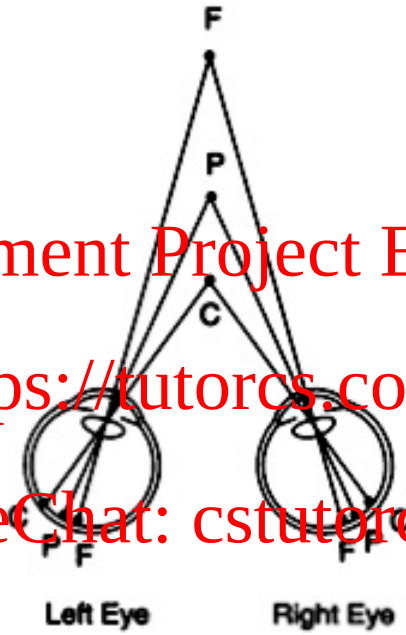
M30242 Graphics and Computer Vision

Lecture 11 Stereo Vision

Overview

- Intro to binocular vision
- Setup and terminology
- Principle of triangulation
- Correspondence matching problem
- Matching algorithm

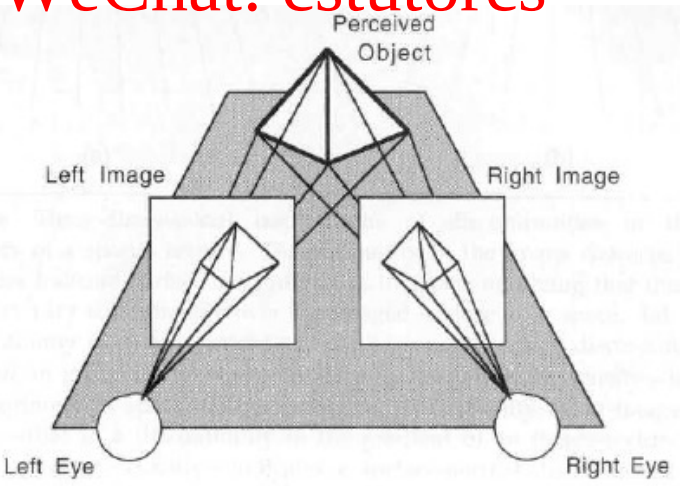
Stereo Vision



- Binocular *disparity* (displacement) arises when a given point in the external world does not project to the same positions on the left and right retinae.
- Depth can be computed from disparity information.

