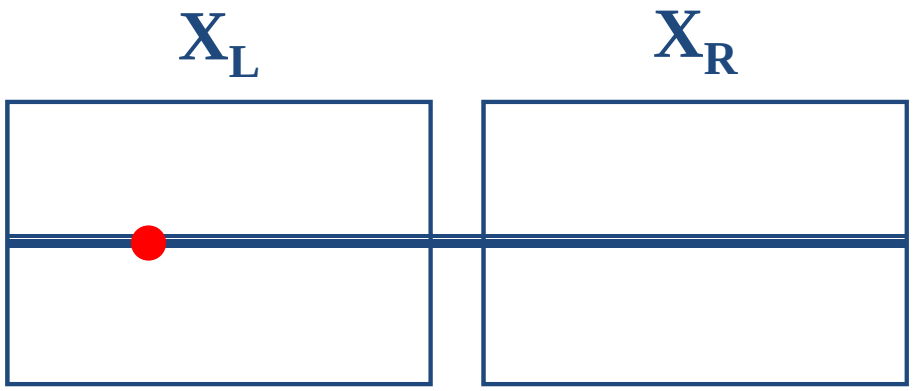




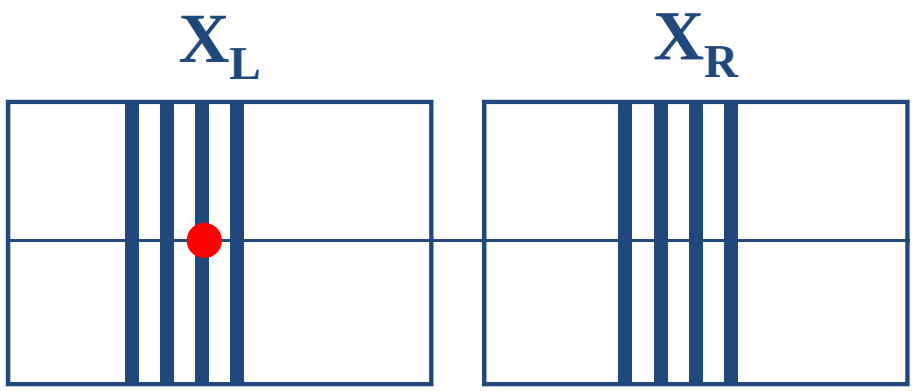
Point does not exist



Exact correspondence

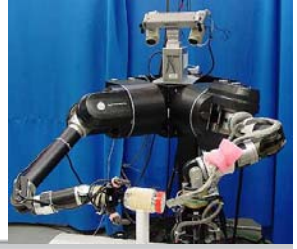


Horizontal lines



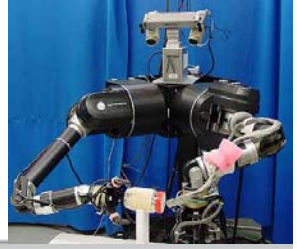
Ambiguous correspondence

Area-based Matching

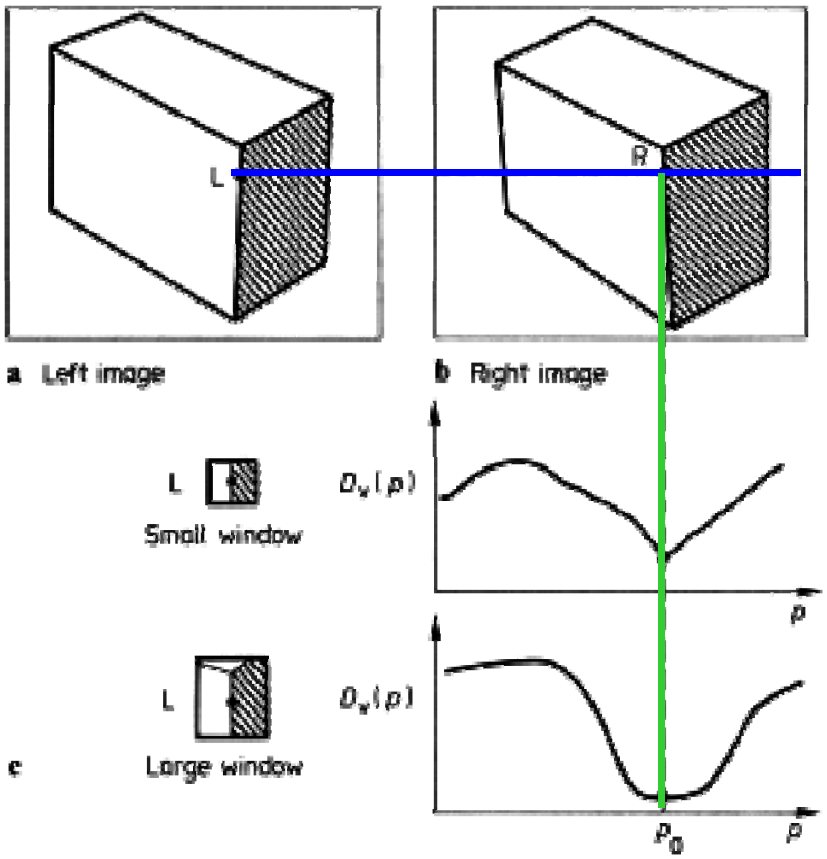


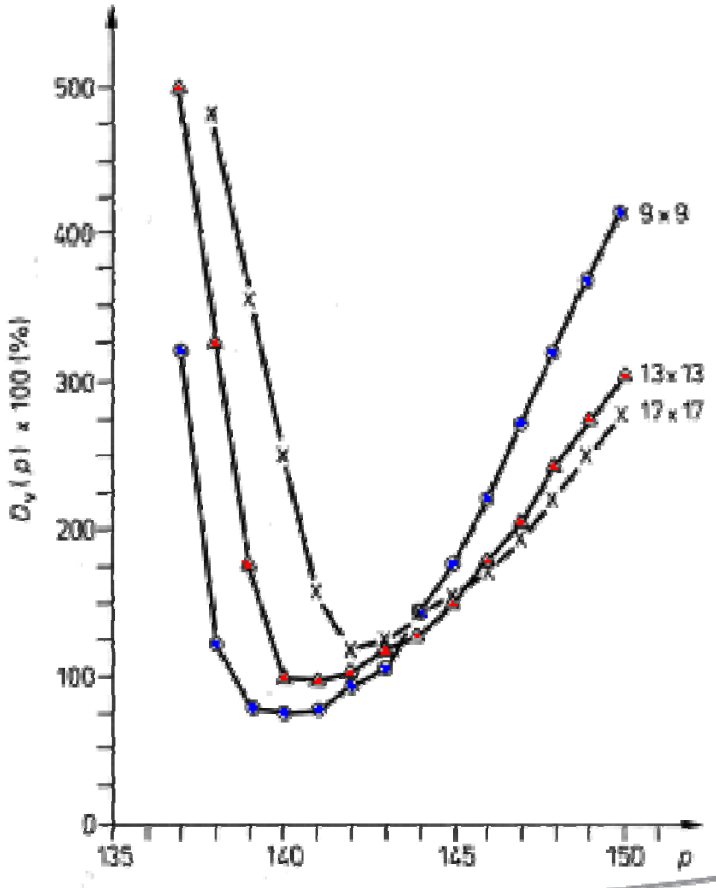
- Choice of window size
  - Factors to considers:
    - Ambiguity
    - Noise sensitivity
    - Sensitivity towards viewpoint-related distortions
    - Expected object sizes
    - Frequency of depth jumps

Correspondence Search

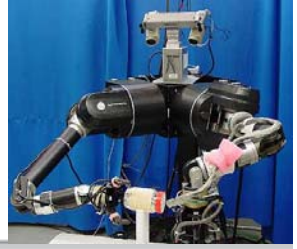


- Correspondence is strongly dependent on window size used
  - Different algorithms like adaptive matching, pyramids ….

