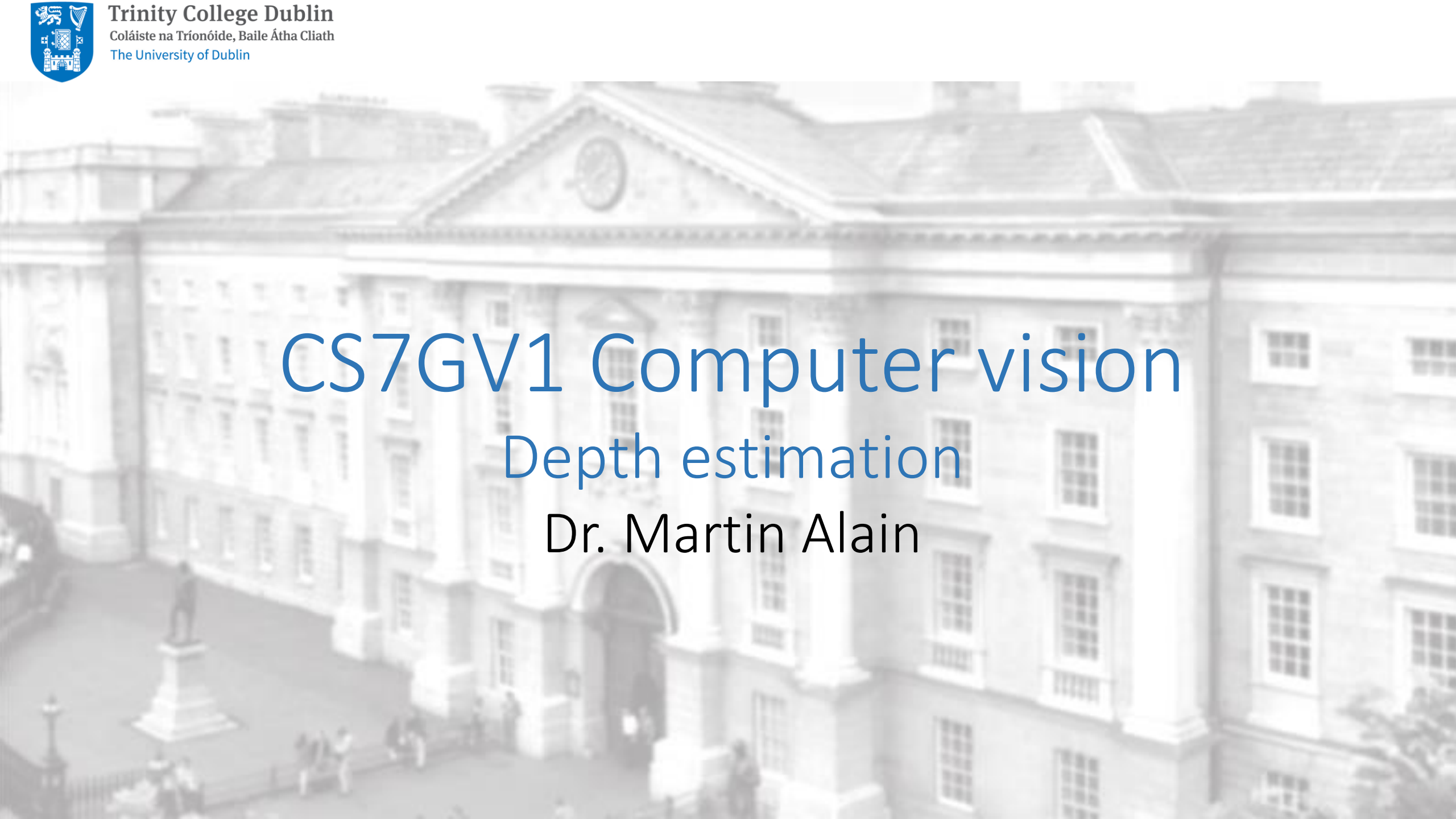


CS7GV1 Computer vision

Depth estimation

Dr. Martin Alain

Introduction – the pinhole camera model

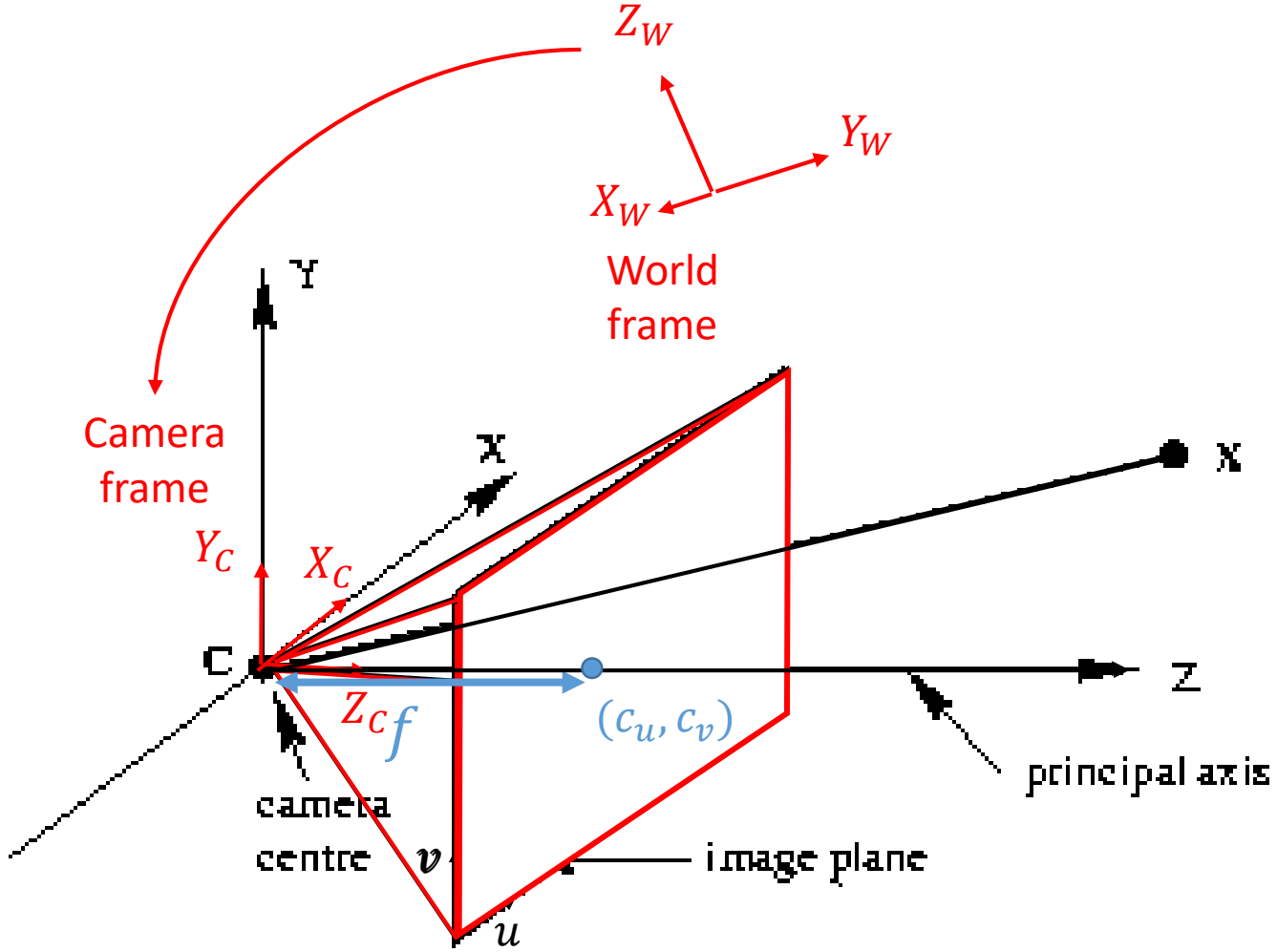
Intrinsic matrix:

$$K = \begin{bmatrix} f_x & s_x & 0 & c_u \\ s_y & f_y & 0 & c_v \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Extrinsinc matrix:

$$[R \mid t] = \begin{bmatrix} r_{1,1} & r_{1,2} & r_{1,3} & t_x \\ r_{2,1} & r_{2,2} & r_{2,3} & t_y \\ r_{3,1} & r_{3,2} & r_{3,3} & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

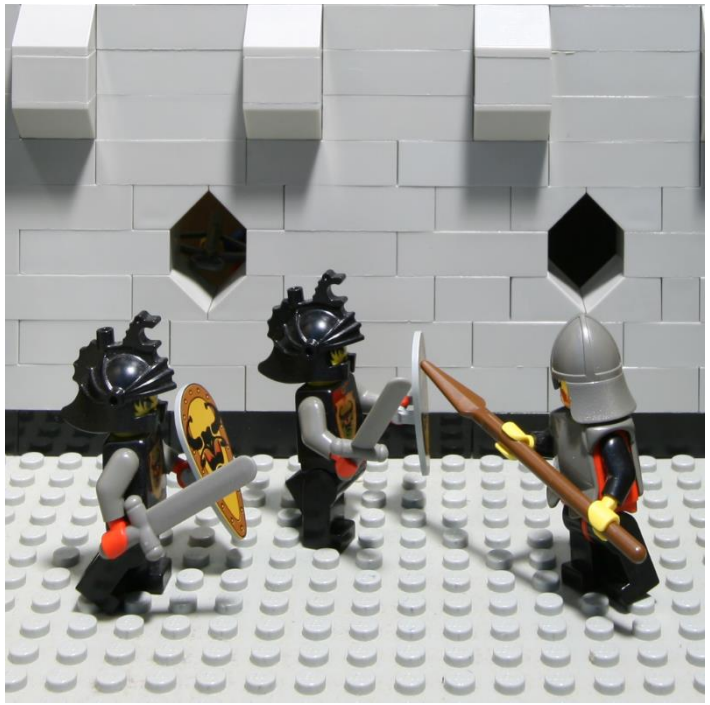
$$K[R \mid t] \begin{bmatrix} X_W \\ Y_W \\ Z_W \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \Rightarrow \begin{bmatrix} u/w \\ v/w \\ 1 \end{bmatrix}$$



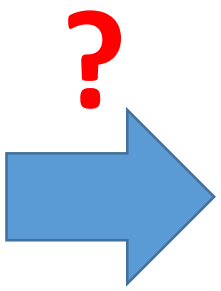
Hartley and Zisserman "Multiple View Geometry in Computer Vision"