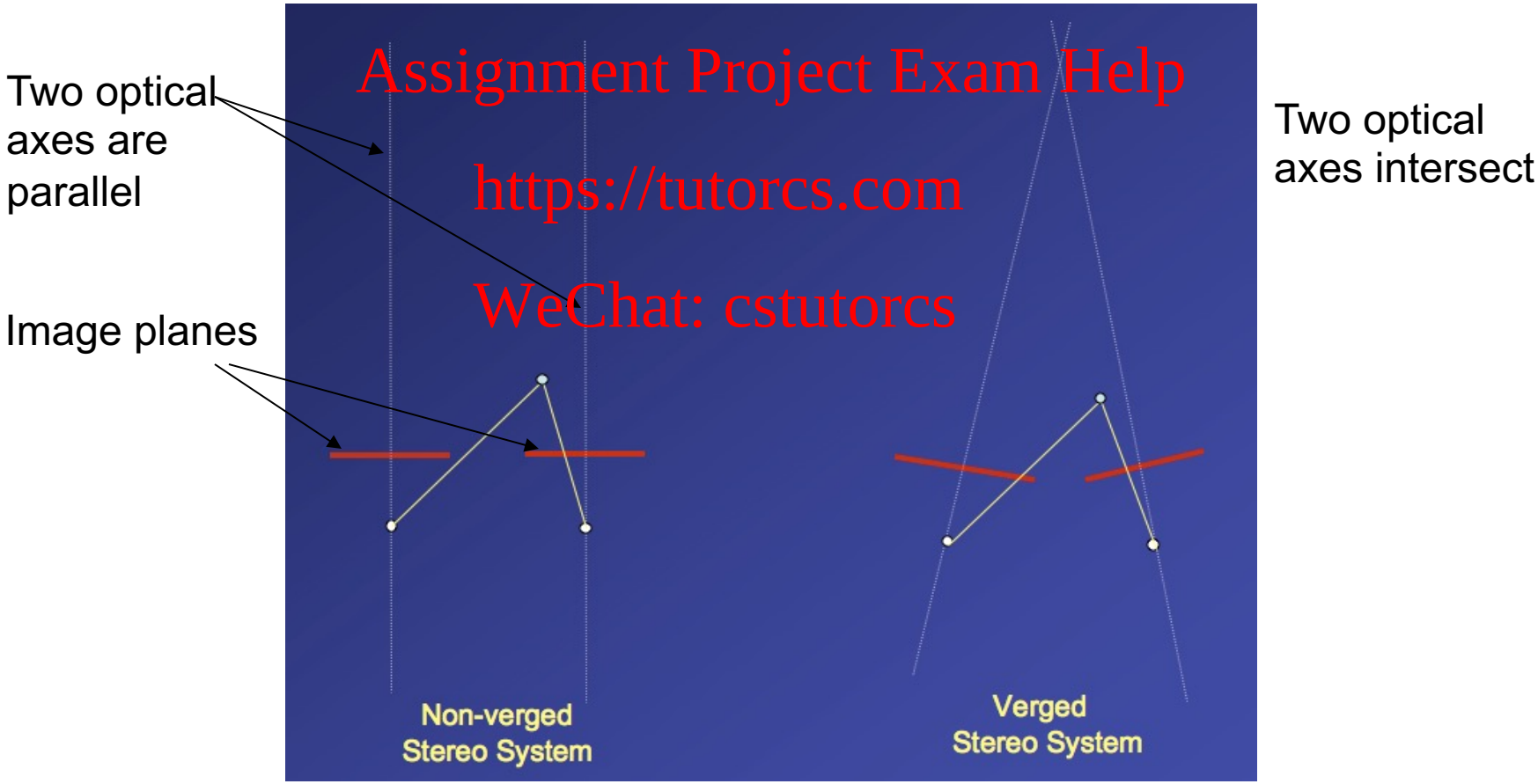
Binocular Vision

- When a vision system uses disparity between two images to calculate the depth, we call it binocular vision system.
- The two images can be acquired by two separate cameras, or by a single camera that takes the left and right images at different positions.
- For convenience, we assume two separate cameras are used.