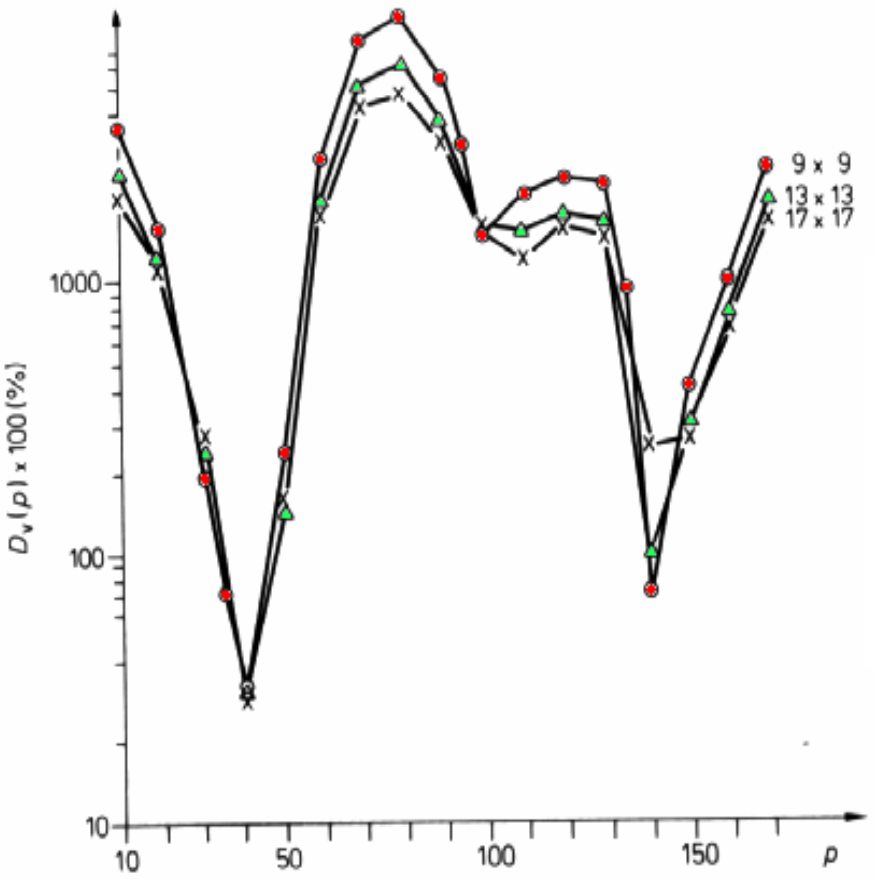


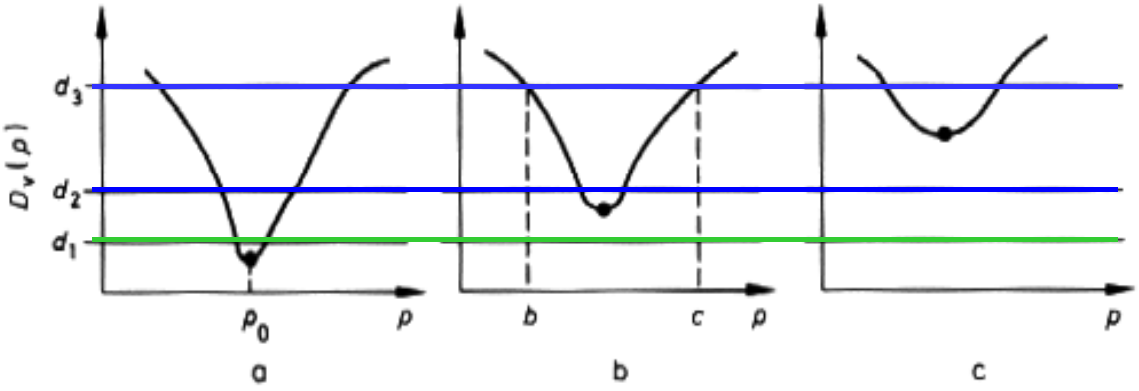


Correspondence Search



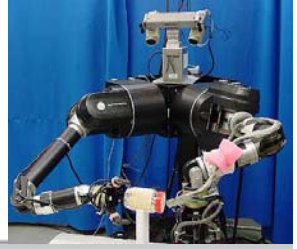
- Solution of ambiguity with threshold or additional constraints
- Different threshold values

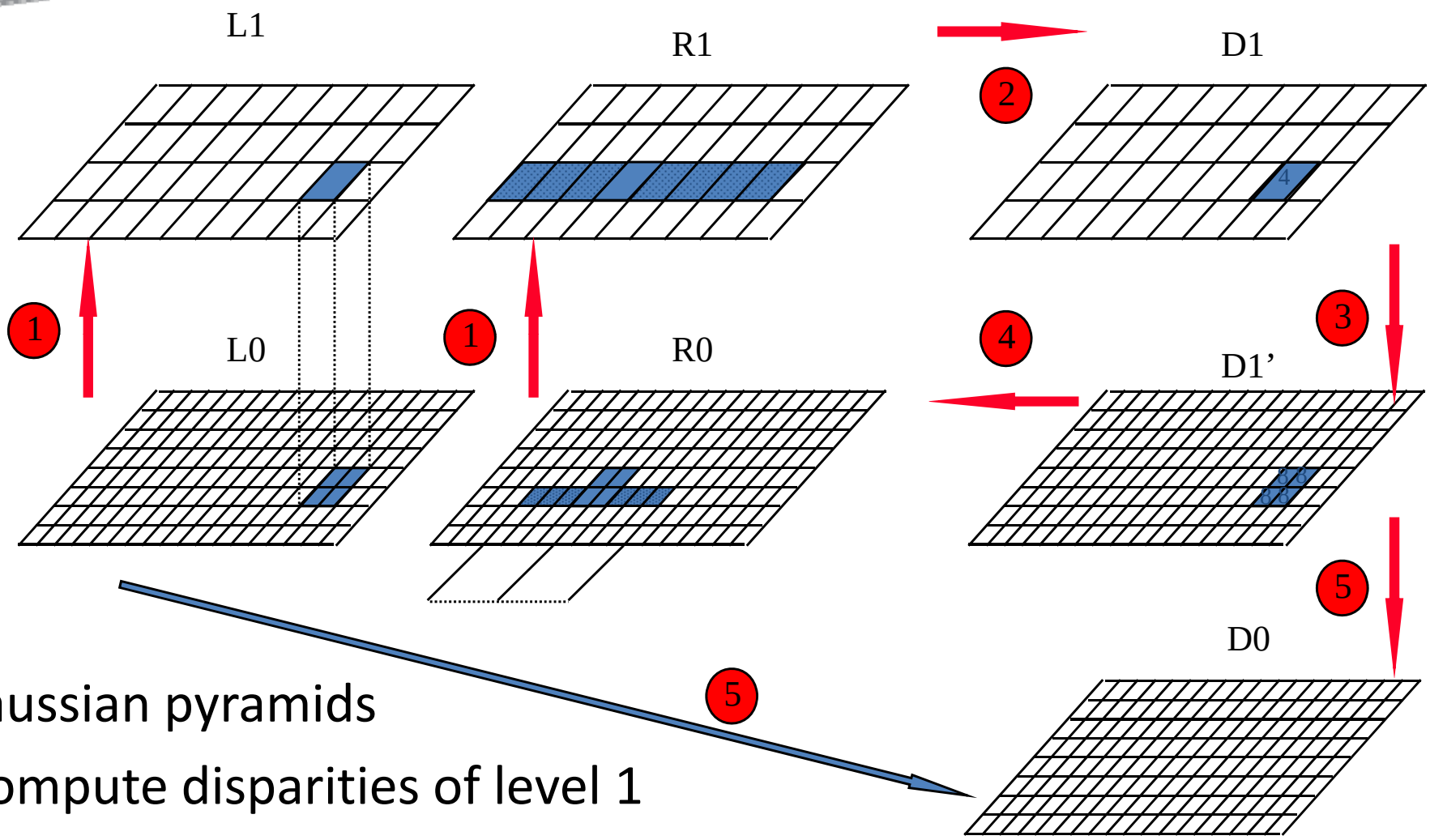




found **continue searching** **not found**

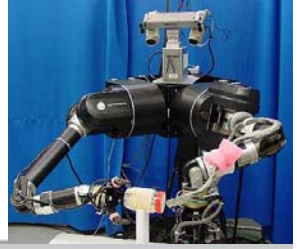
Hierarchical Matching



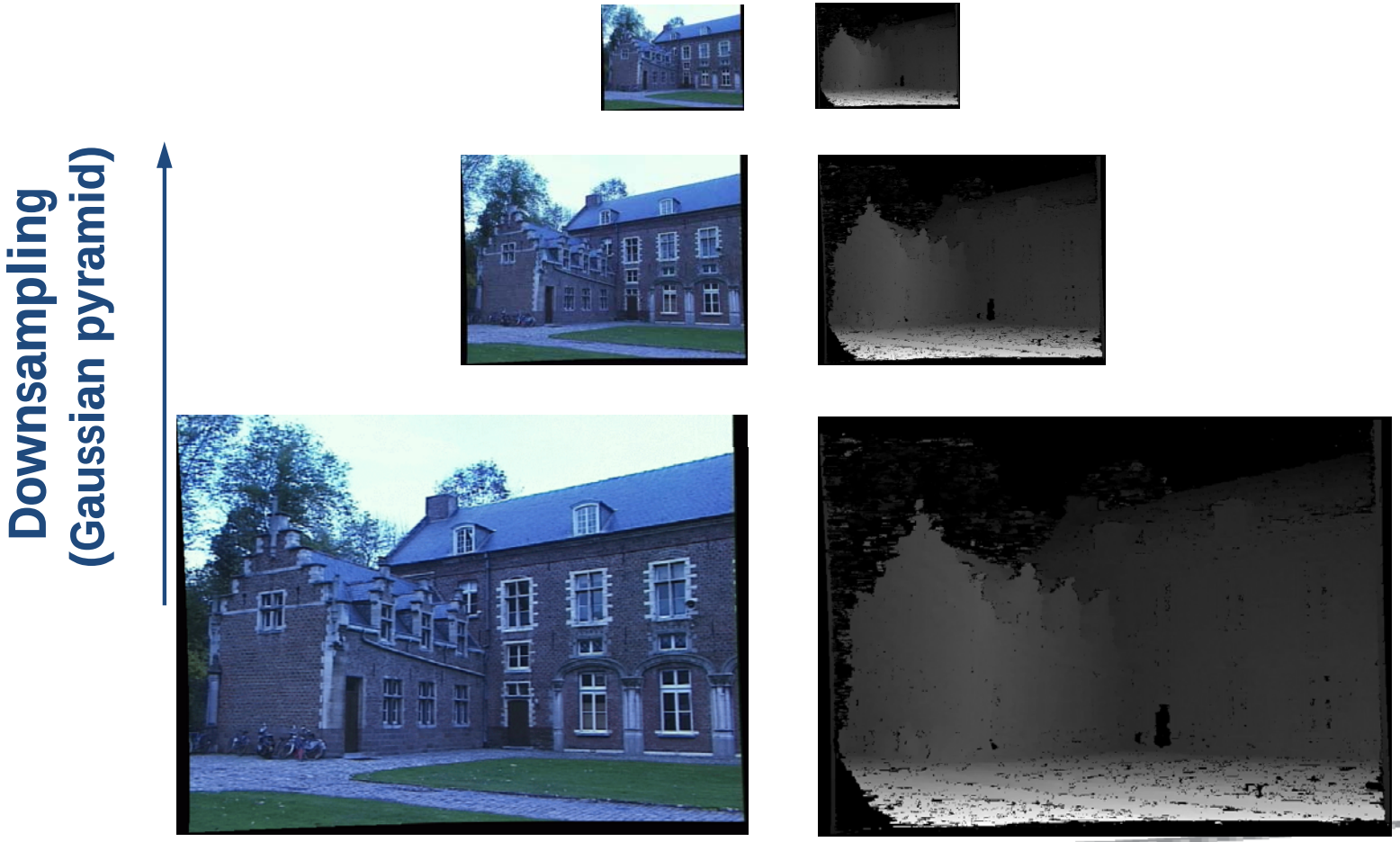


- Gaussian pyramids
- Compute disparities of level 1
- Project values
- Compute disparities of lower level