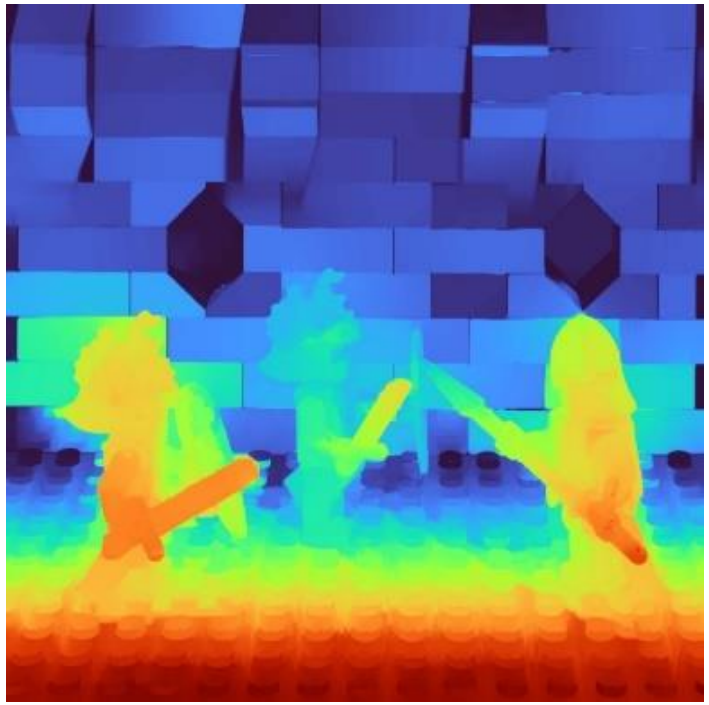
Introduction – depth estimation

- Inverse problem of reconstructing 3D information from 2D images

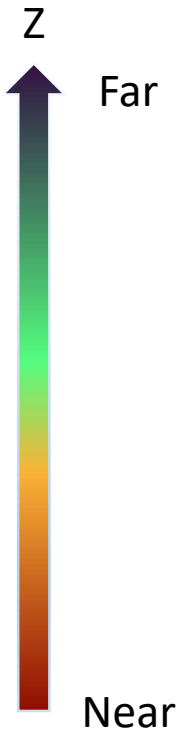


Input image





Depth map



Stereo depth estimation

$$Similar\ triangles:$$

$$\frac{f}{z} = \frac{u_1}{\Delta x_1} = \frac{u_2}{\Delta x_2}$$

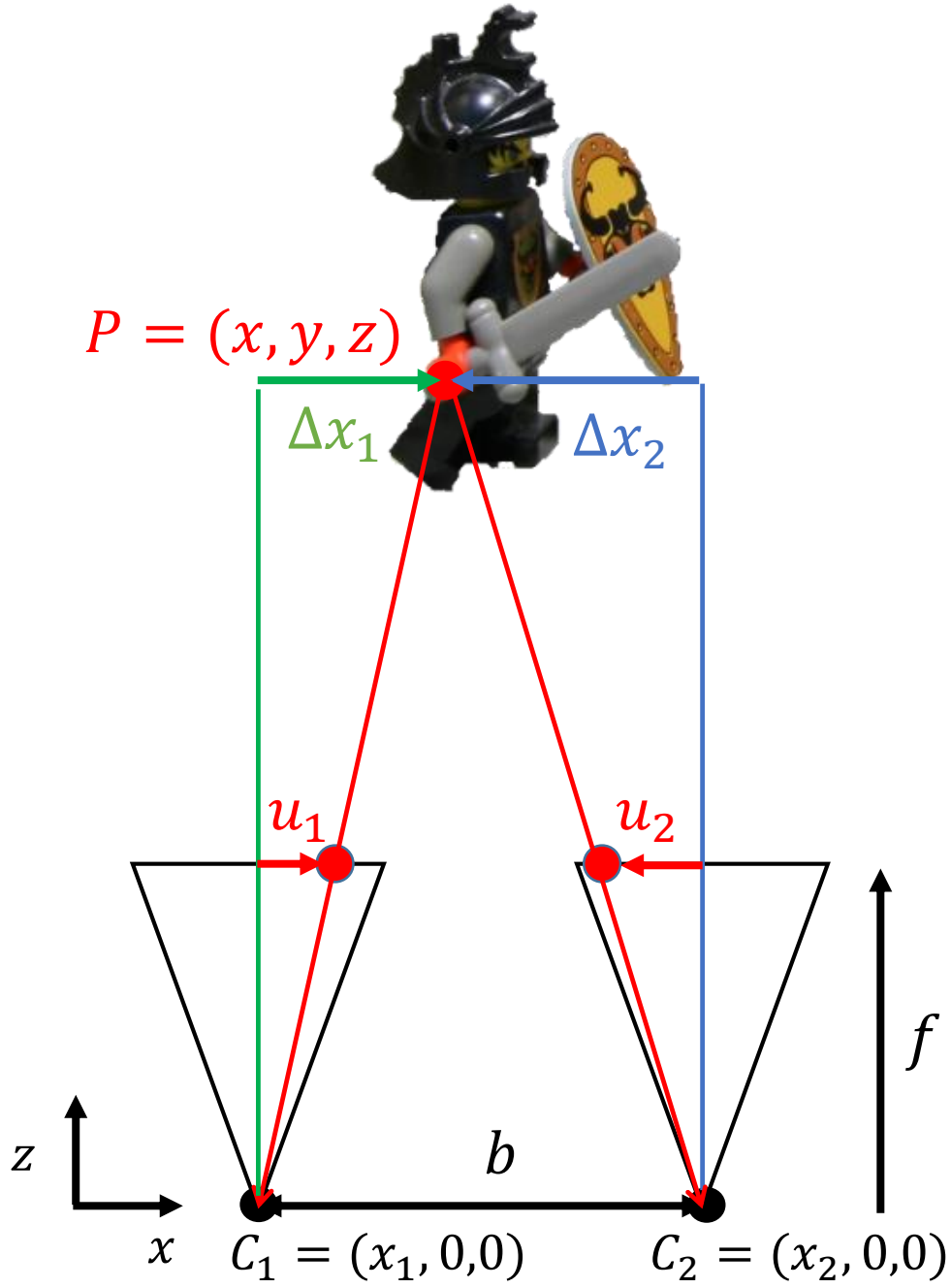
$$Baseline:$$

$$b = \Delta x_1 - \Delta x_2$$

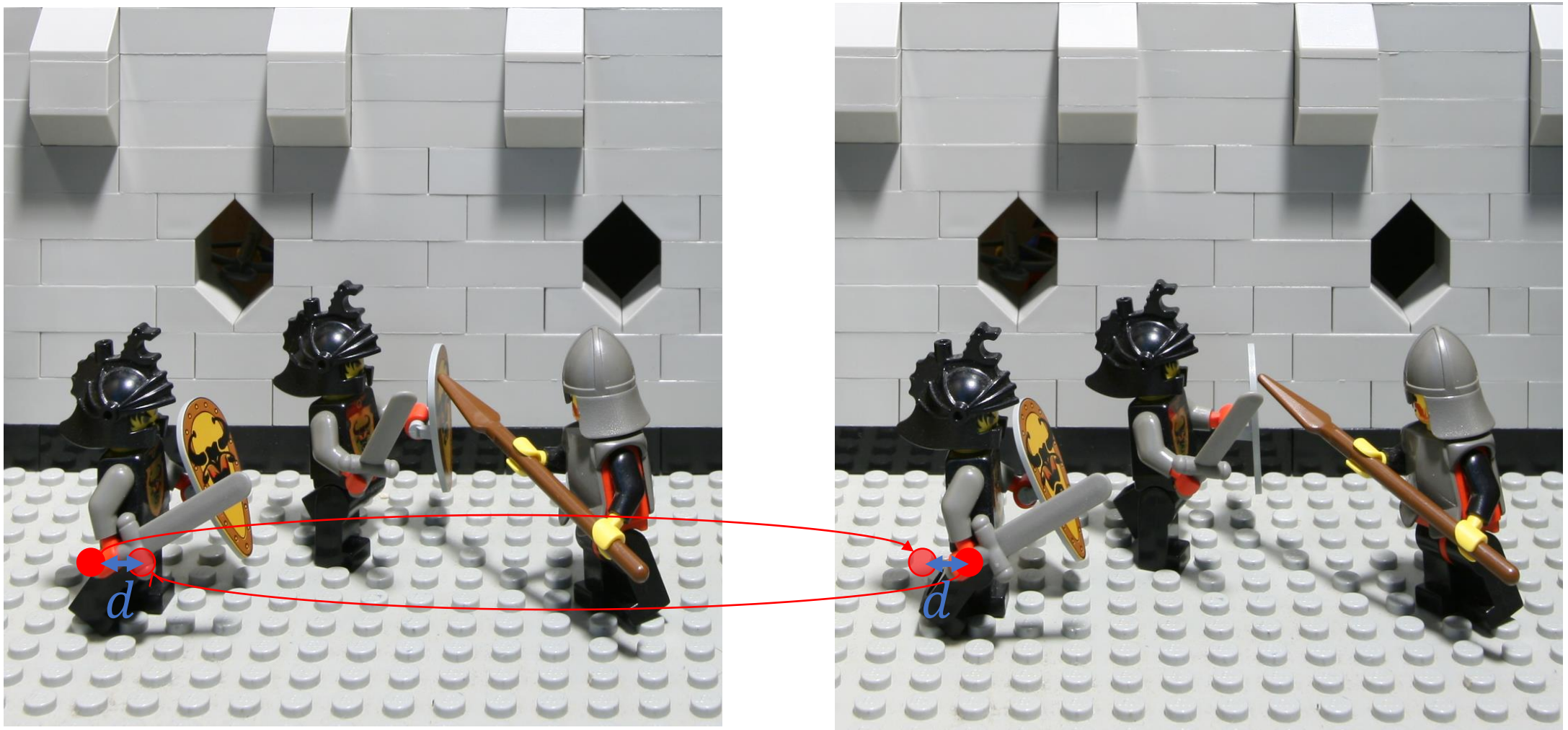
$$= \frac{u_1 z}{f} - \frac{u_2 z}{f}$$

$$= (u_1 - u_2)\frac{z}{f}$$

$$\boxed{\Rightarrow z = \frac{bf}{d}}$$



Stereo disparity estimation



Left image

Right image

Stereo disparity estimation

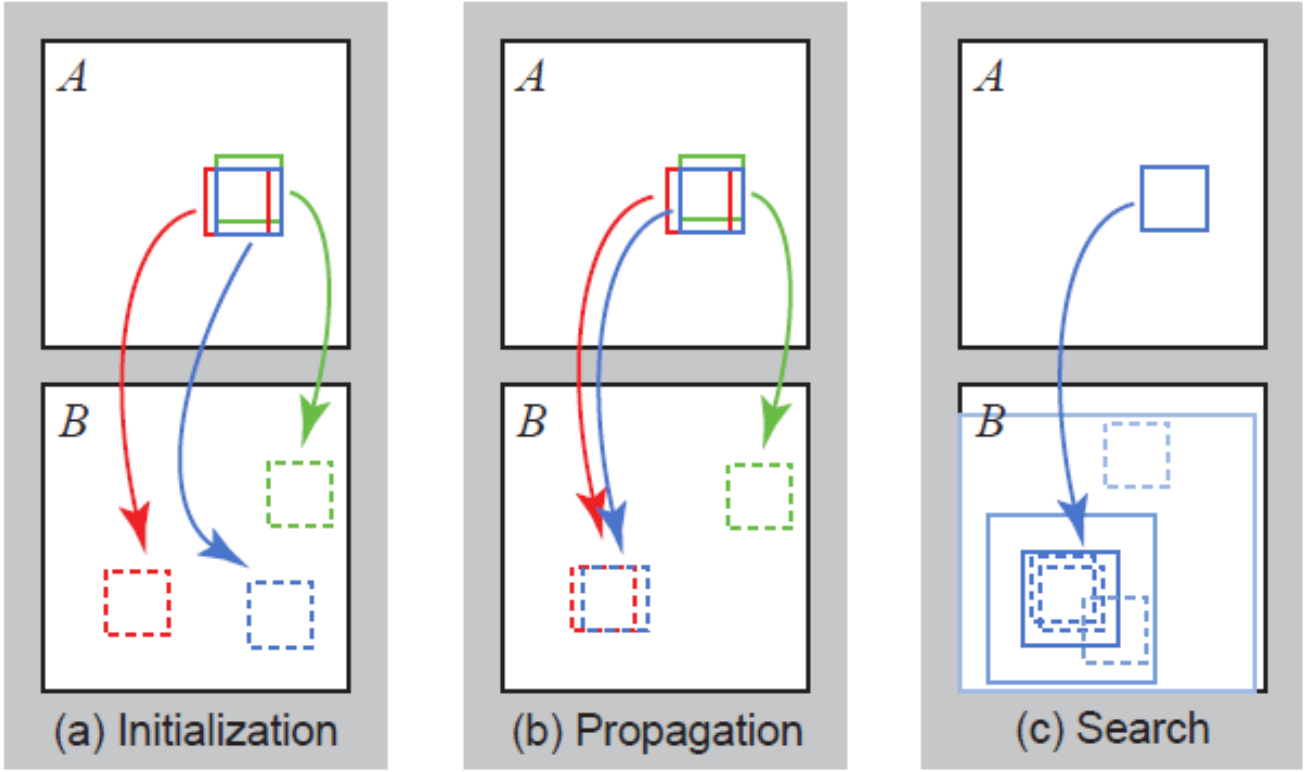