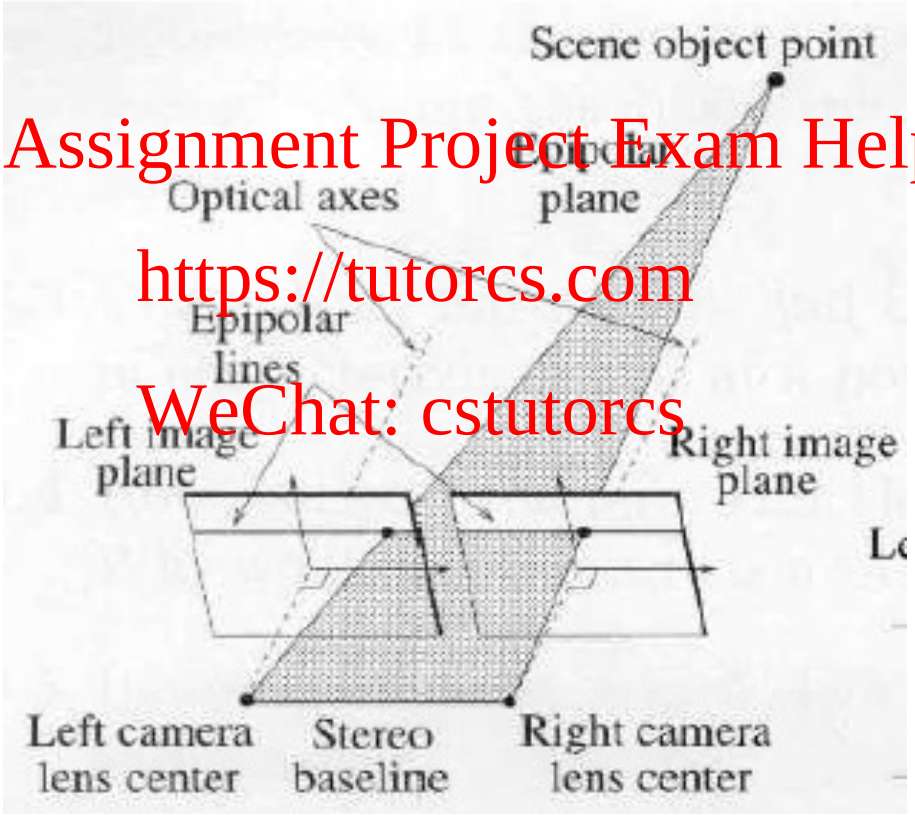
Camera Configurations

- Two types of binocular vision systems



Non-verged Configuration

Non-verged system



Verged Configuration

Verged system

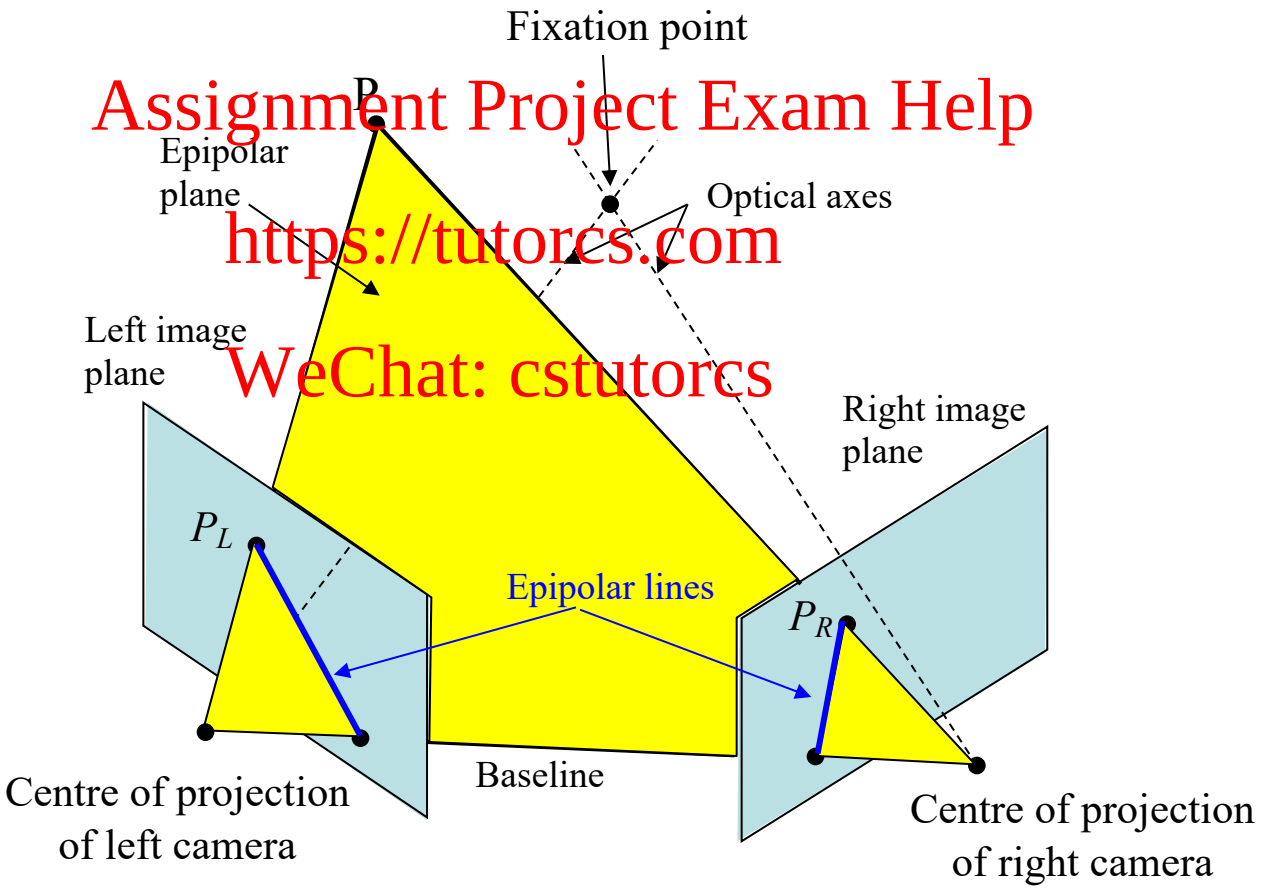


Image P_L and P_R are called a conjugate pair

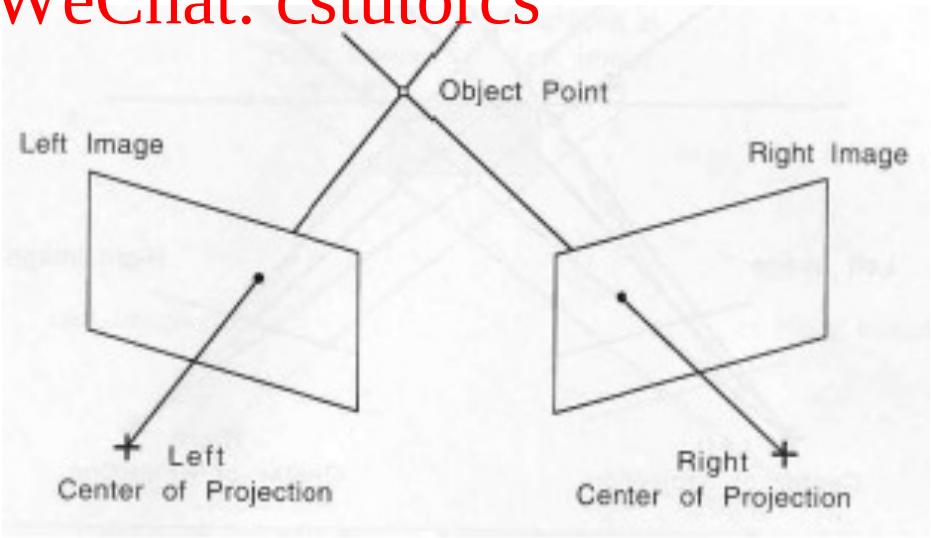
Terminology

Summary

- Fixation point: the point of intersection of the two optical axes.
- Baseline: the distance between the centers of projection.
- Epipolar plane: the plane passing through the centers of projection and a point in the scene.
- Epipolar lines: the intersection of the epipolar plane with the image planes.
- Conjugate pair: any point in the scene that is visible in both cameras will be projected to a pair of image points in the two images.
- Disparity: the distance between corresponding image points when the two images are superimposed.

Depth Recovery

- Binocular stereo vision determines the position of a point in space by finding *the intersection* of the lines of projection of the point in two cameras. The line of projection is the line that passes through the image of the point and the centre of projection of the camera.
- The intersection point can be found by the principle of *triangulation*.



Triangulation (Non-verged)

Question: find the position of P, i.e., the values of Z_p, and X_p

We have:

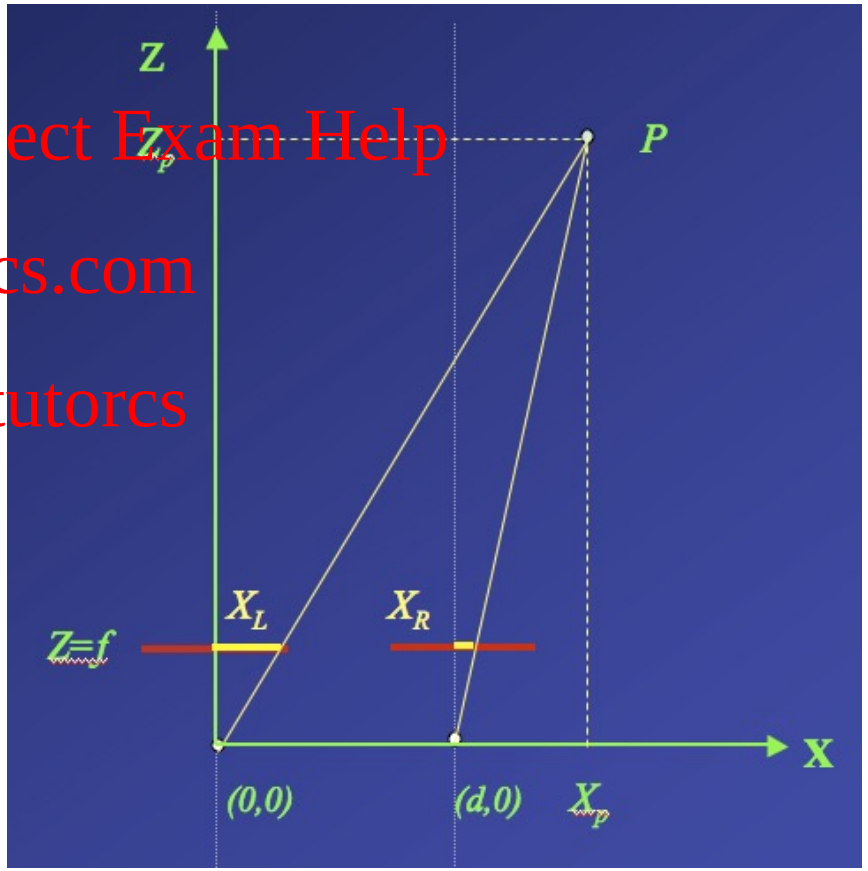
$$\frac{Z_p}{X_p} = \frac{f}{X_L} \qquad \frac{Z_p}{X_p - d} = \frac{f}{X_R}$$