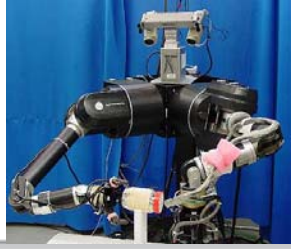
Hierarchical Stereo Matching



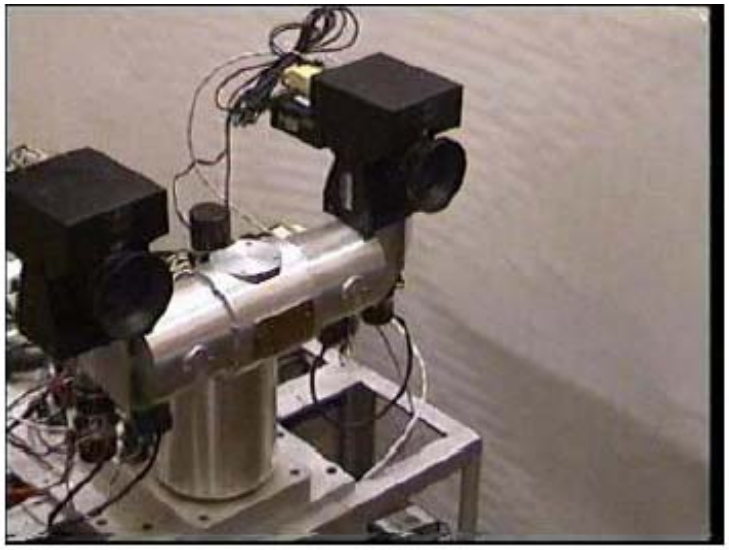
- Allows faster computation
- Deals with large disparity ranges



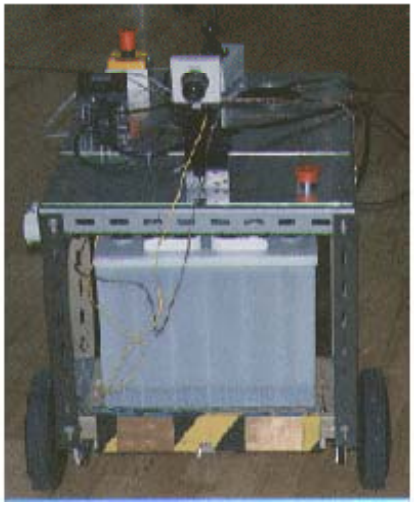
Example



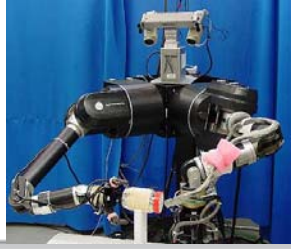
Stereo head



Camera on a mobile vehicle



Example



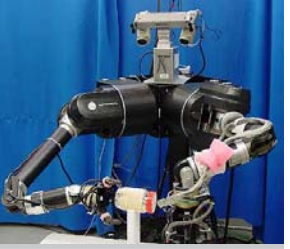
left image

right image

right image

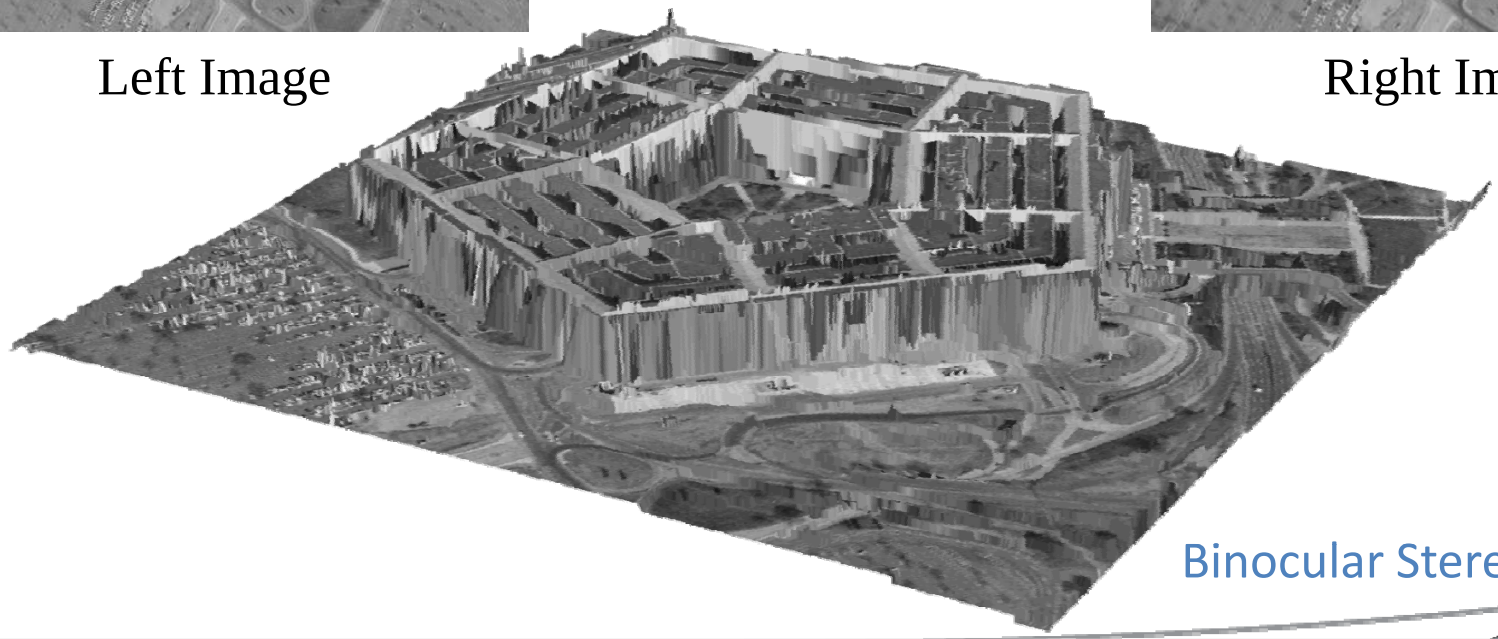
depth map

Pentagon Example



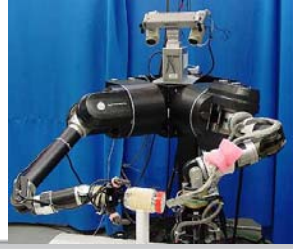
Left Image

Right Image



Binocular Stereo

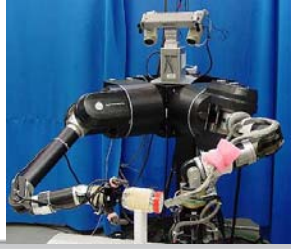
Problems with Window-based Matching

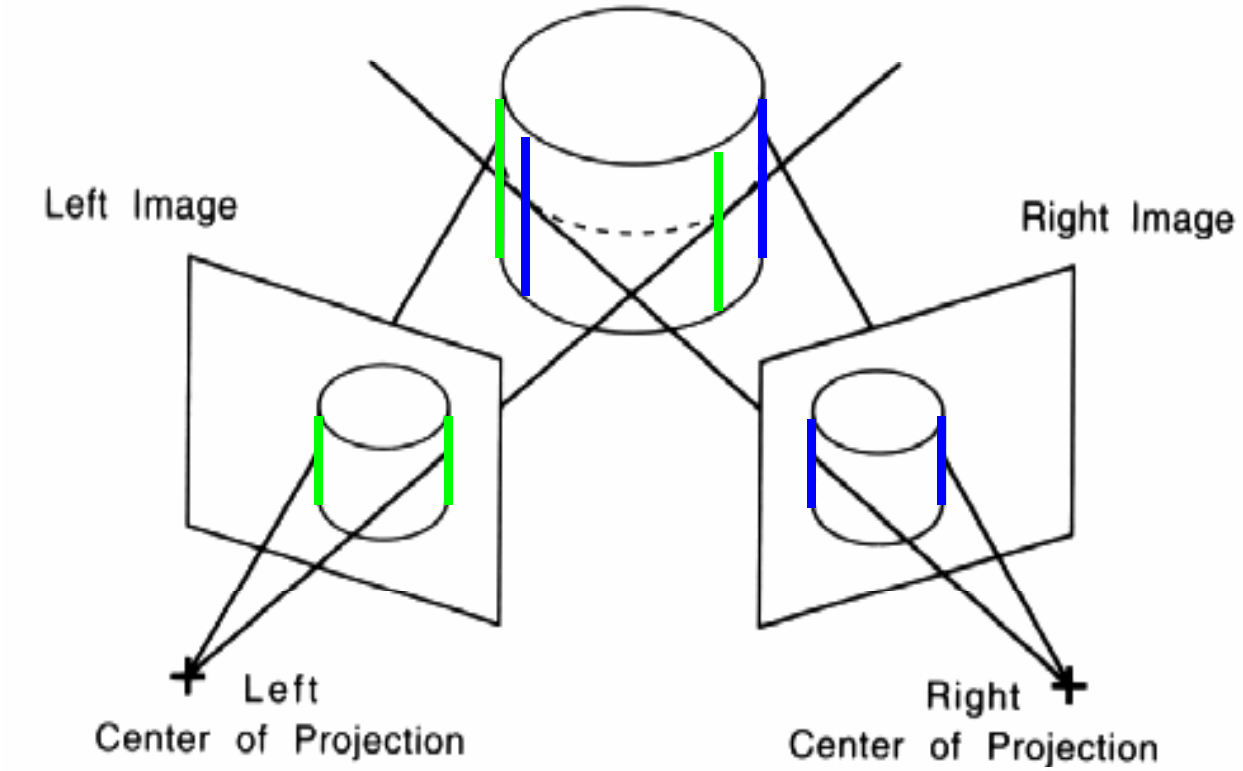


- Disparity within the window must be constant.
- Bias the results towards frontal-parallel surfaces.
- Blur across depth discontinuities.
- Perform poorly in textureless regions.
- Erroneous results in occluded regions