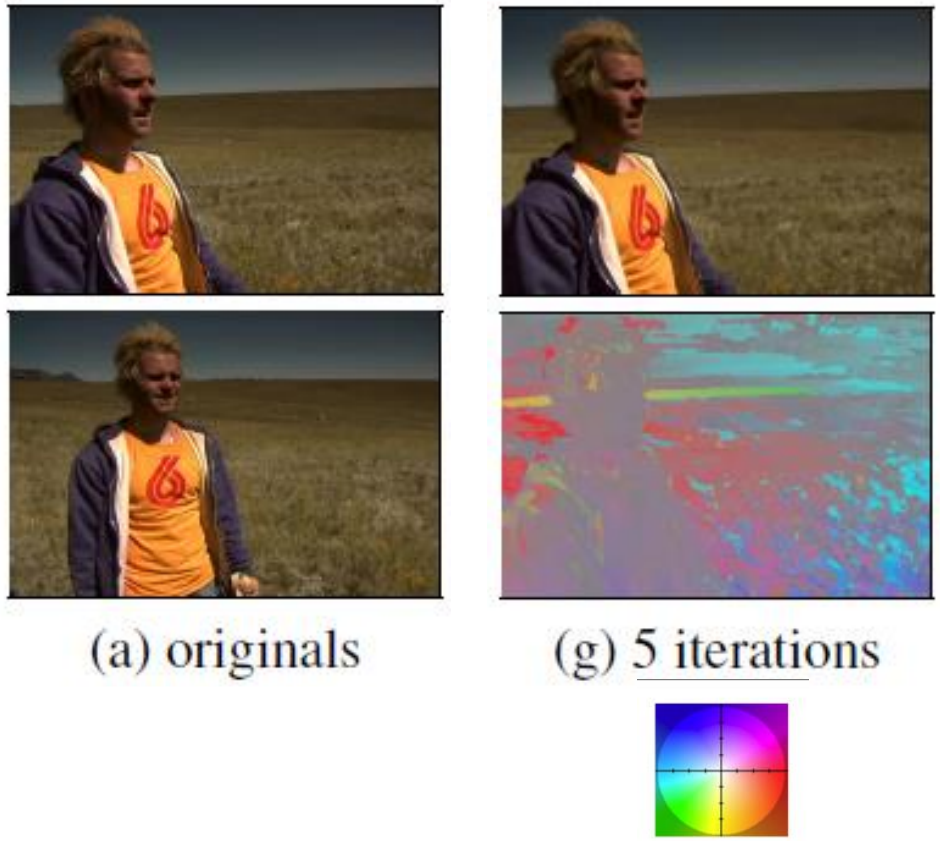


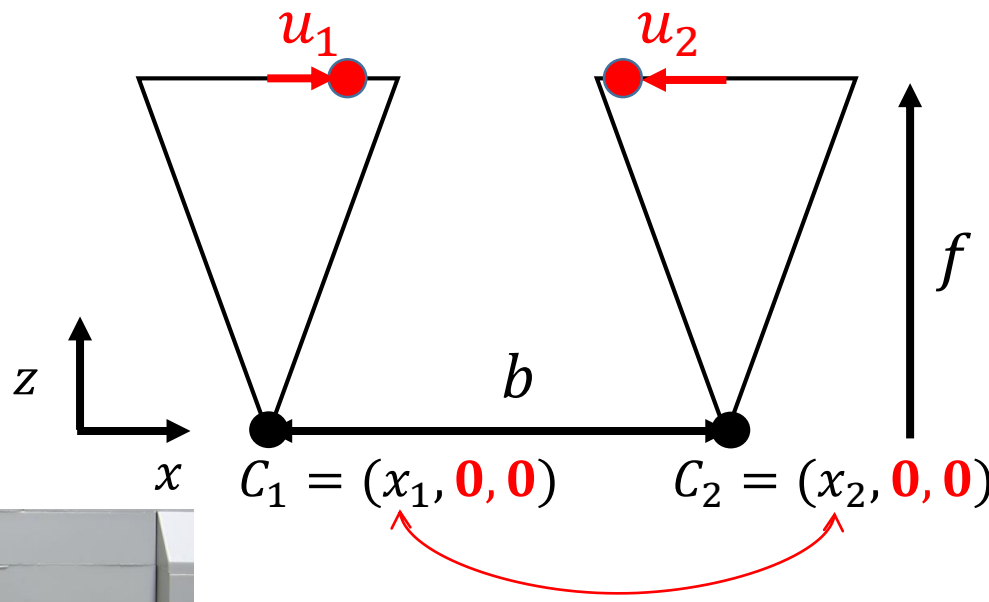
Figure 2: *Phases of the randomized nearest neighbor algorithm: (a) patches initially have random assignments; (b) the blue patch checks above/green and left/red neighbors to see if they will improve the blue mapping, propagating good matches; (c) the patch searches randomly for improvements in concentric neighborhoods.*

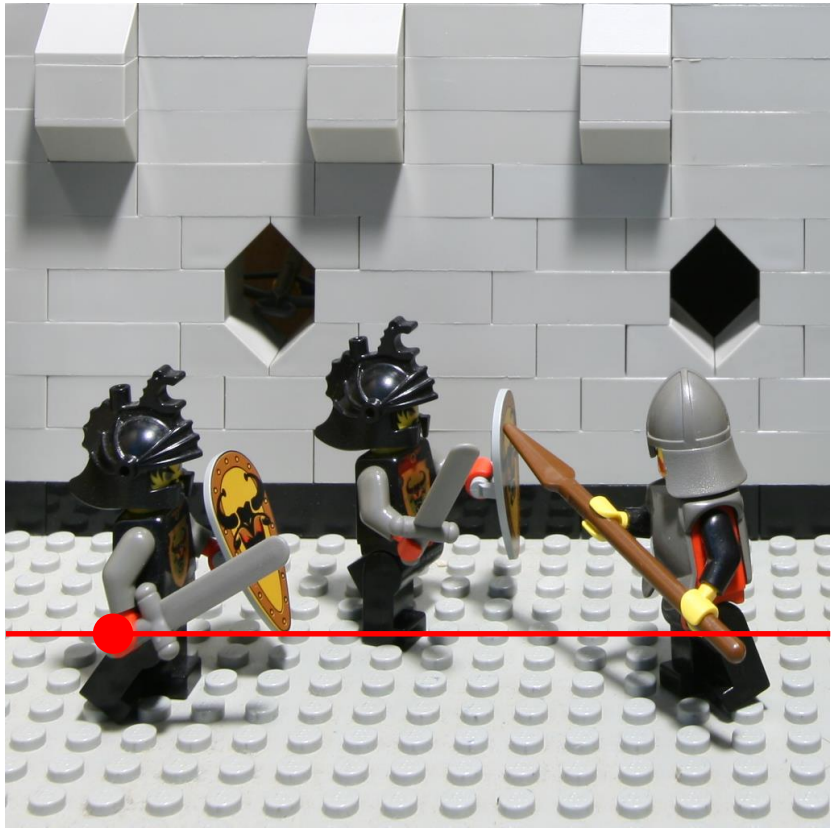


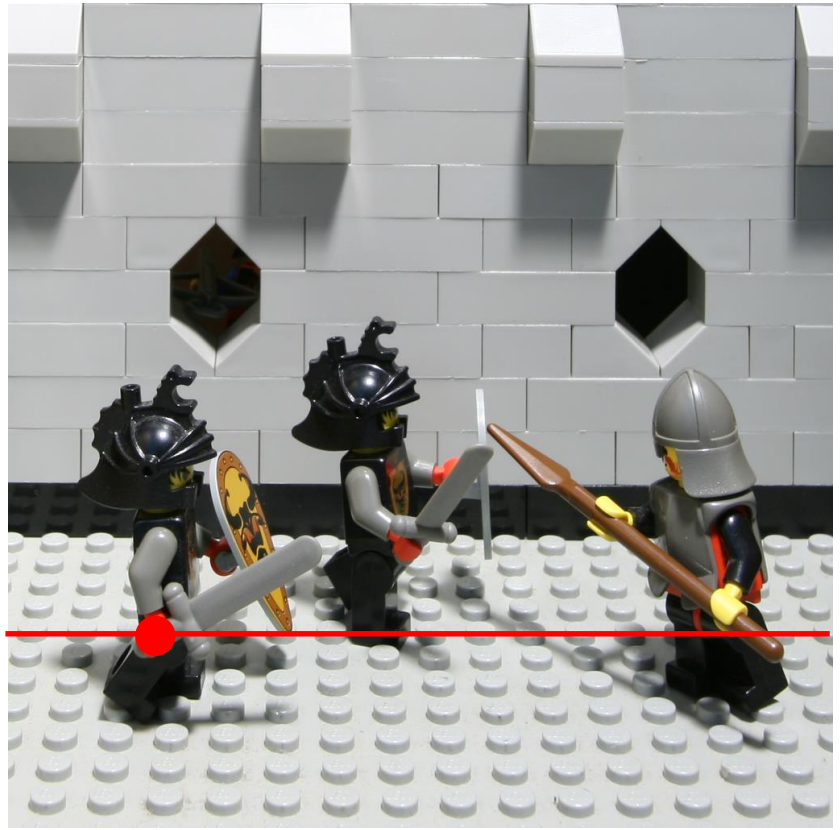
Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman.
"PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing."
ACM Transactions on Graphics (Proc. SIGGRAPH) 28(3), August 2009.