Where f is the camera focal length, and d the length of baseline

$$Z_p = \frac{fX_P}{X_L} = \frac{f(X_p - d)}{X_R}$$

$$X_P = \frac{dX_L}{X_L - X_R} \qquad Z_P = \frac{fd}{X_L - X_R}$$

$X_L - X_R$ =disparity

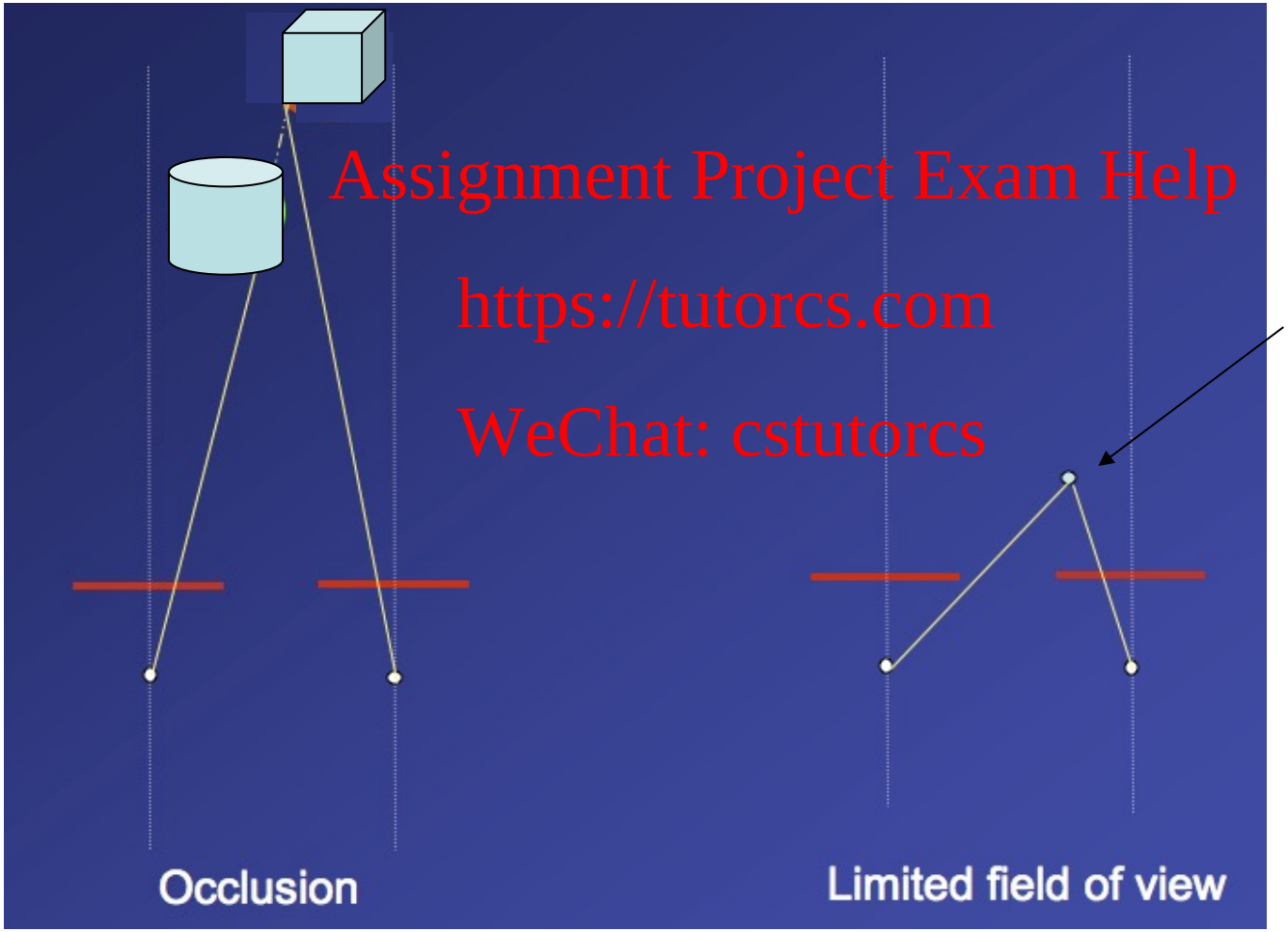


Therefore, if disparity is known, then Z_p, and X_p are known

Correspondence Problem

- To decide the disparities, one needs to have efficient & reliable methods for finding the matching the features in the left and right images.
- This is the so-called correspondence matching problem of stereo (and multiple view) vision: Given a point/feature, $\boldsymbol{p}$, in the left image, find its conjugate point/feature in the right image.
- The disparities of all matching features (or all image pixels) form a *disparity map.*
- By triangulation, the disparity map can be converted to a *depth map* of the scene (in terms of the distances of all feature points from the camera or from any other reference objects/systems).

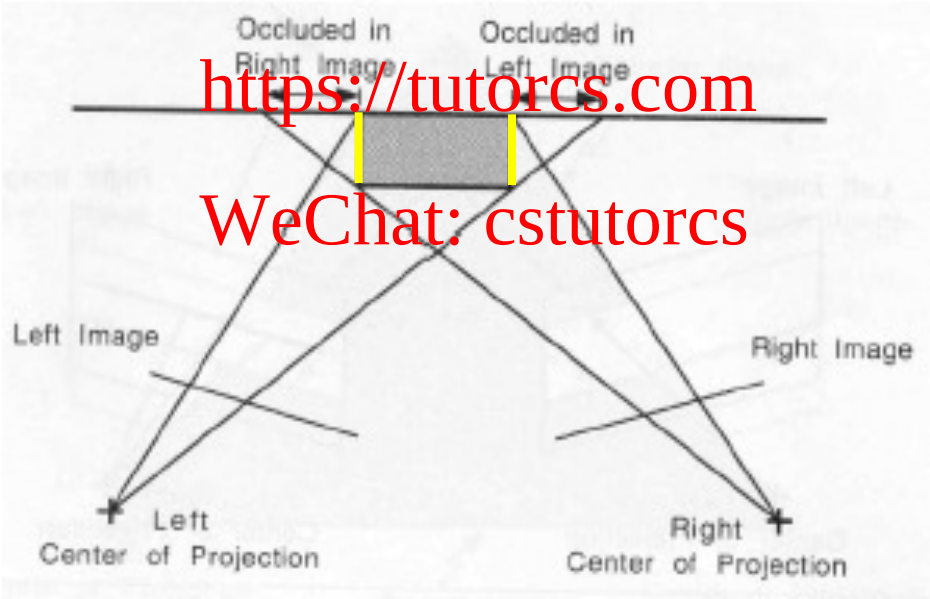
Correspondence Matching is Hard



Objects close to camera may be visible only to one of the cameras only

Cont'd

- Self-occlusion



Cont'd

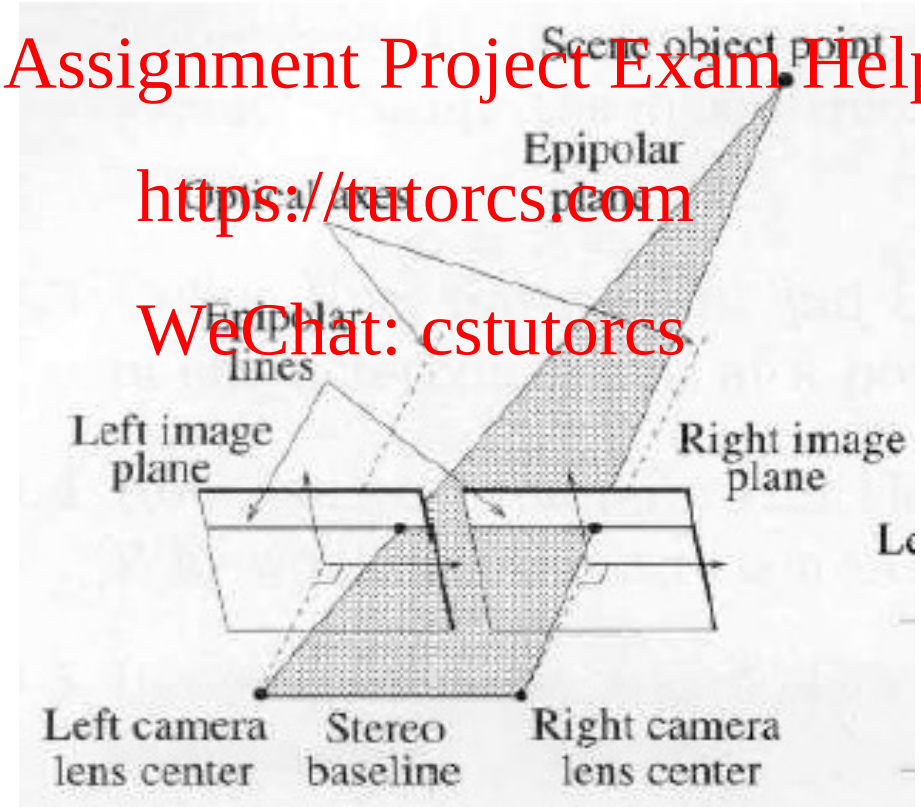
- Photometric issues:
  - Image noises,
  - Specular highlights, etc.
- Issues of surface structures/attributes
  - Lack of texture,
  - Repeating texture/texels.