Feature-based Matching

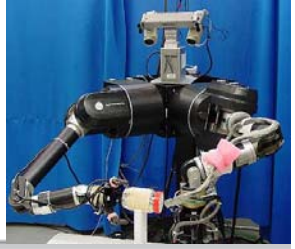
Feature-based Matching

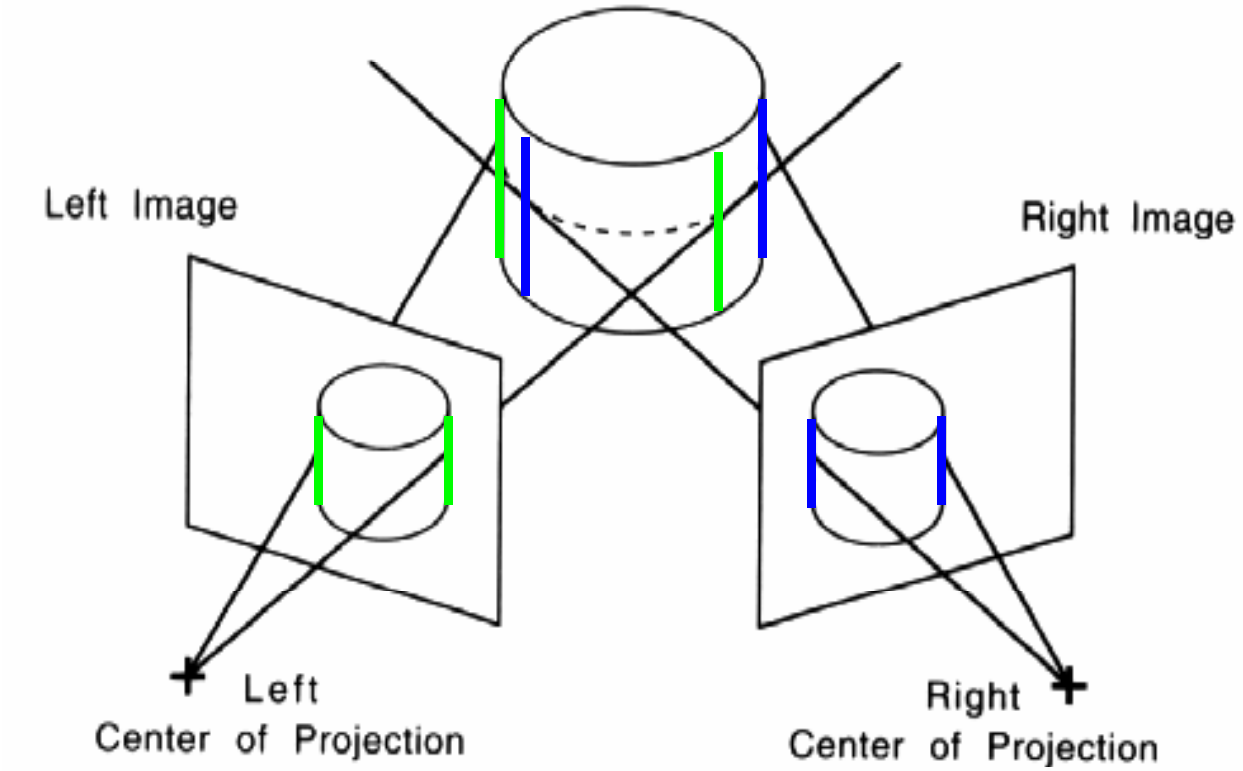




- Viewpoint-independent edges as a prerequisite for correct interpretation: violation = wrong values.
  - Edges are basis of evaluation
  - Advantage: very accurate
  - Disadvantage: very few range points

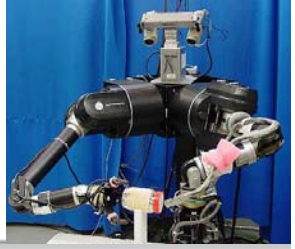
Feature-based Matching



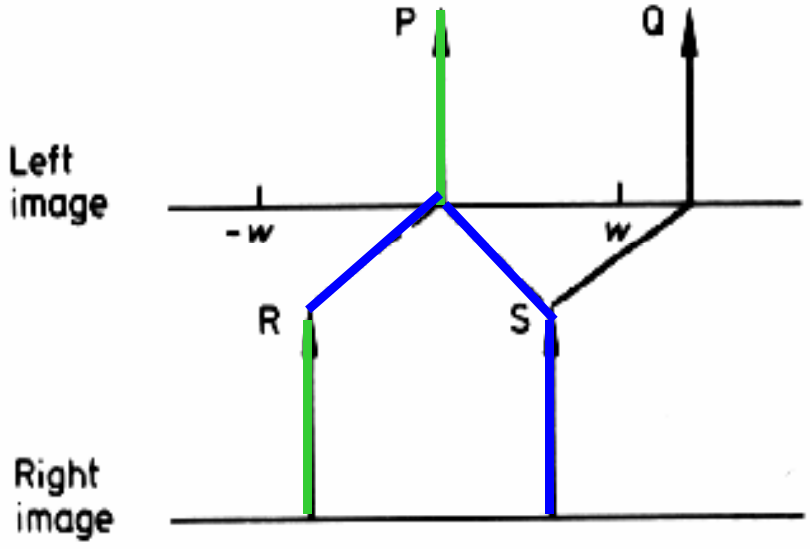


- Viewpoint-independent edges as a prerequisite for correct interpretation: violation = wrong values.
  - Edges are basis of evaluation
  - Advantage: very accurate
  - Disadvantage: very few range points

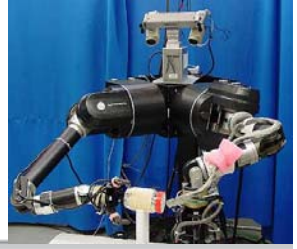
Feature-based Matching



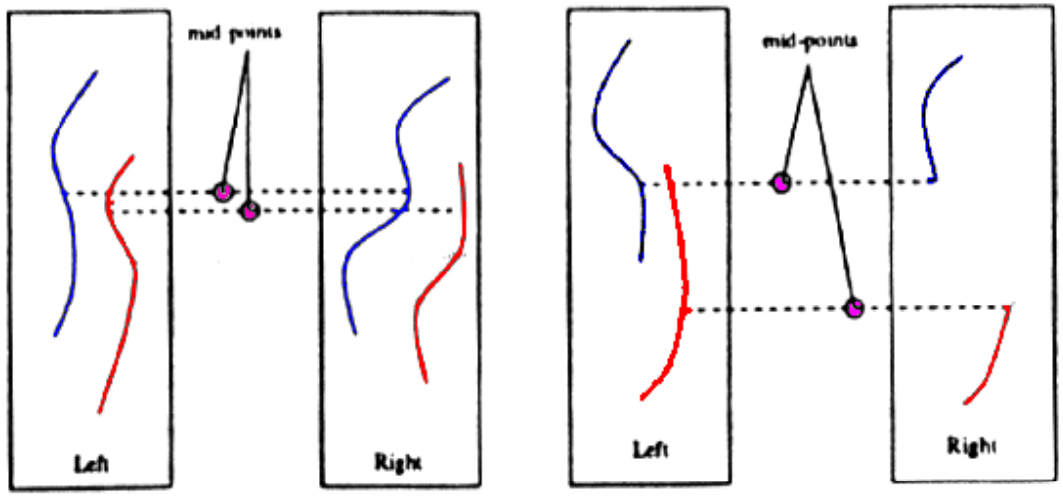
- Order of edges must be preserved
  - Pulling effect by shape and curve of edges
  - unique assignment possible



Contours as Features

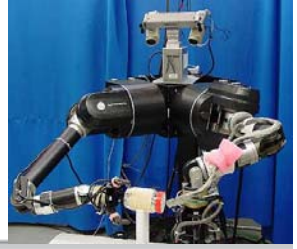


- Restrictions on the set of edges => Candidates for stereo
  - only vertical contours (> 30 to x-axis)
  - no intersecting contours
  - Epipolar line may be crossed only once
  - Contours must have minimum length



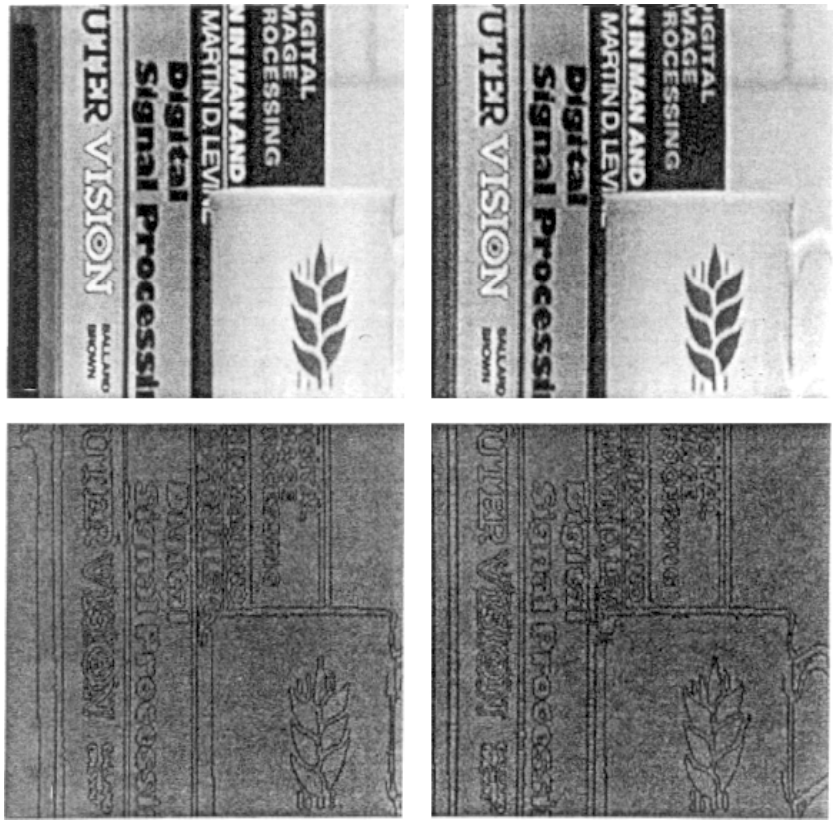
  - Contours must have similar shape
  - Contours must have similar angle to epipolar line
  - Contours have similar gradient magnitude
  - Order of the contours must be retained

Results



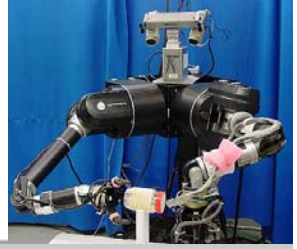
- Zero Crossings of the original images with a predefined threshold value are the basis for evaluation:

Left image **Right image** **Ground Truth**

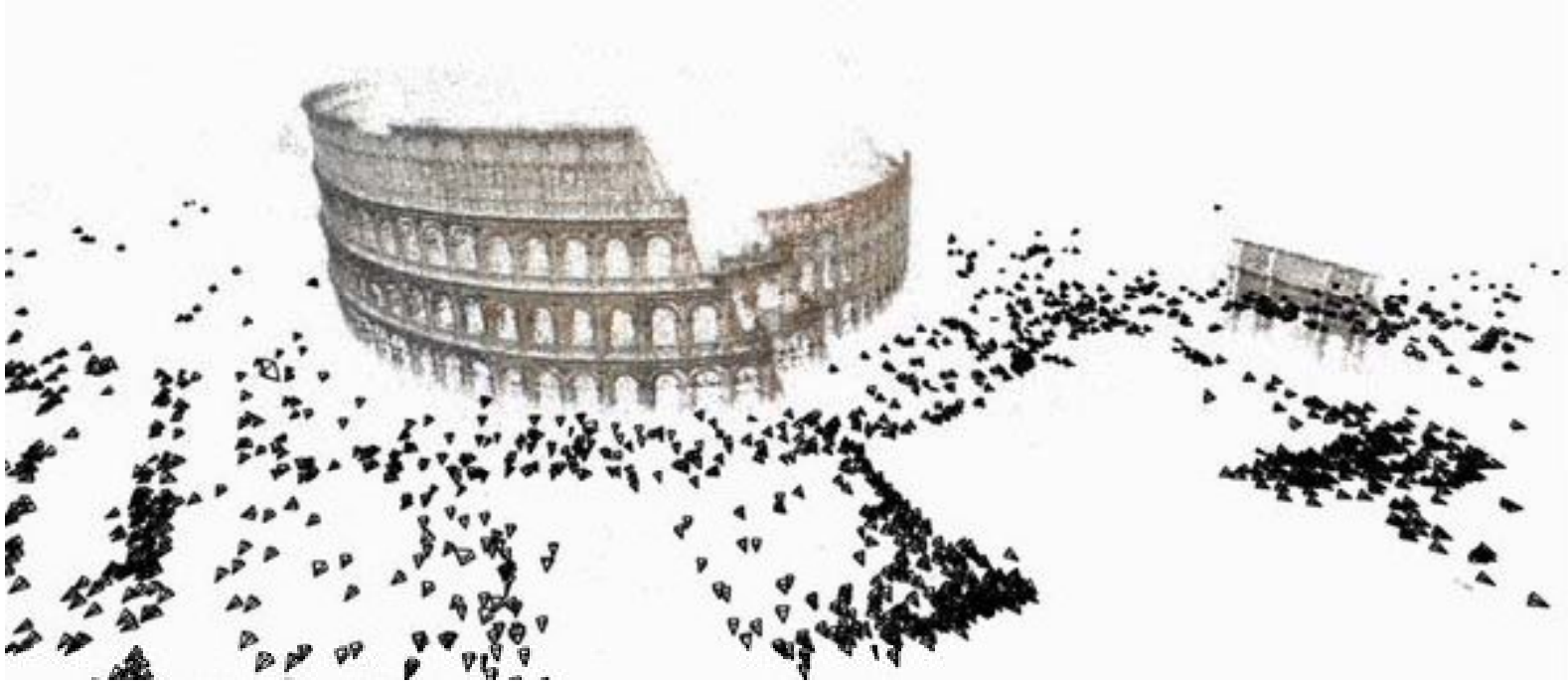


Zero crossings **Results**

Building Rome in a Day

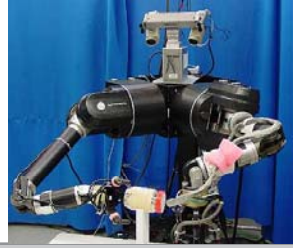


- Entering the search term Rome on Flickr returns more than two million photographs.
- reconstructing entire cities from images harvested from the web

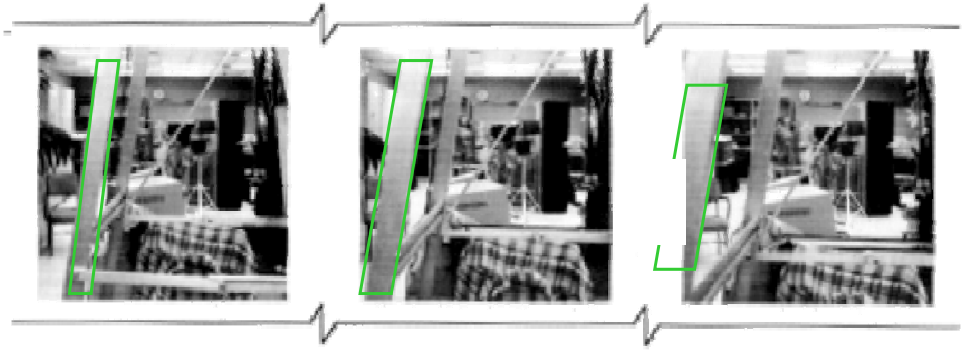


Structure from Motion (SfM)

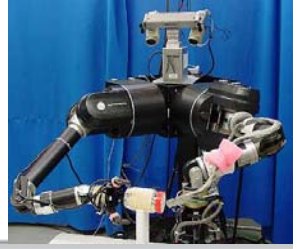
Structure from Motion



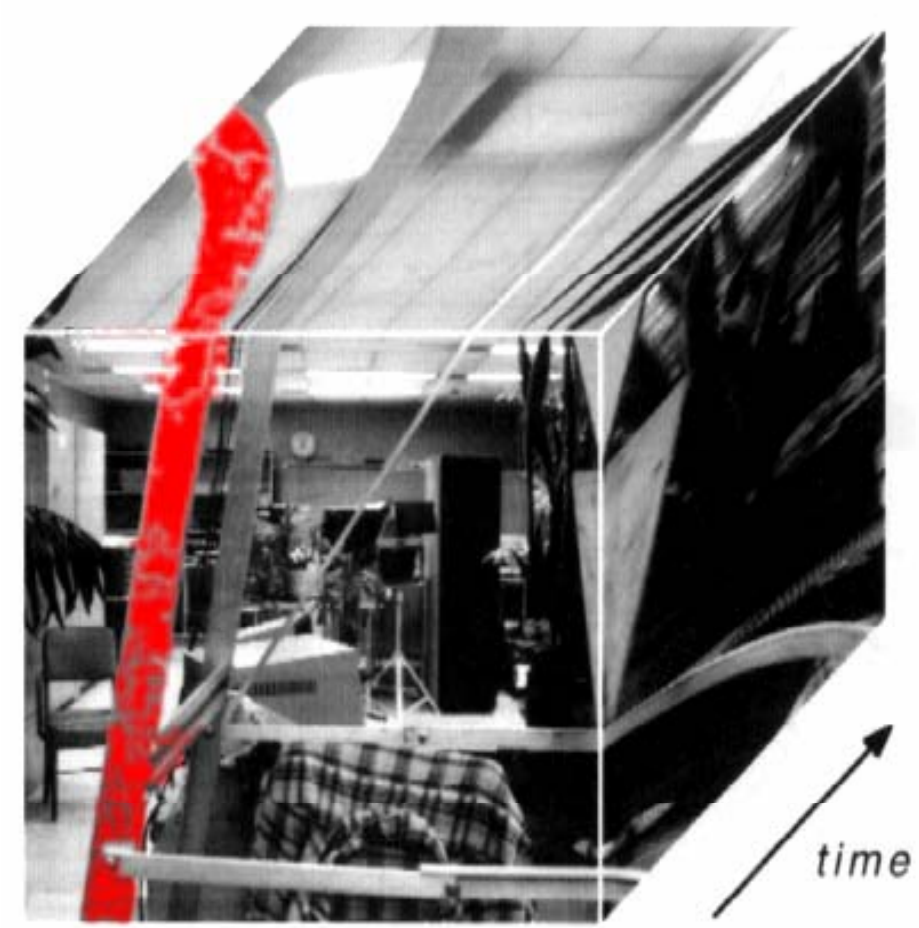
- Motion of an observer relative to the environment:
  - Information about **movement** of viewer
  - **Depth information** of the environment (cf. stereo)
  - Motion parallax = **Motion disparity**
- Problem: direction and amount of camera movement
  - Motion field **estimation**
  - Motion field **analysis**
  - Shift of viewpoint => motion of objects



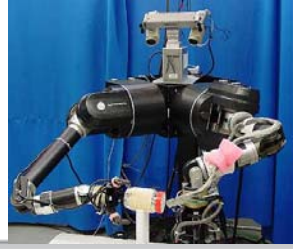
Assumptions for Motion



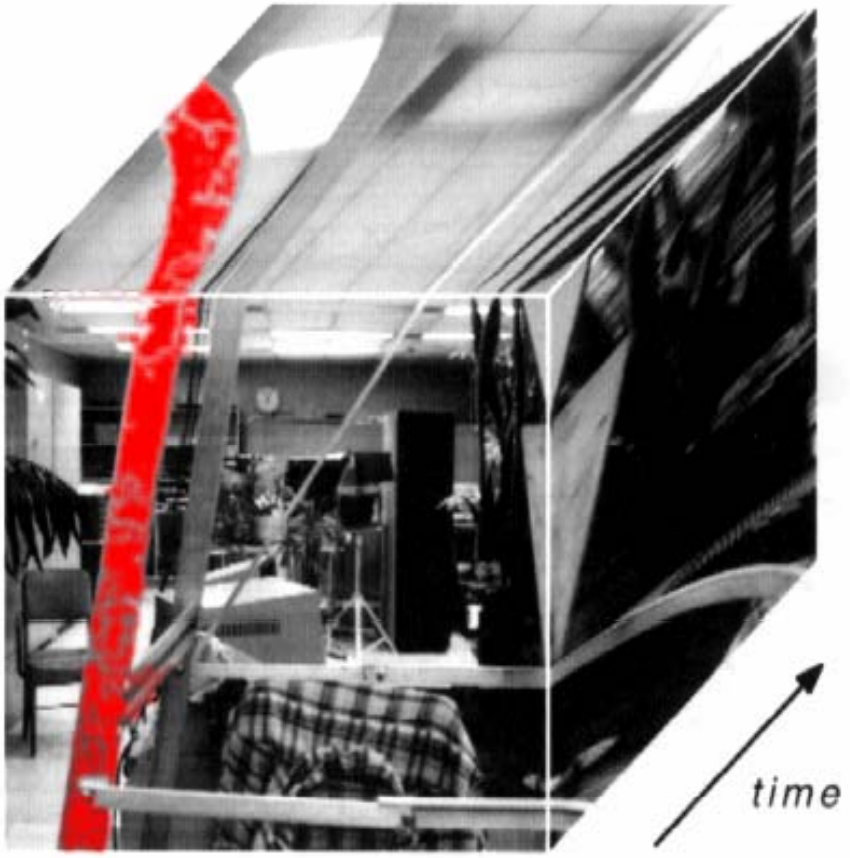
- Camera moves uniformly
- Camera moves along a displacement vector
- Time intervals between images are the same
- Objects are stationary
- Objects may not change temporarily
- Third Dimension = Time dimension
- Spatio-temporal greyscale
  - Spatio = image space
  - Temporal = time space