Stereo disparity estimation

- Epipolar constraint



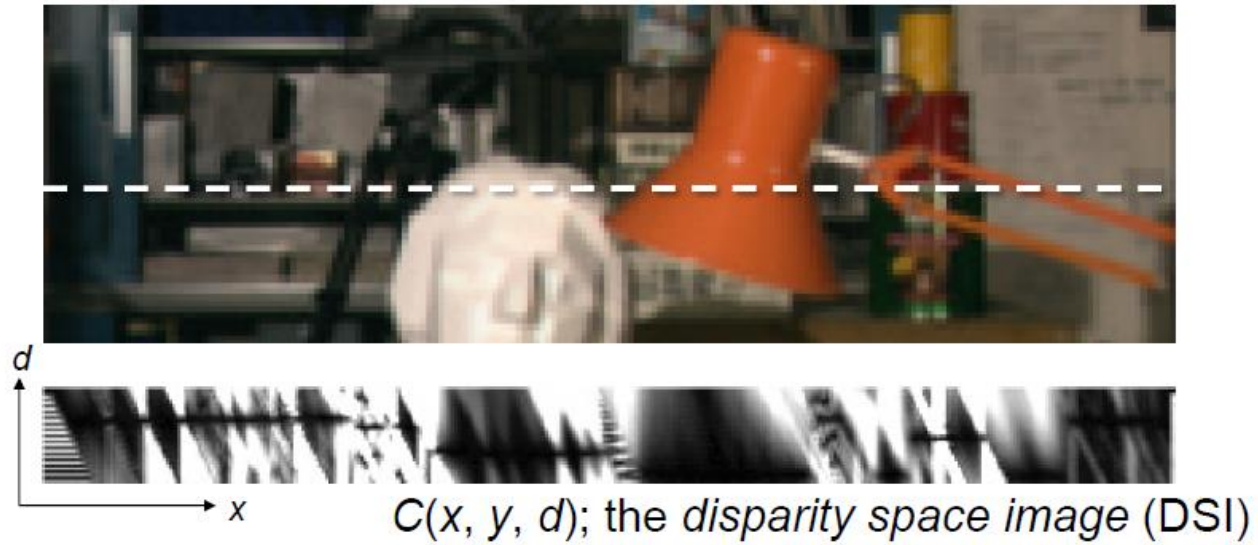




$$Disparity\ estimation\ is\ a\ 1D\ problem$$
$$\neq$$
$$2D\ Optical\ flow$$

Cost volume minimization

$I(x, y)$ $J(x, y)$



$C(x, y, d)$; the *disparity space image* (DSI)

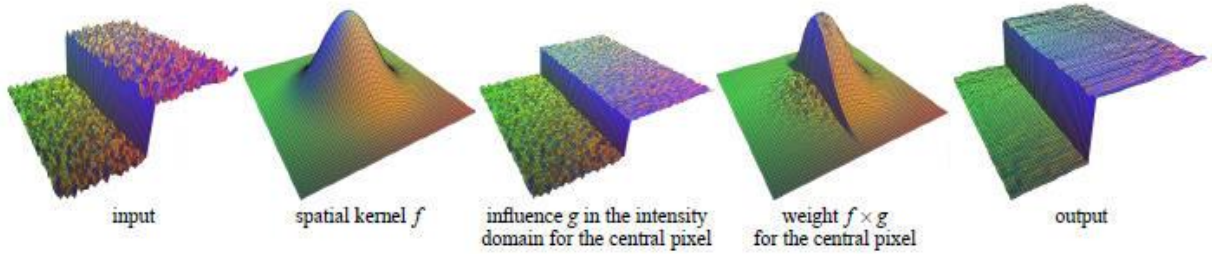
The disparity is then computed by

$$d(x, y) = \arg\min_{d'} C(x, y, d')$$

Source: [Raquel Urtasun (lecture notes),](#) N. Snavely

Cost volume filtering

- Edge-aware filter



C. Tomasi and R. Manduchi, "Bilateral Filtering for Gray and Color Images", *Proceedings of the 1998 IEEE International Conference on Computer Vision*, Bombay, India.

C. Rhemann, A. Hosni, M. Bleyer, C. Rother and M. Gelautz, "Fast cost-volume filtering for visual correspondence and beyond," *CVPR 2011*, 2011, pp. 3017-3024, doi: 10.1109/CVPR.2011.5995372.

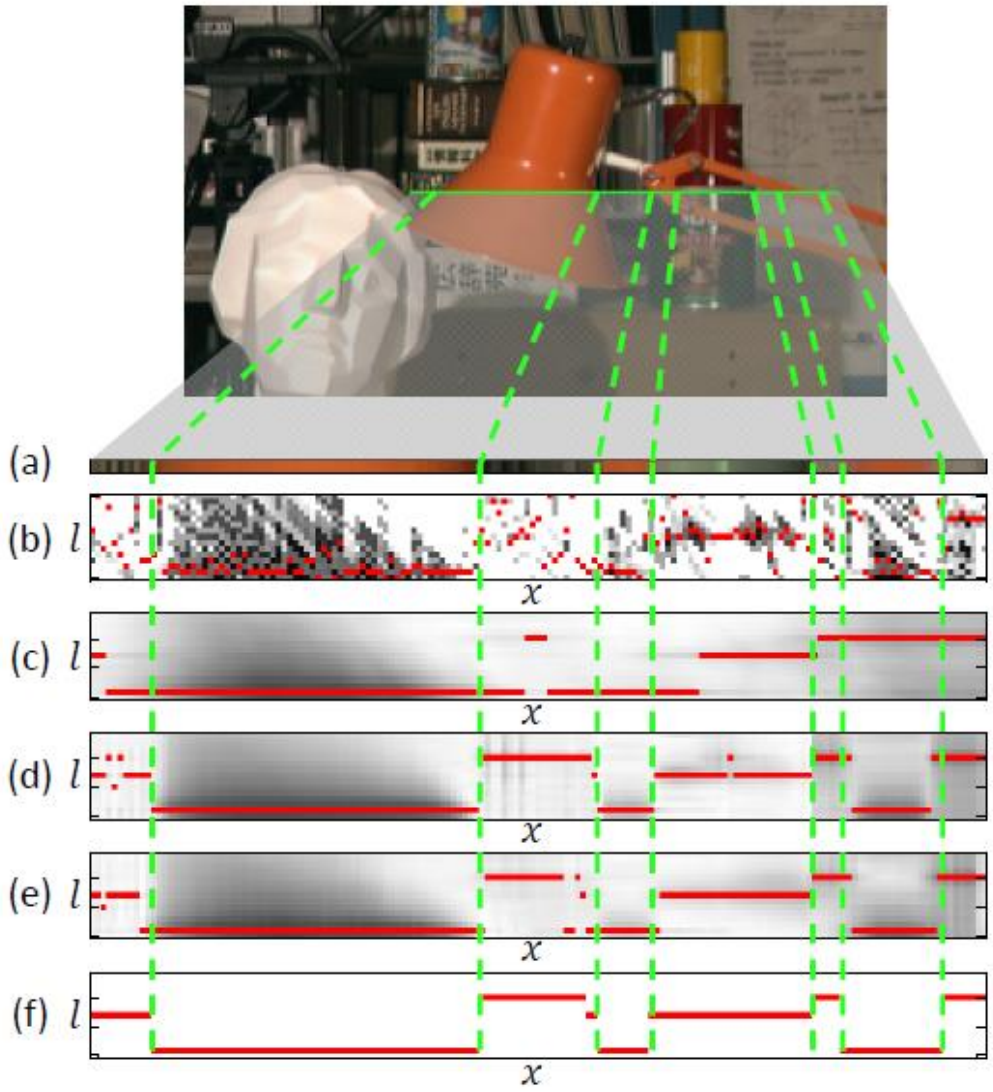


Figure 1. **Cost volume filtering.** (a) Zoom of the green line in the input image. (b) Slice of cost volume (white/black/red: high/low/lowest costs) for line in (a). (c-e) Cost slice smoothed along x and y-axes (y is not shown here) with box filter, bilateral filter and guided filter [11], respectively. (f) Ground truth labeling.

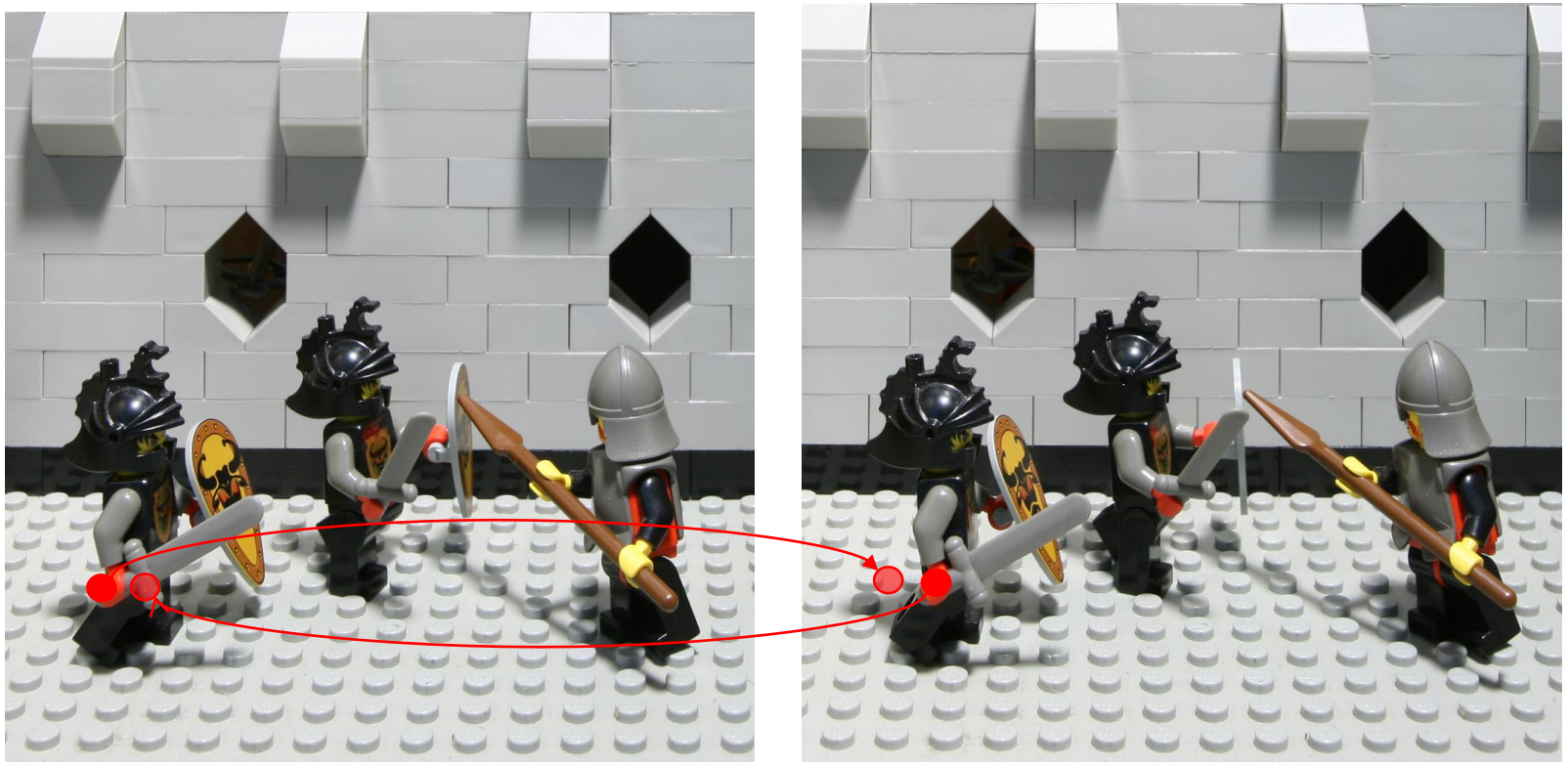
Cost volume minimization

- Global optimization techniques
  - Markov Random Fields
  - Graph cut

$$E(D) = E_{data}(D) + \lambda E_{smooth}(D)$$

Disparity refinement

- Consistency check
- Hole filling
- Sub pixel estimation



Left image

Right image

Rectification

transforming a stereo pair taken under general conditions into the ideal configuration