Useful Constraints

- Correspondence matching is the core problem of stereo vision.
- Searching for correspondence by brutal force is very inefficient and not suitable for real-time applications.
- Research has revealed a few constraints that can simplify and speed up correspondence matching, e.g.,
  - Epipolar constraint,
  - Continuity constraint,
  - Disparity constraint and others.

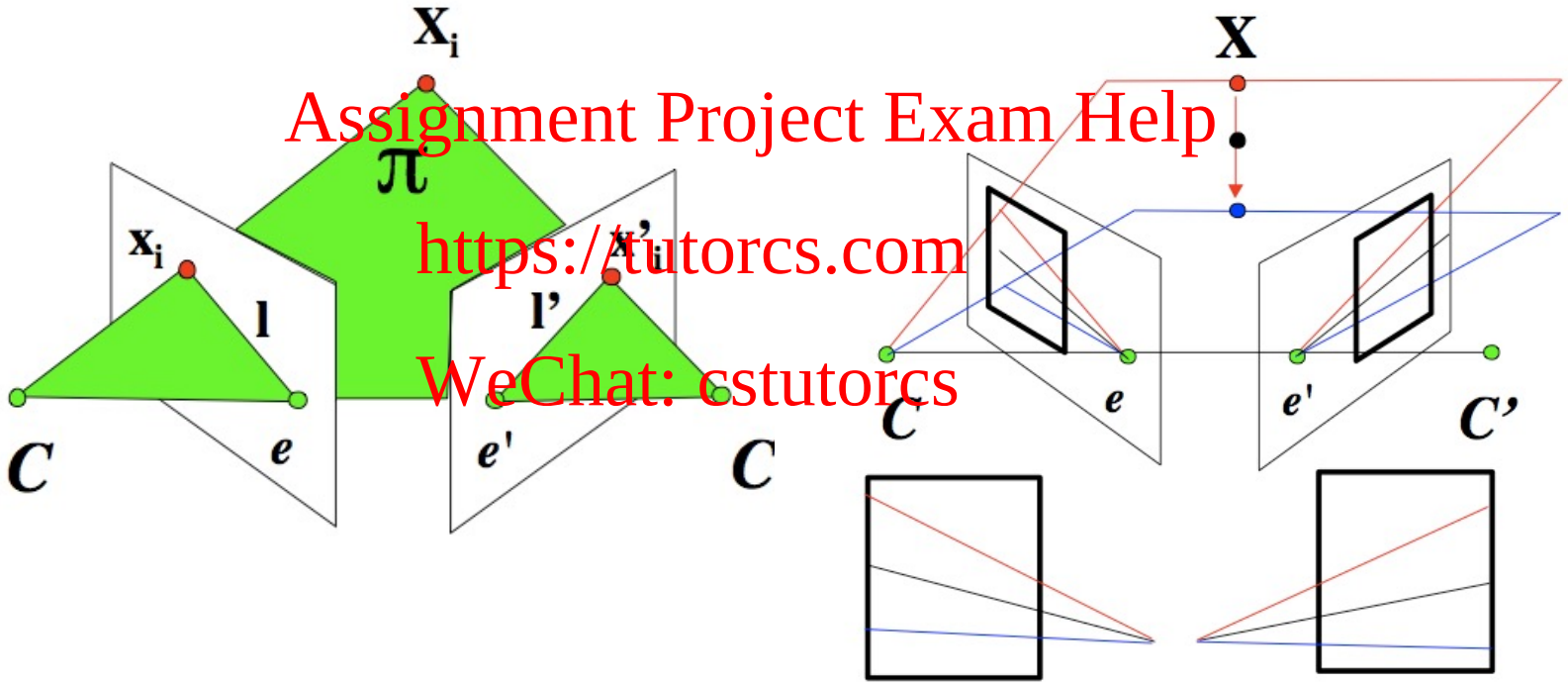
Epipolar Constraint

- Correspondence can only be found on the corresponding epipolar lines.