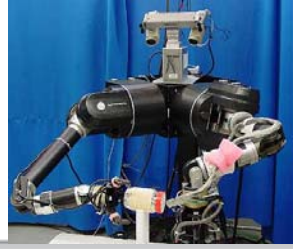
Motion – Range Image Computation



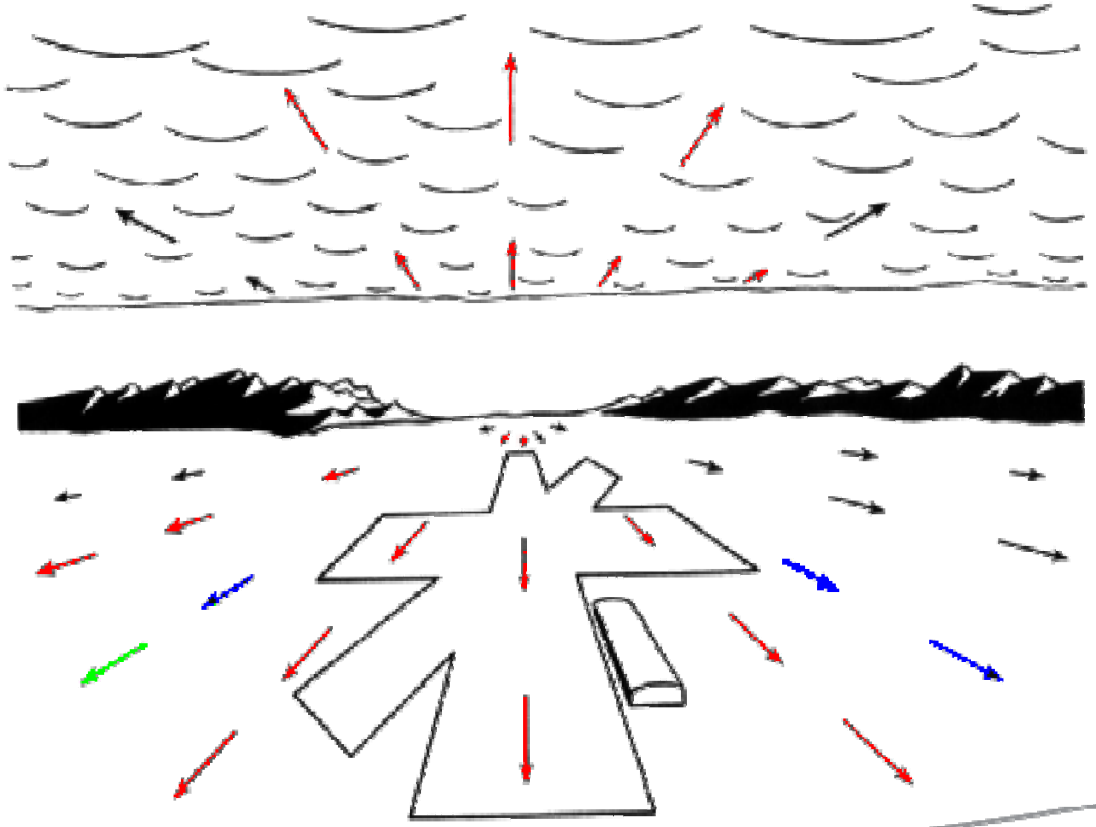
- Range Image Computation possible if (prerequisites):
  - Knowledge of the direction of movement of the camera
  - Knowledge of the speed of the camera
- Algorithm: **Geometric Stereo**



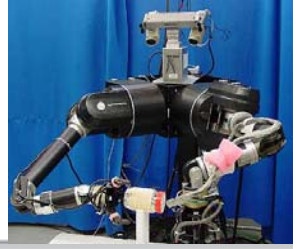
Motion Field



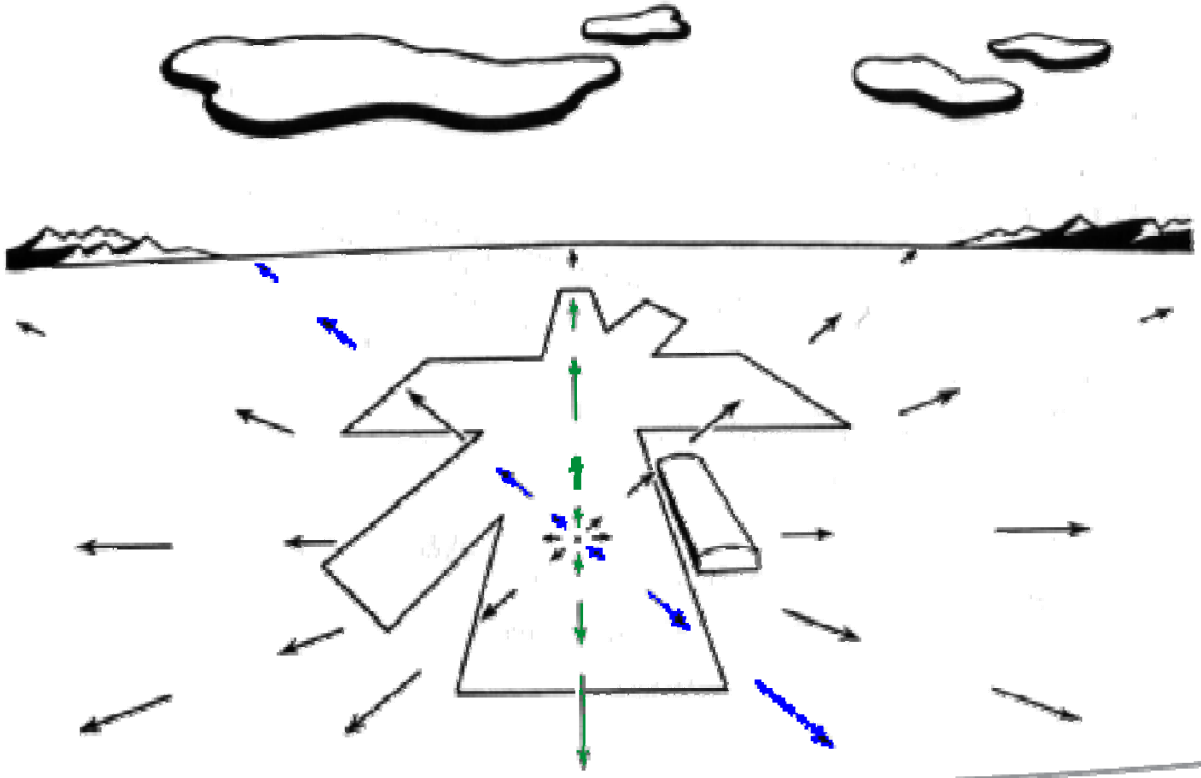
- The motion field of a camera which is in motion relative to the scene is characterized by vectors representing the motion of the corresponding scene points.
- Camera that does not rotate:
  - Vectors point radially to or from a focus
  - FOE: Focus Of Expansion
  - FOC: Focus Of Contraction
  - Point where the motion vector of the camera intersects image plane



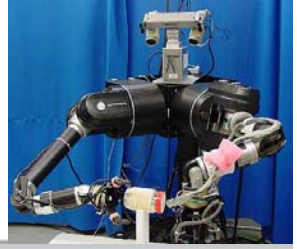
Motion Field



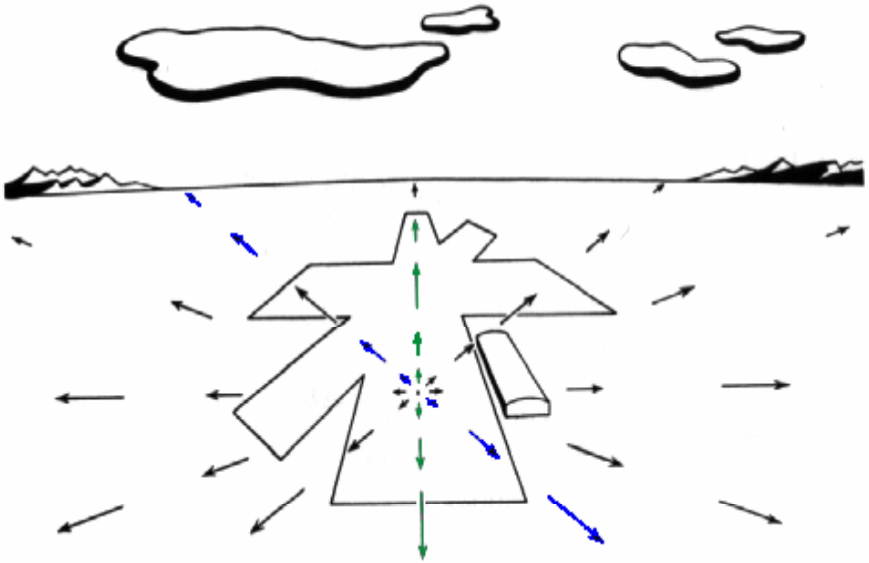
- Length of the vector is:
  - inversely proportional to the distance of the point
  - proportional to the sine between the direction in which the point lies and direction of movement of the camera



Motion Vector

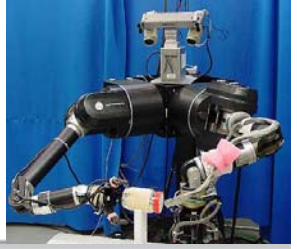


- Determination of the motion vector through image plane velocity
- Determination of the FOE (divergence of vectors = forward movement) or FOC (convergence of the vectors = backward motion)