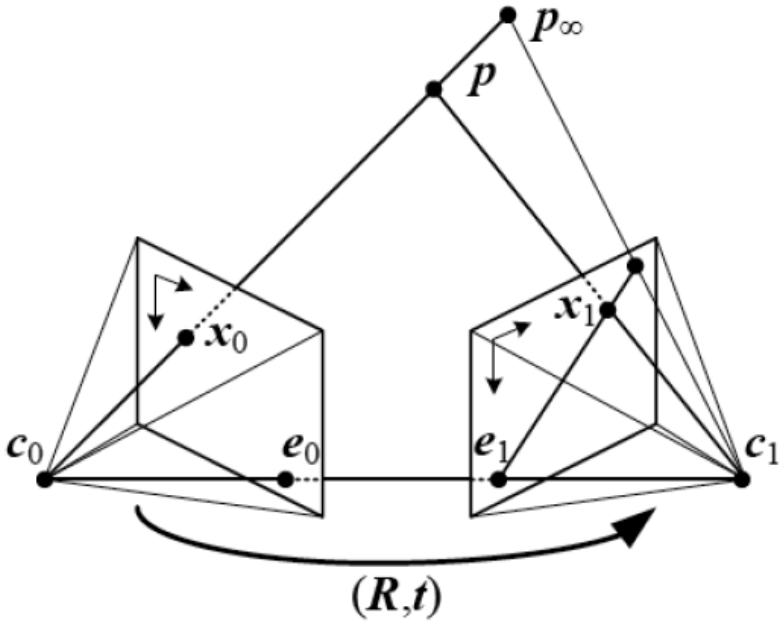
Involves a rotation of one image so that the optical axes of the two image coordinate systems are parallel

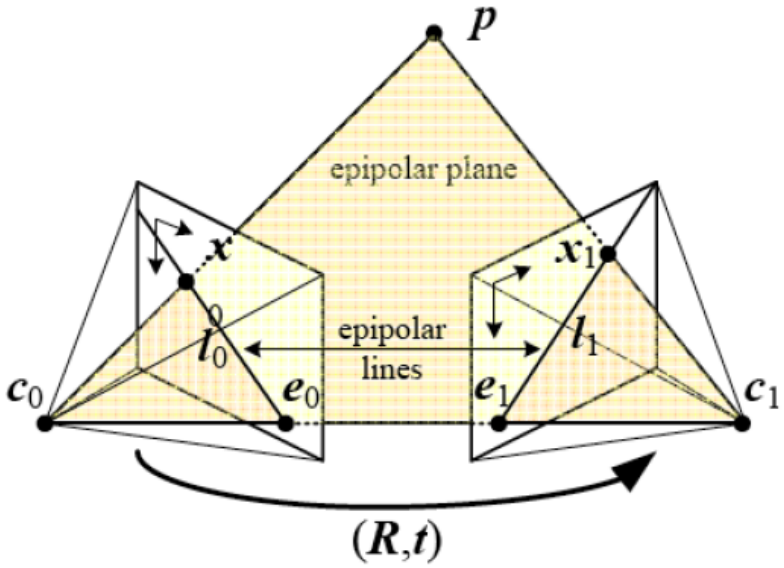
Simplifies computational structure of stereo matching algorithm

But requires interpolation to create rotated image and can create a large rectified image if the rotation angles are large.

Source: Raquel Urtasun (lecture notes), Ramani

Generalisation

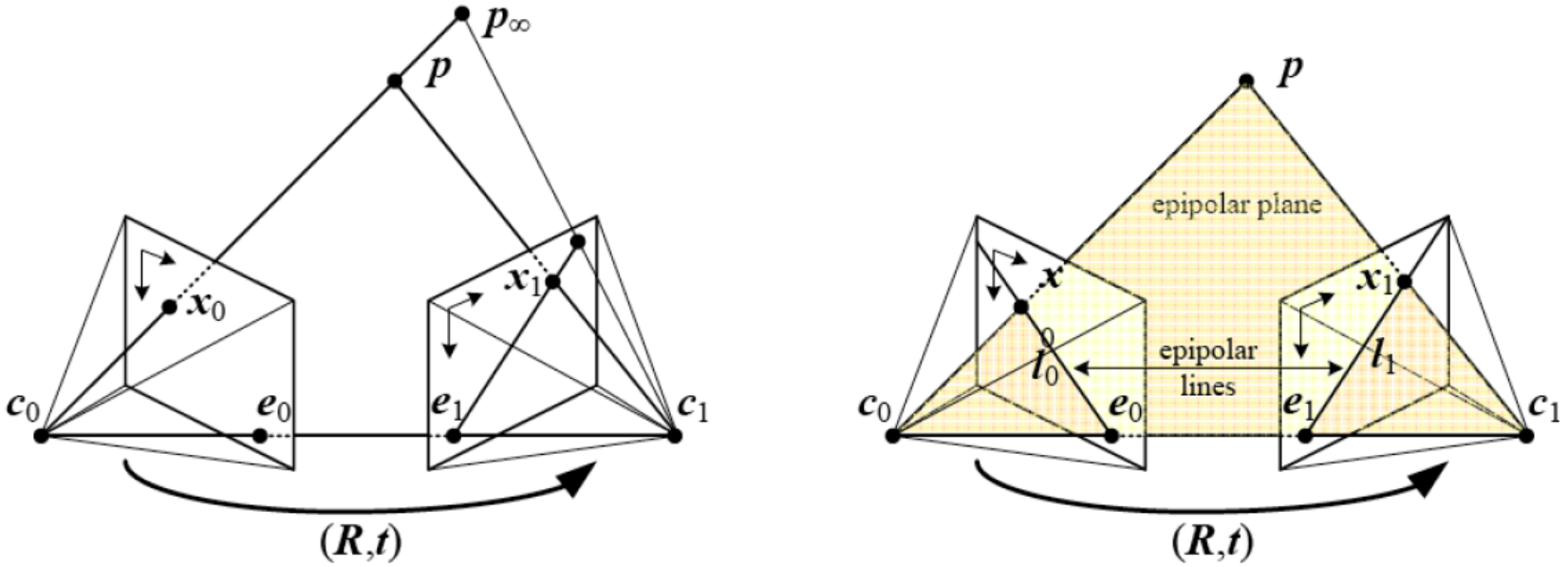




- Pixel in one image $\mathbf{x}_0$ projects to an **epipolar line** segment in the other image
- The segment is bounded at one end by the projection of the original viewing ray at infinity $\mathbf{p}_\infty$ and at the other end by the projection of the original camera center $\mathbf{c}_0$ into the second camera, which is known as the **epipole** $\mathbf{e}_1$.

Source: [Raquel Urtasun (lecture notes),]

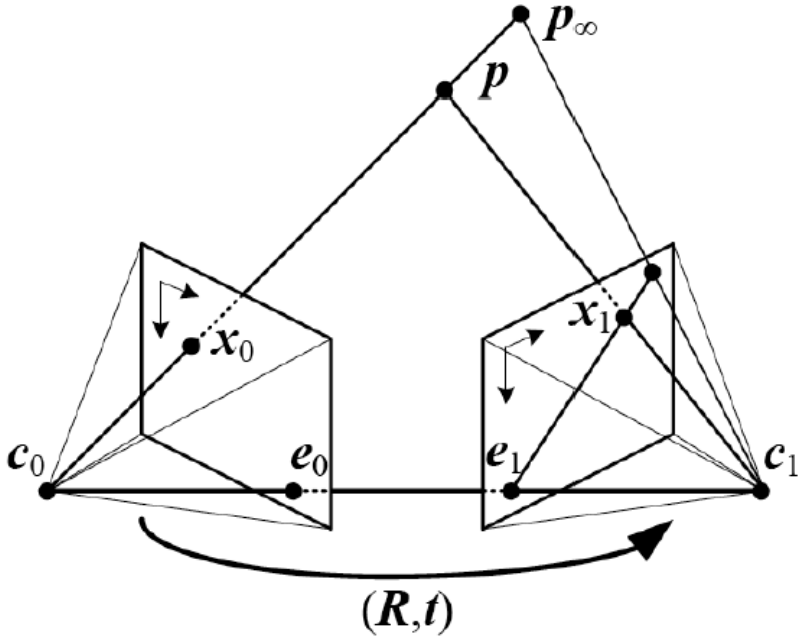
Generalisation



- If we project the epipolar line in the second image back into the first, we get another line (segment), this time bounded by the other corresponding **epipole** e_0
- Extending both line segments to infinity, we get a pair of corresponding epipolar lines, which are the intersection of the two image planes with the **epipolar plane** that passes through both camera centers $\mathbf{c}_0$ and $\mathbf{c}_1$ as well as the point of interest **p**

Source: Raquel Urtasun (lecture notes),

Generalisation

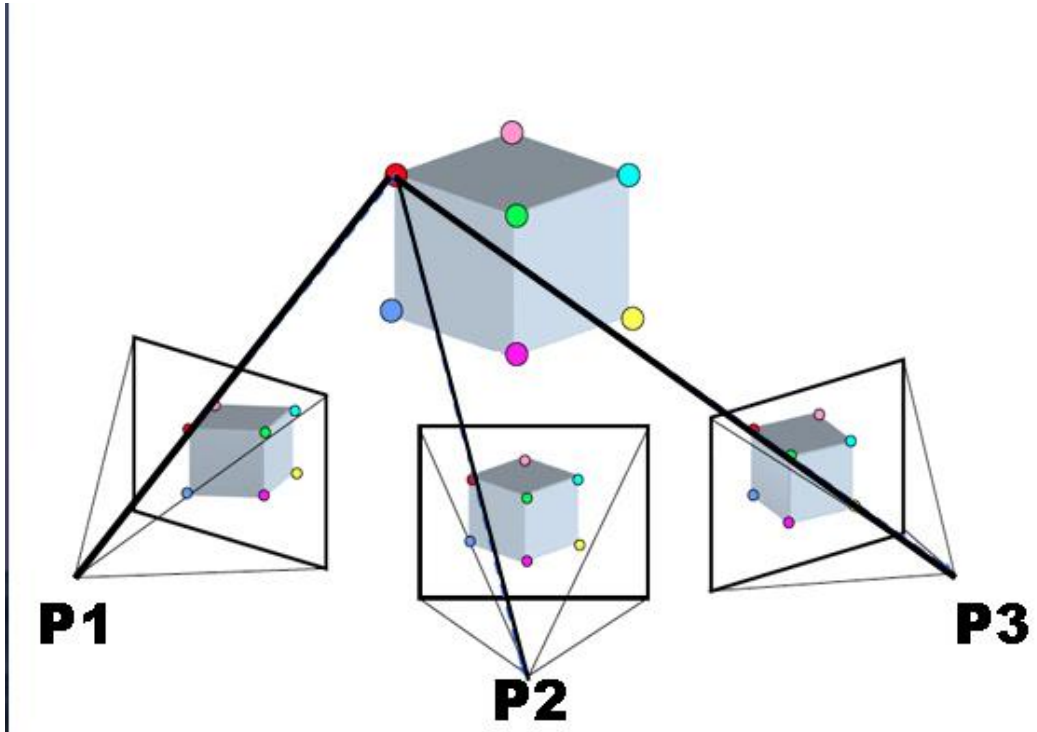


- Projective geometry depends only on the cameras internal parameters and relative pose of cameras (and not the 3D scene)
- Fundamental matrix **F** encapsulates this geometry $\mathbf{x}_0^T \mathbf{F} \mathbf{x}_1 = 0$