Non-verged configuration

Cont'd

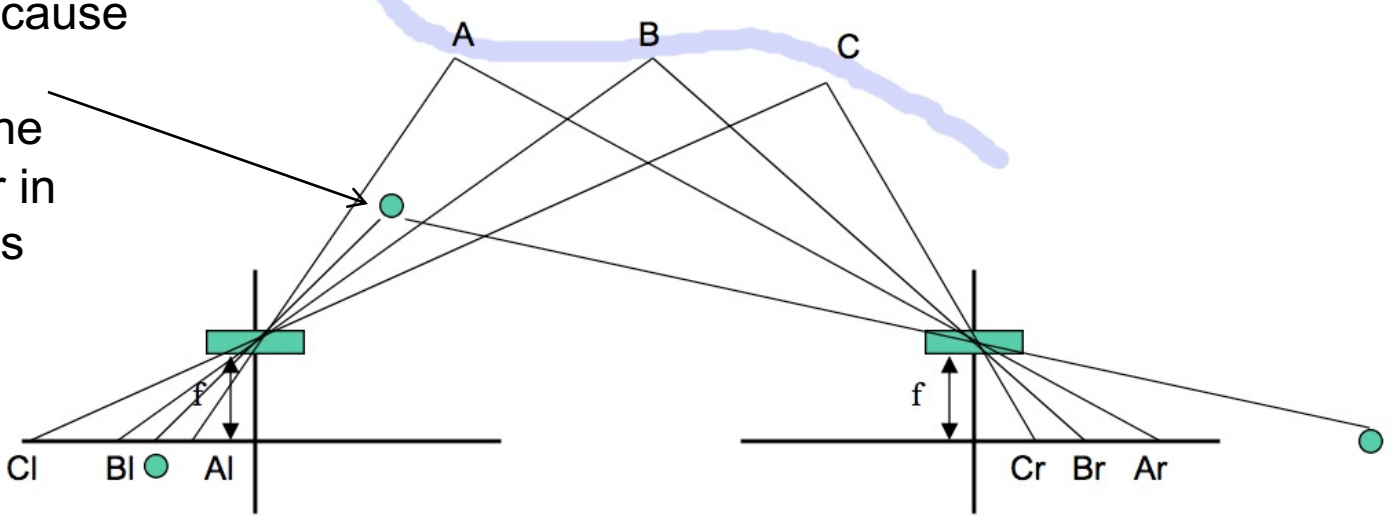


Verged System

Continuity Constraint

- Continuity constraint: if we are looking at a continuous surface, the images of points along a given epipolar line will follow the same order.

This point is unlikely on the surface, because it does not appear in the same order in both images



Disparity Constraint

- Disparity limit: limiting the range of disparity can significantly reduce the search space
- Disparity gradient (change rate): disparity changes slowly over most of the image. Exceptions occur at and near the occluding boundaries where we have either discontinuities in disparity or large disparity gradients as the surface recedes away from sight.

Other Constraints

- There exist also problem-specific constraints, e.g., colours, features, etc.
- Developing efficient methods for correspondence matching is still a research problem.

Correspondence Searching

- Various algorithms/techniques have been used for correspondence matching.

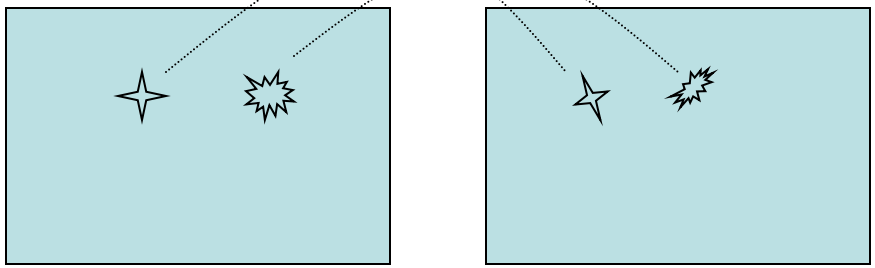
- Two main classes of algorithms:

  - Feature-based: attempt to establish correspondence by matching a sparse sets of image features.

  - Region-based: attempt to establish correspondence by matching a selected region in one image to a region in the other image.

Feature-Based Matching

- Matching is based on comparing the features (e.g., corners).
- The matching criteria are usually measured against feature vectors (e.g., the sign and orientation of gradient vectors)
- Searching for matches within a given disparity range.



- Characteristic: reliable & efficient but a dense set of features are hard to find or define.

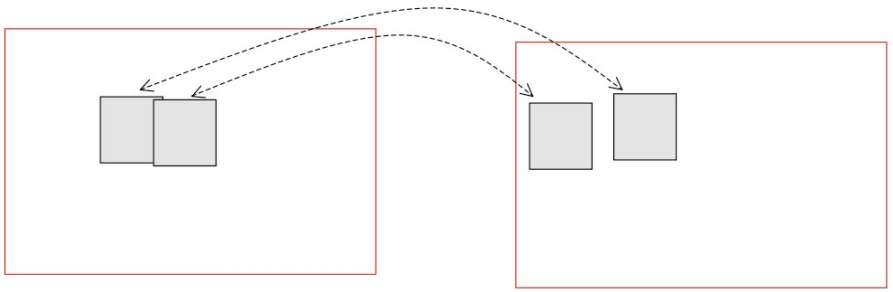
Region-Based Matching

- Matching is done by selecting a region (regardless the feature) in one image and trying to find a match for it in the second image by minimising/maximizing some measures, e.g.,
  - Minimising the sum of squared difference (SSD)

$$\sum (I_1(i,j) - I_2(i,j+d))^2$$

  - Maximising the normalized cross-correlation (NCC)

$$\frac{\sum_{i,j}[I_1(i,j) - \bar{I}_1][(I_2(i,j+d) - \bar{I}_2]}{\sqrt{\sum_{i,j}[I_1(i,j) - \bar{I}_1]^2 \sum_{i,j}[I_2(i,j+d) - \bar{I}_2]^2}}$$



Where capped I_1 and I_2 are the means of pixels of the template and the searched region

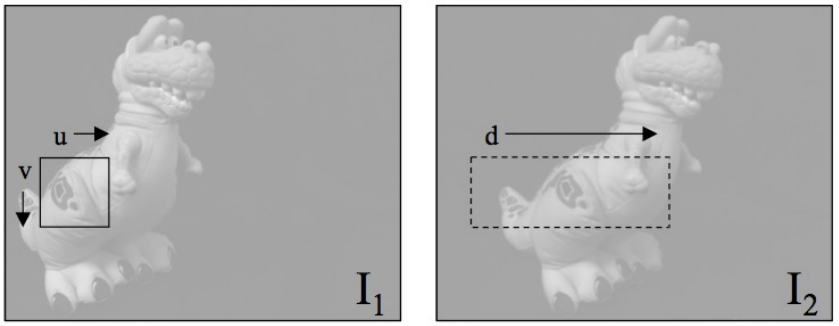
Feature-based v.s. Region-based

- Feature-based leads to sparse disparity maps.
  - Interpolation to fill the gaps in between.
  - Matching is easier to achieve and accuracy tend to be high if the features are distinctive enough.