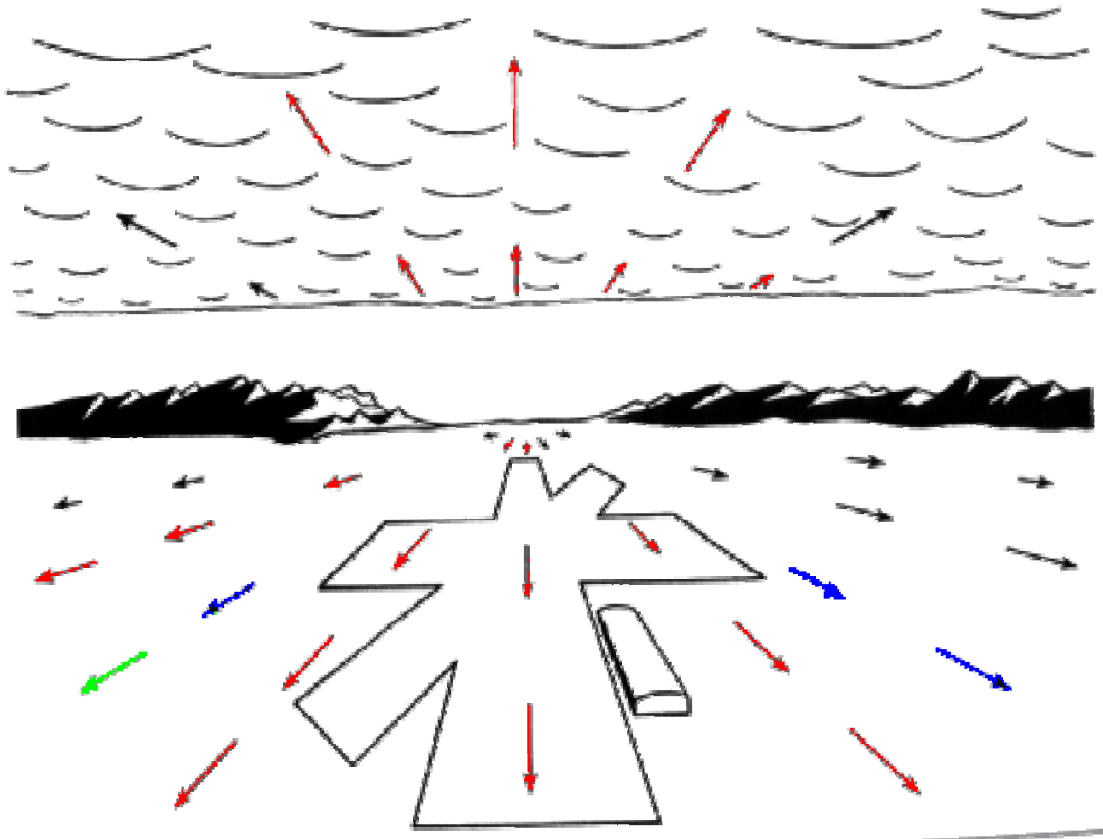


- Analogy to Stereo:
  - Baseline of the stereo images => Any projection of a scene point must move along the epipolar line
  - converge at the epipole = FOE or FOC
  - Caution! Not every point that has no vector magnitude is FOE (FOC) => infinite distance = no movement

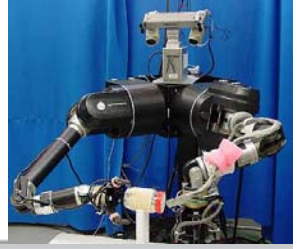
FoE and FoC



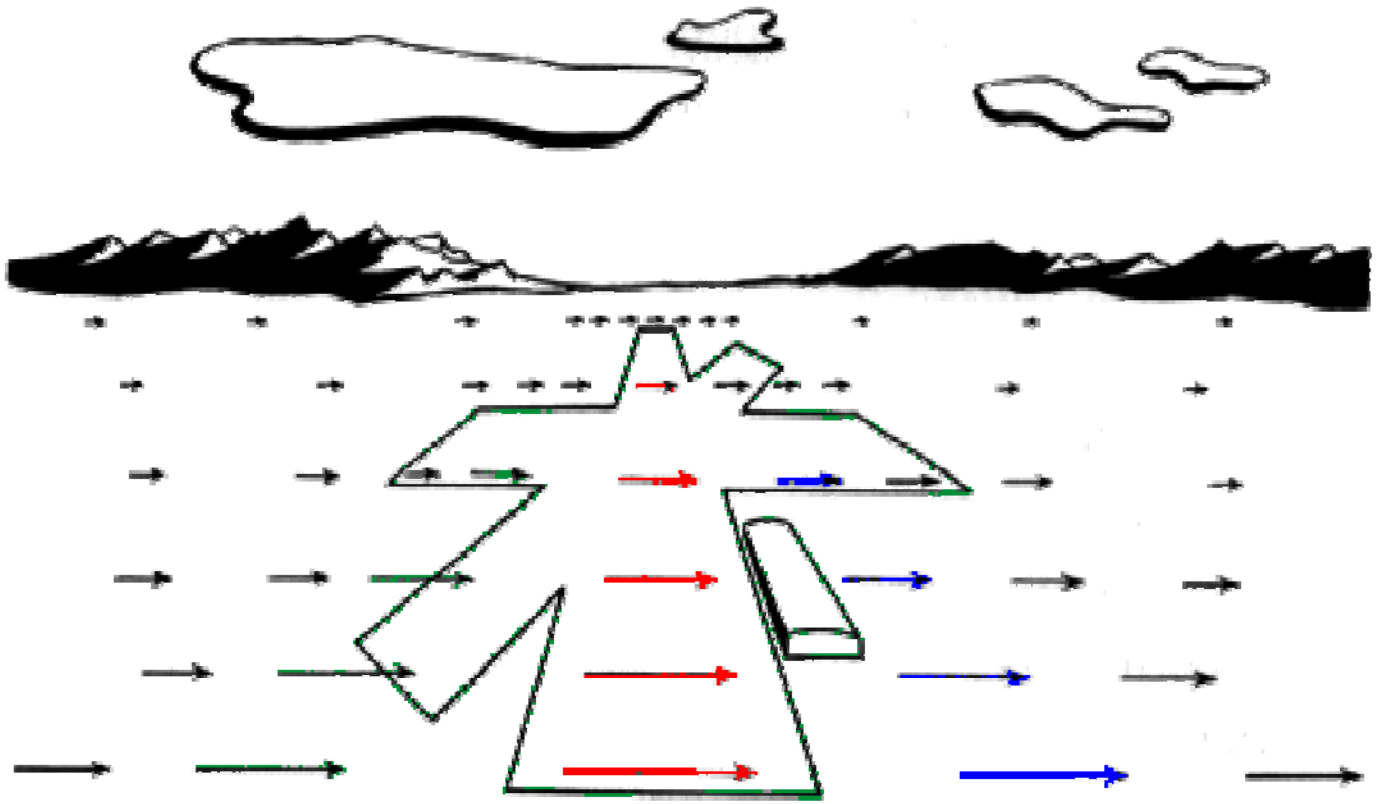
- Intersection between moving direction and image plane
- Amount of movement = 0
- Convergence (divergence) of all motion vectors.



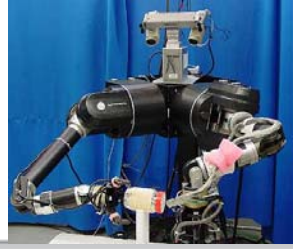
Special Case: Stereo



- FOE and FOC are at infinity

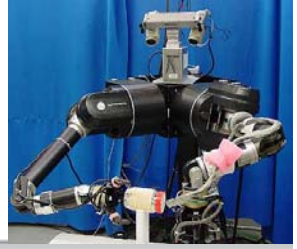


Motion Field Estimation

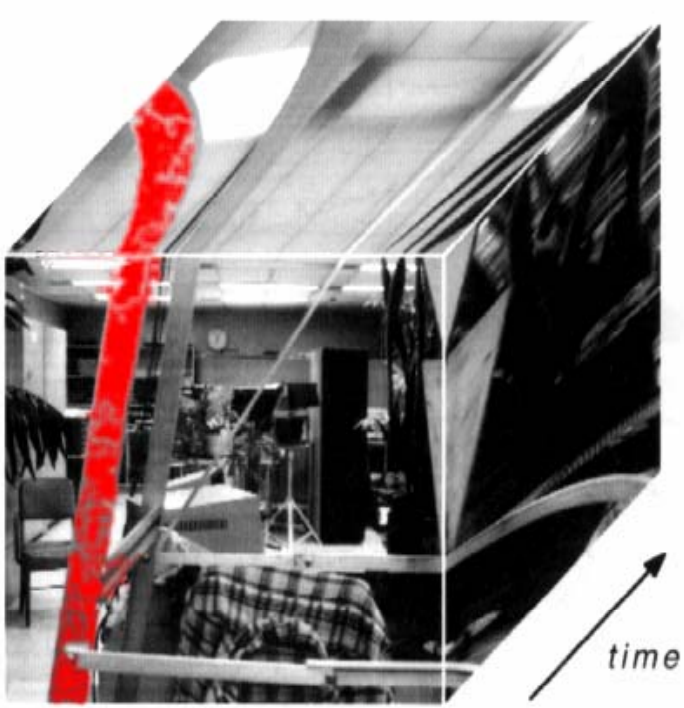


- Problem: Determination of corresponding points in two images
  - sparsely occupied vector field
  - same problem as stereo - just moving direction of the camera not known
  - epipolar line not known in the beginning
- In order to find correspondence between images:
  - high temporal sampling = low differences
  - either unchanged intensities in both images or unchanged edges in both images

Motion Field Estimation

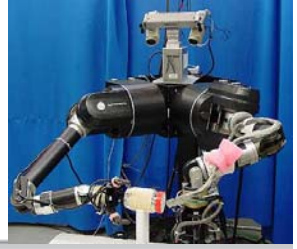


- Similar to area-based and feature-based stereo:
  - Motion: Intensity Flow or Edge Flow (Optical Flow)
  - But: In Motion epipolar lines are not known!
- All Isobrightness lines or image edges have to be tracked
  - Combination of the two assumptions: unchanged intensities and image edges