Source: [Raquel Urtasun (lecture notes),]

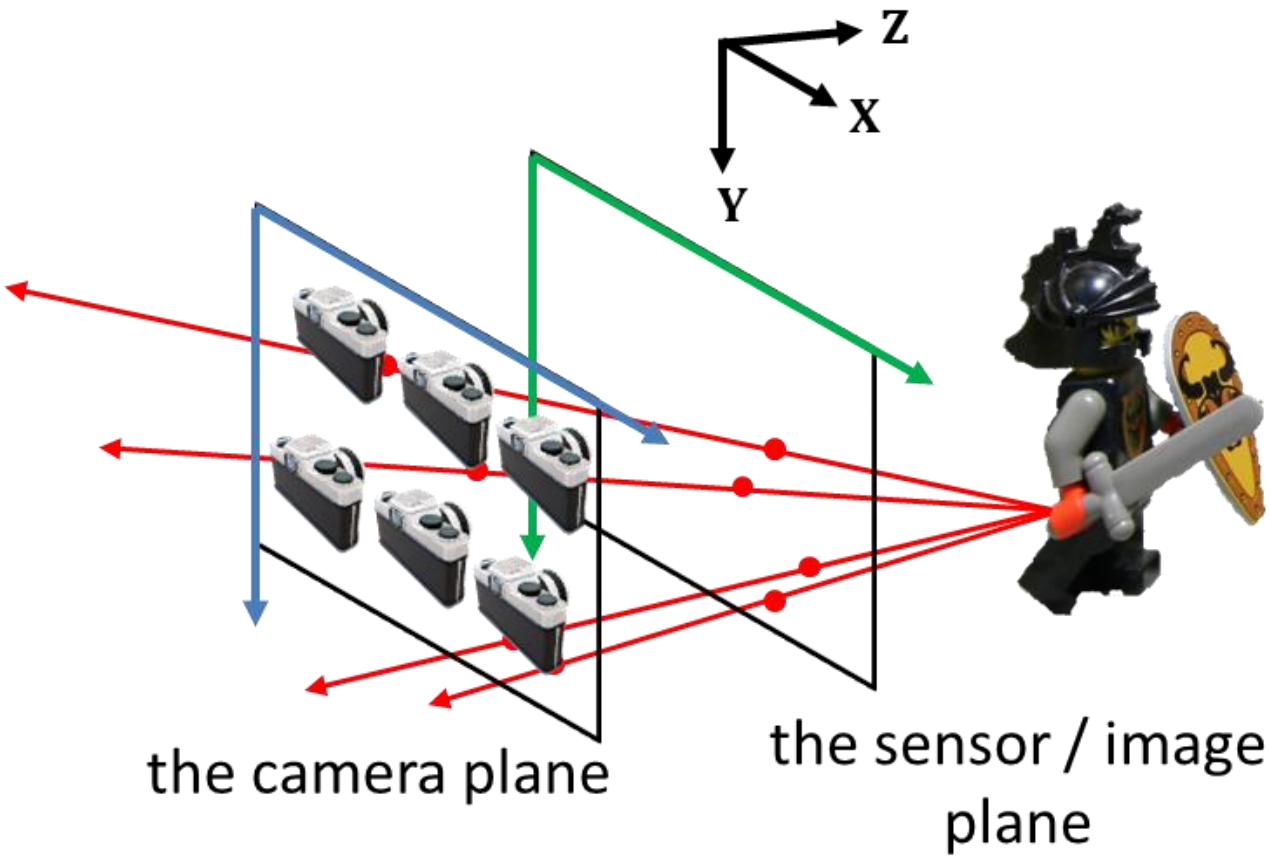
Generalisation - Multiview

- Structure from Motion (SfM
- Simultaneous localization and mapping (SLAM)