- Region-based matching works when there is texture or a variation in surface feature.
  - Easier to implement.
  - Can be sensitive to changes in surface orientation and illumination.

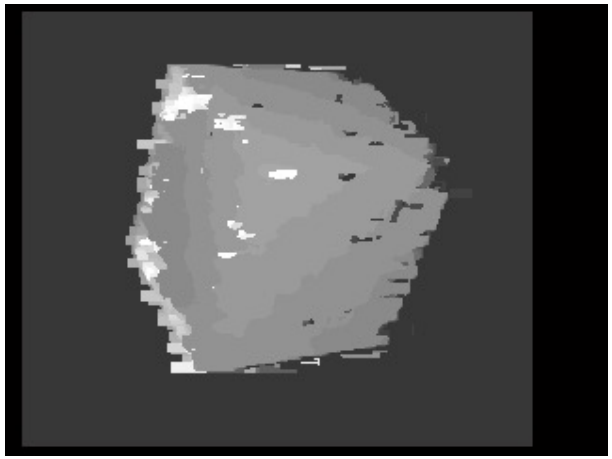
Region-Based Search Algorithm (Pseudocode)

```
for  i = 1:nrows
      for j=1:ncols
            best(i,j) = -1
            for k = mindisparity:maxdisparity
                  c = ComputeMatchMetrics(I1(i,j),I2(i,j+k),winsize)
                  if (c > best(i,j))
                        best(i,j) = c
                        disparities(i,j) = k
                  end
            end
      end
end
```

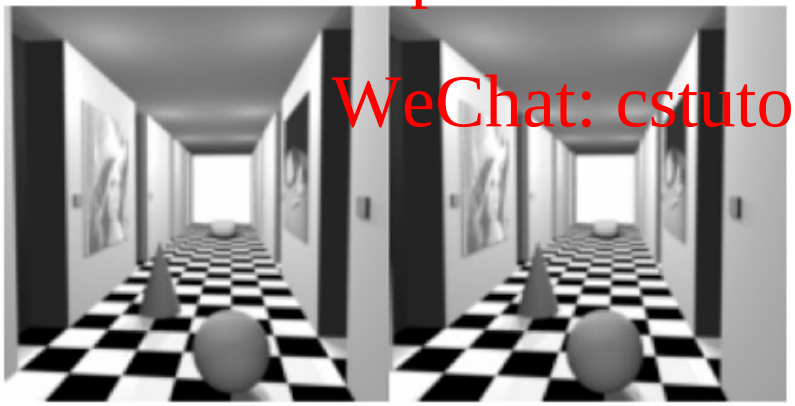
An Example

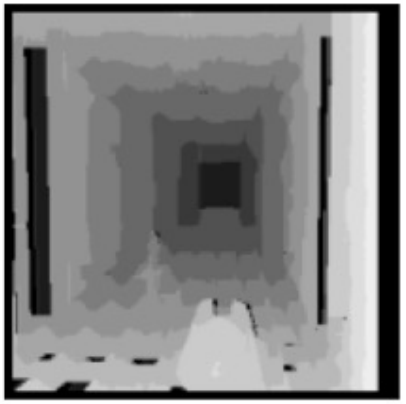
Focal length f = 50mm, baseline = 150mm



Disparity map

More Examples



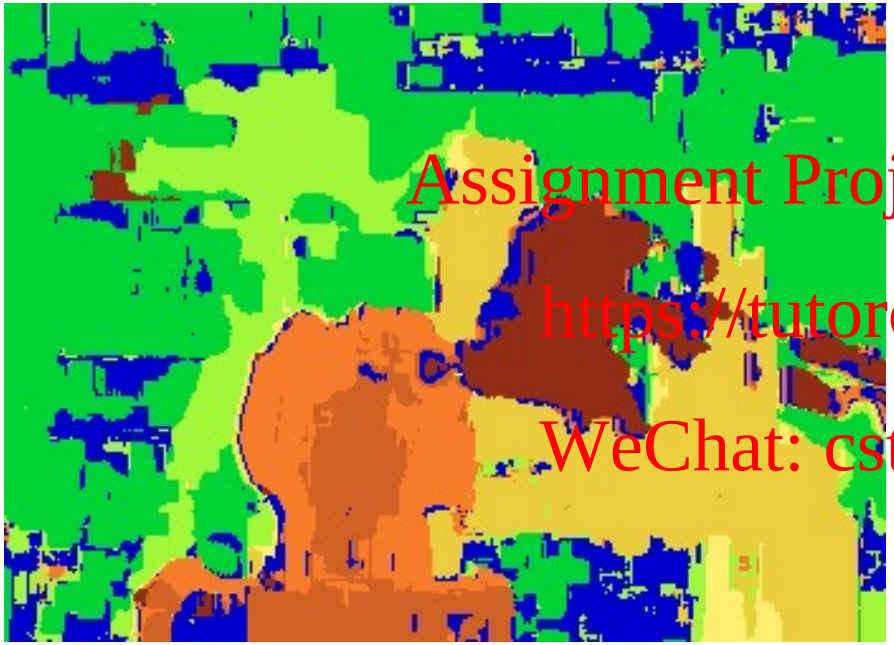


One More Example

An example that had been used for benchmarking stereo vision algorithms

The ground truth

Results of Region-Based Correlation



Result of region-based correlation

The ground truth