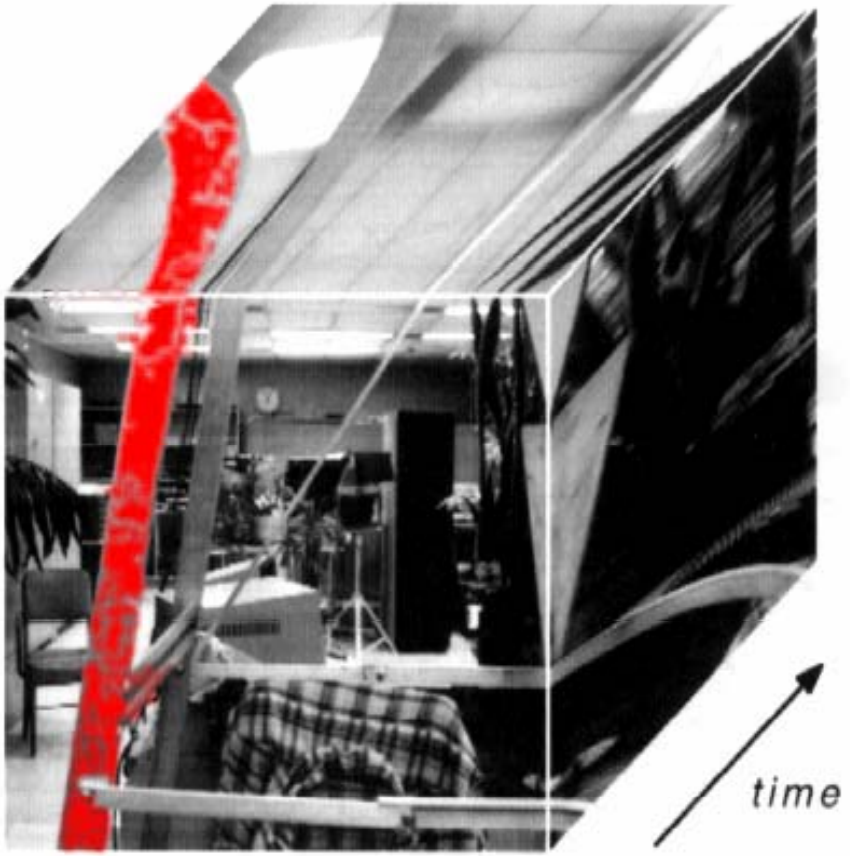


Optical Flow

Motion – Range Image Computation



- Range Image Computation possible if (prerequisites):
  - Knowledge of the direction of movement of the camera
  - Knowledge of the speed of the camera
- Algorithm: **Geometric Stereo**



Optical Flow

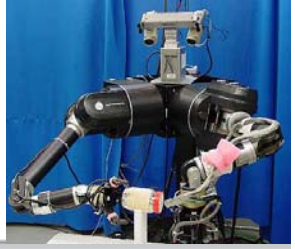
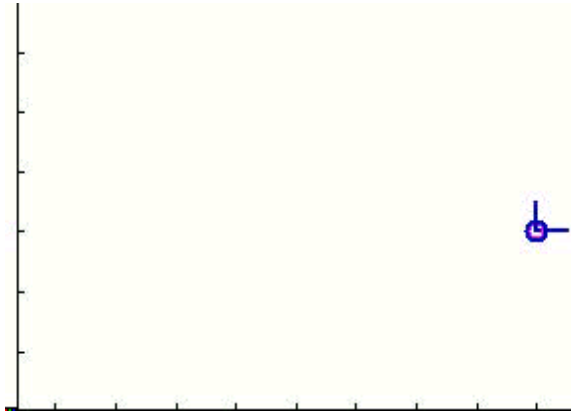


Image sequence
(single camera)

Image tracking

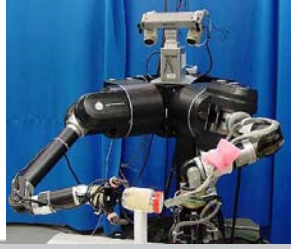
Tracked sequence

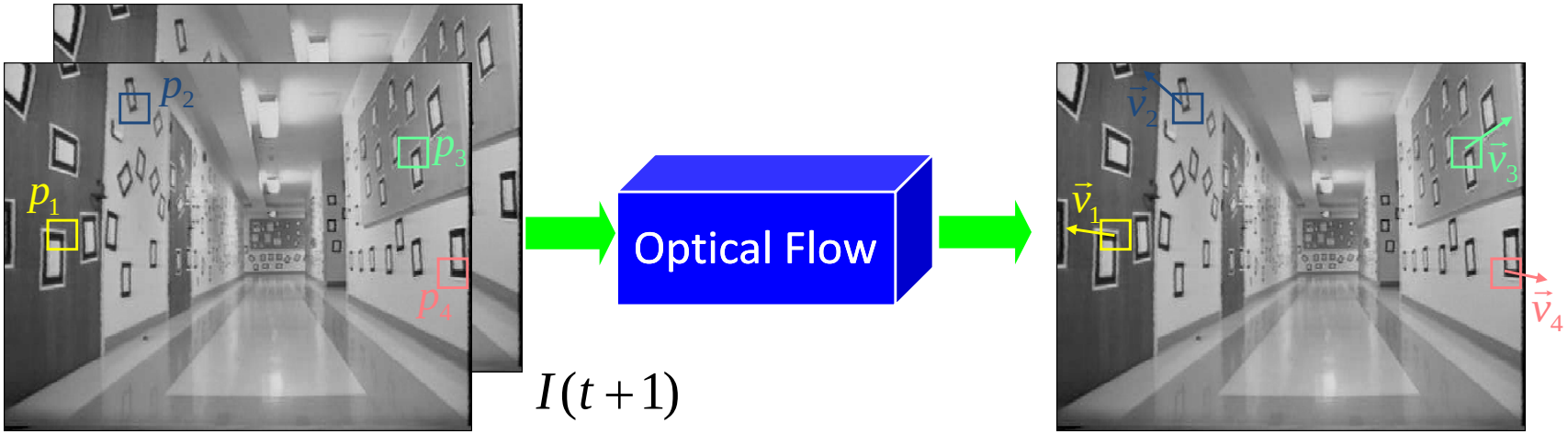
3D computation



3D structure
+
3D trajectory

What is Optical Flow?





$I(t+1)$

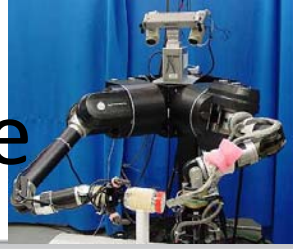
$I(t), \{p_i\}$

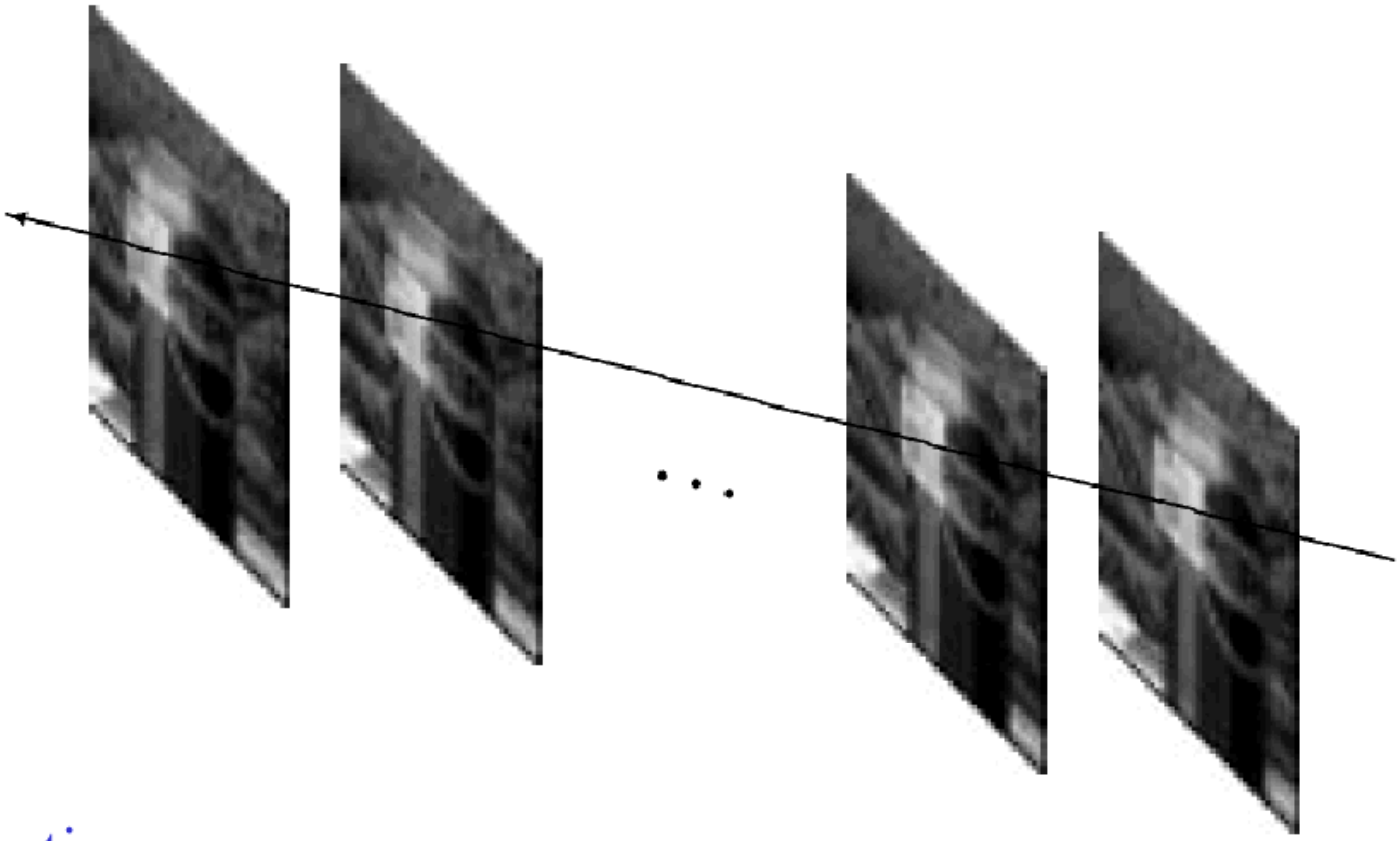
Velocity vectors $\{\vec{v}_i\}$

- Optical flow
  - is the relation of the motion field to 2D displacement of pixel patches on the image plane.
  - the 2D projection of the physical movement of points relative to the observer
- Common assumption:
  - The appearance of the image patches do not change (brightness constancy)

$$I(p_i, t) = I(p_i + \vec{v}_i, t+1)$$

Optical Flow Assumption: Temporal Persistence





Assumption:

The image motion of a surface patch changes gradually over time.

Structure from Motion Example