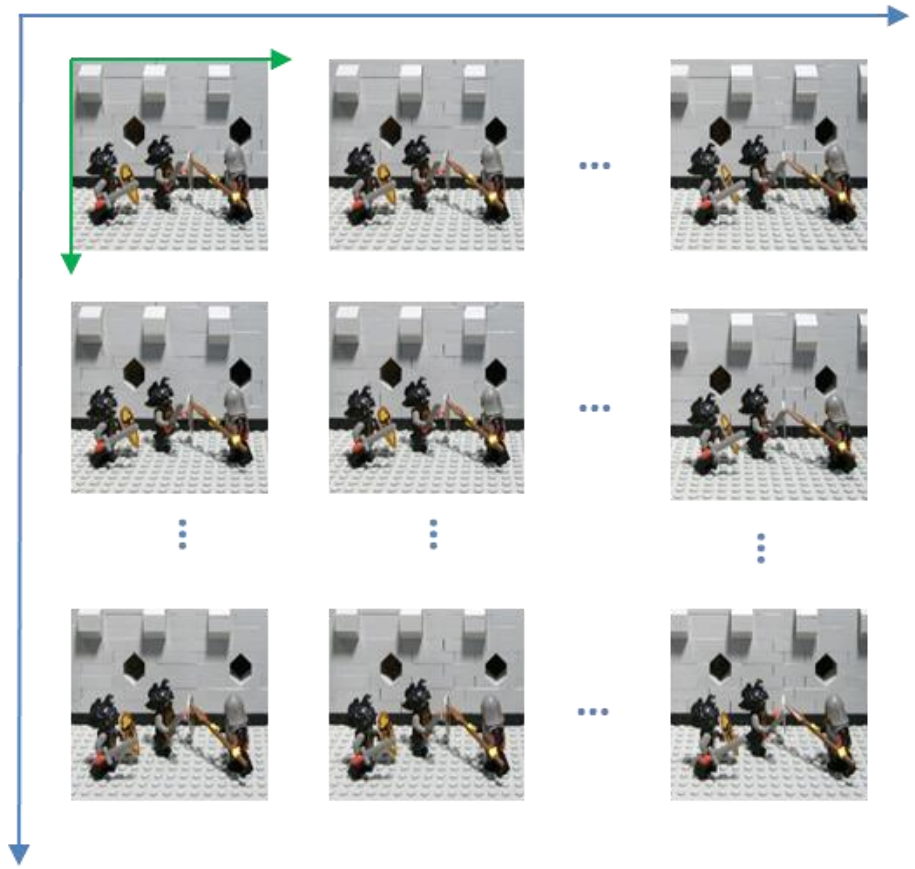


https://en.wikipedia.org/wiki/Structure_from_motion

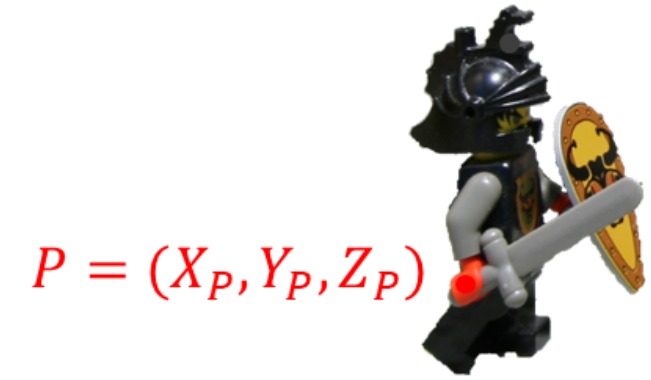
Light fields



$L(s, t, u, v)$

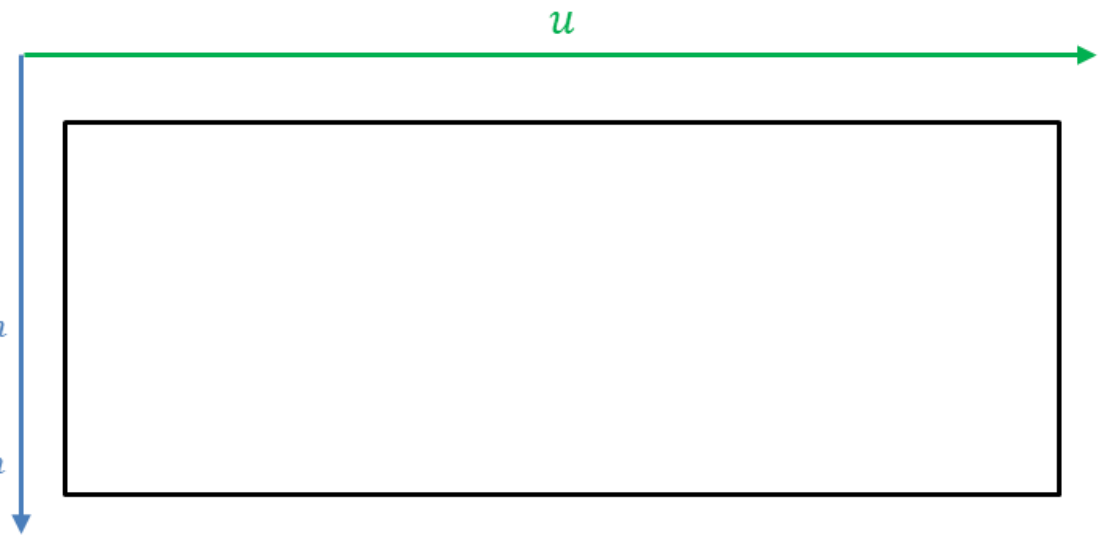


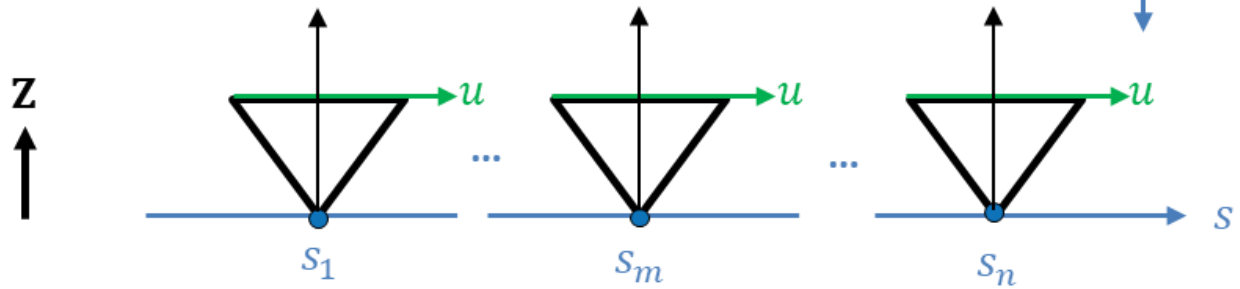
Depth estimation from light fields



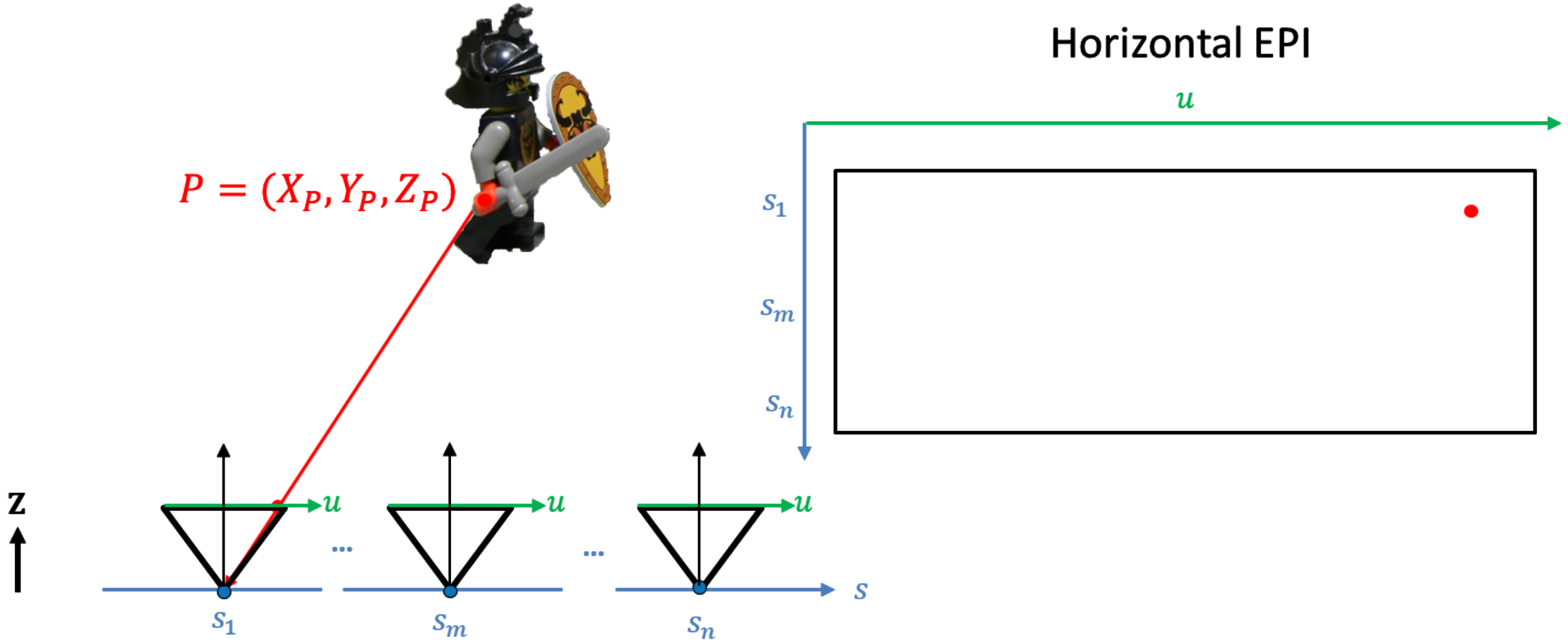
$P = (X_P, Y_P, Z_P)$

Horizontal EPI

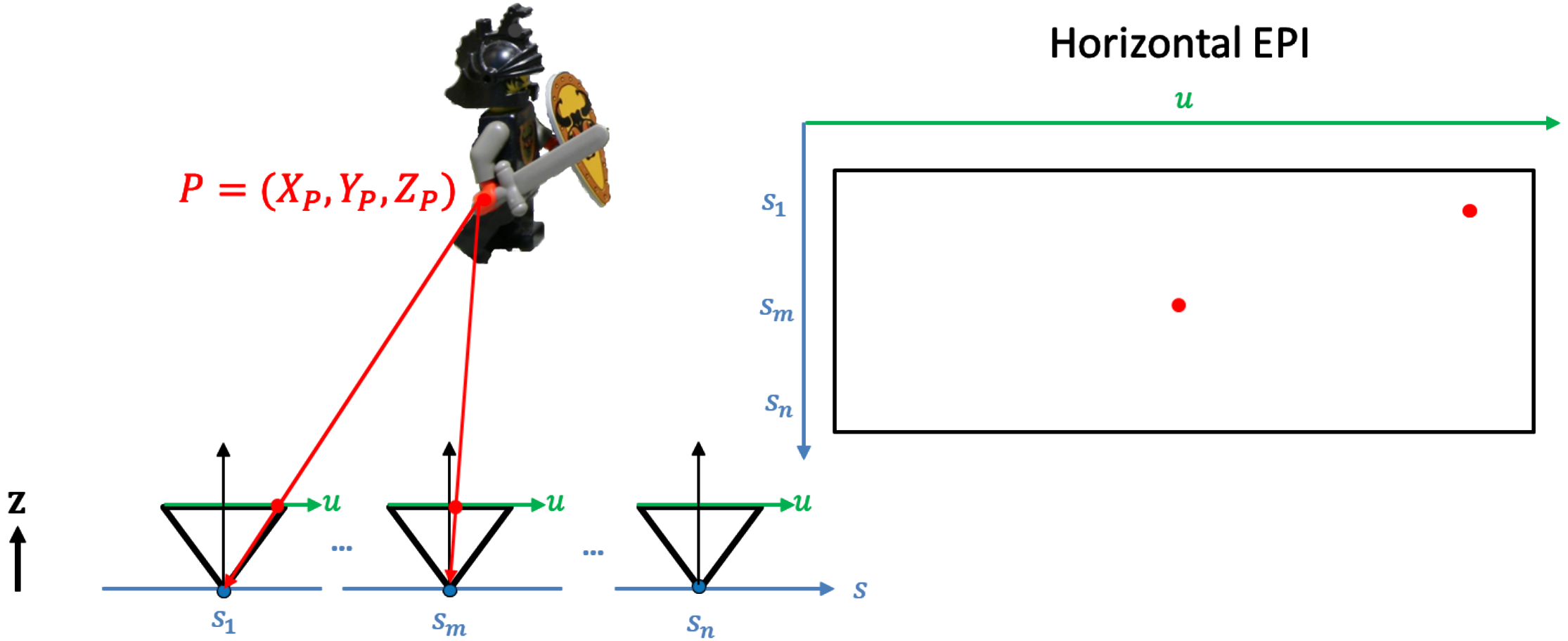




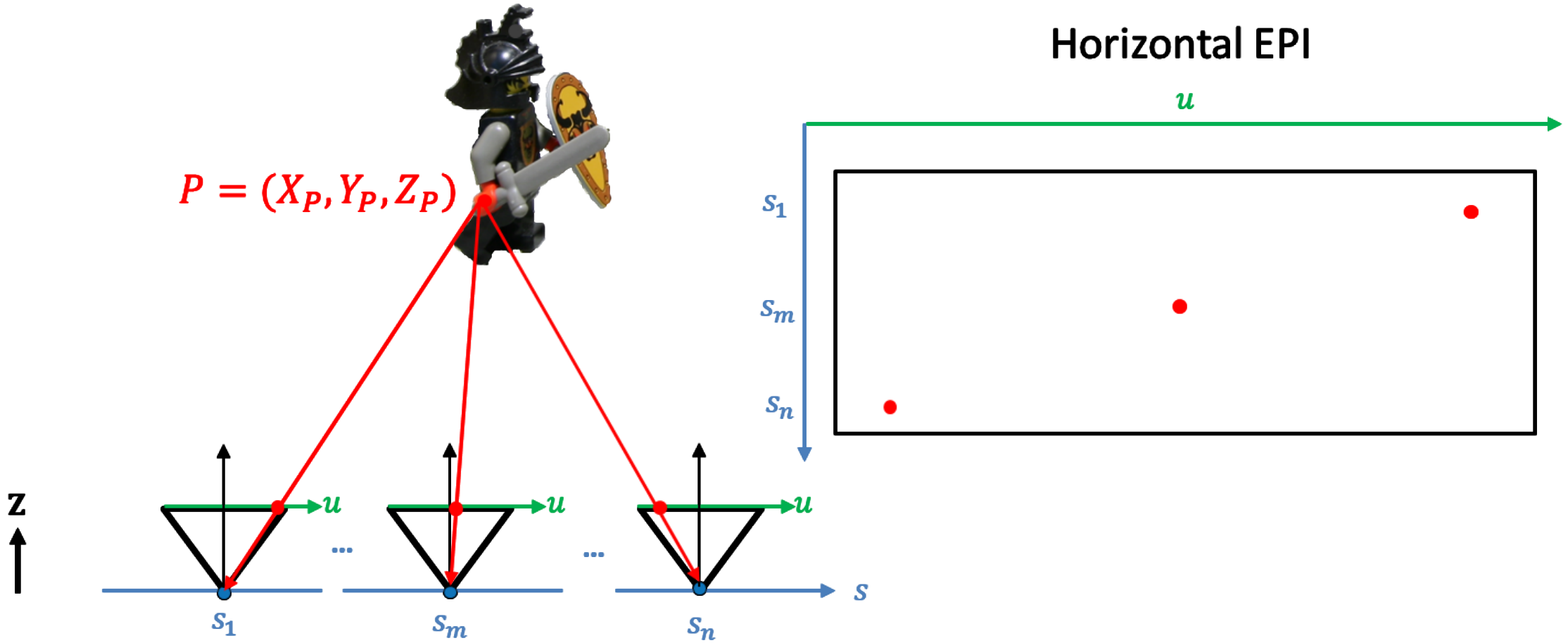
Depth estimation from light fields



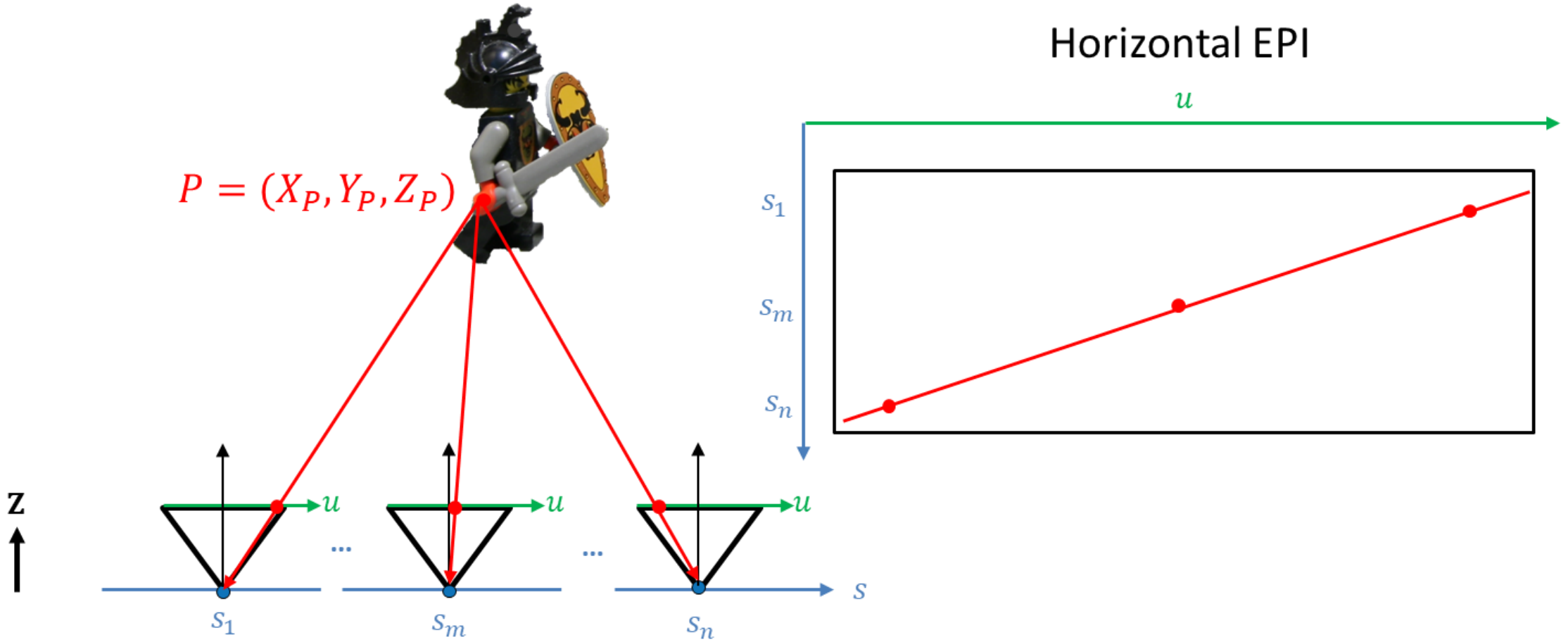
Depth estimation from light fields



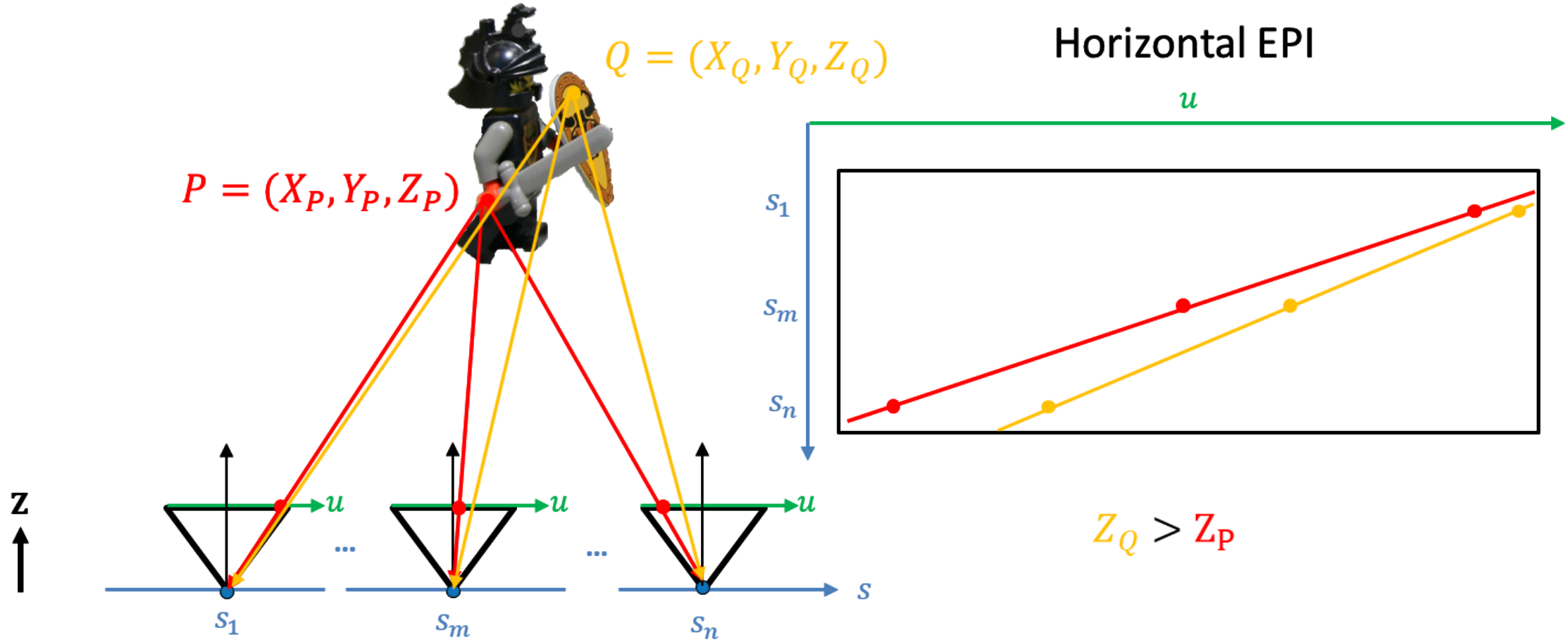
Depth estimation from light fields



Depth estimation from light fields

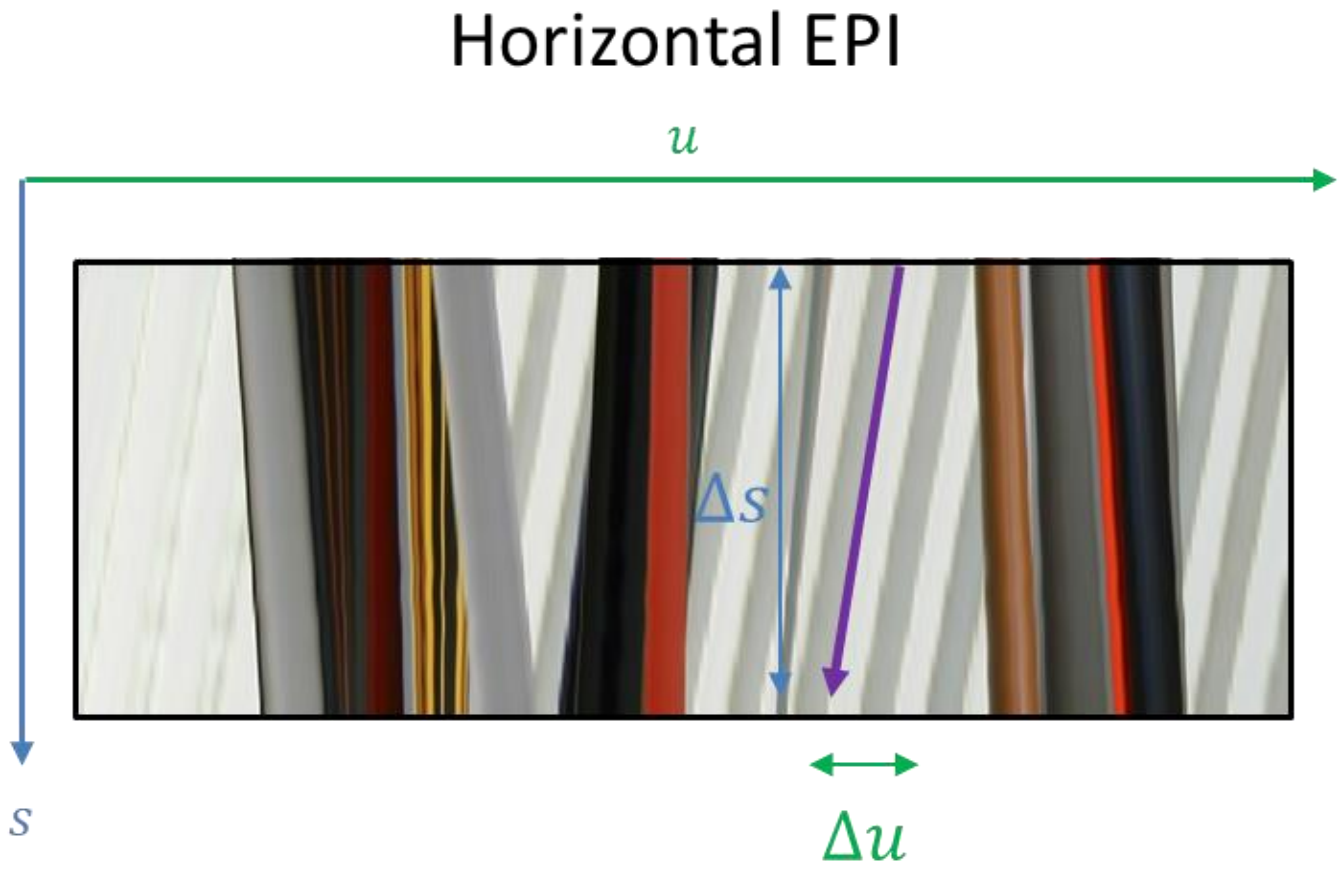


Depth estimation from light fields



Depth estimation from light fields

$$Z \sim \frac{\Delta s}{\Delta u}$$



Shape from X

- Shape from shading
- Shape from texture
- Shape from defocus

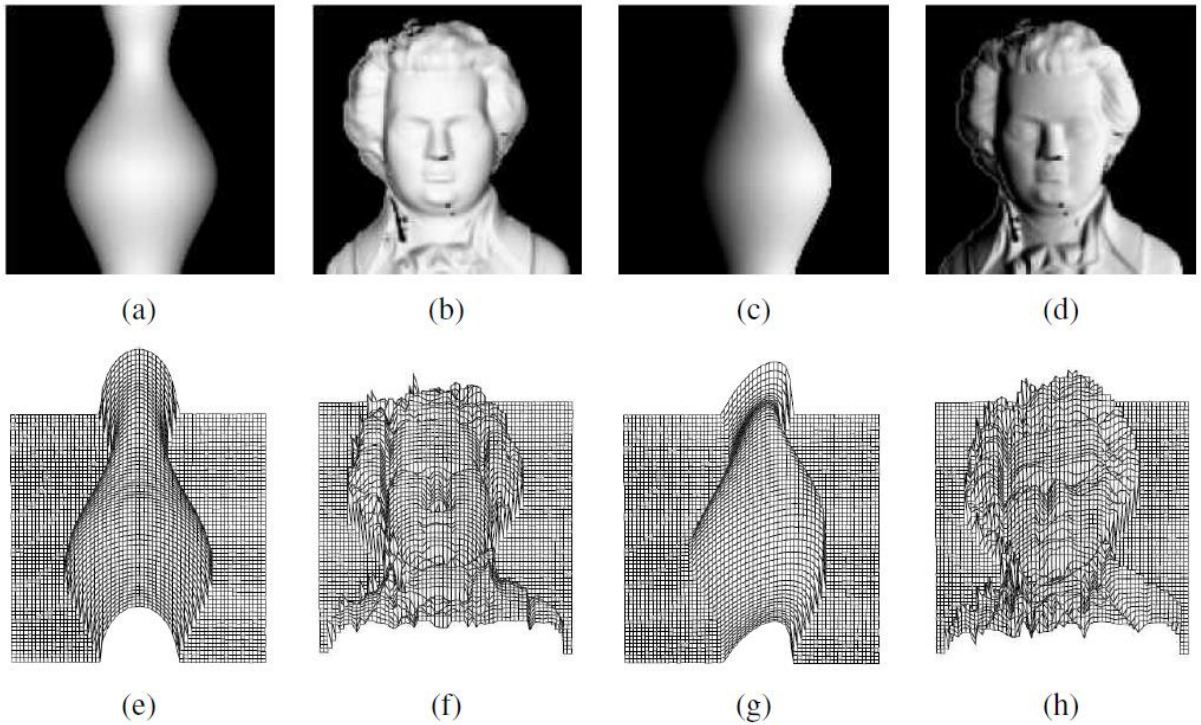


Figure 13.2 *Synthetic shape from shading (Zhang, Tsai* et al. *1999) © 1999 IEEE: shaded images, (a–b) with light from in front* $(0, 0, 1)$ *and (c–d) with light from the front right* $(1, 0, 1)$*; (e–f) corresponding shape from shading reconstructions using the technique of Tsai and Shah (1994).*

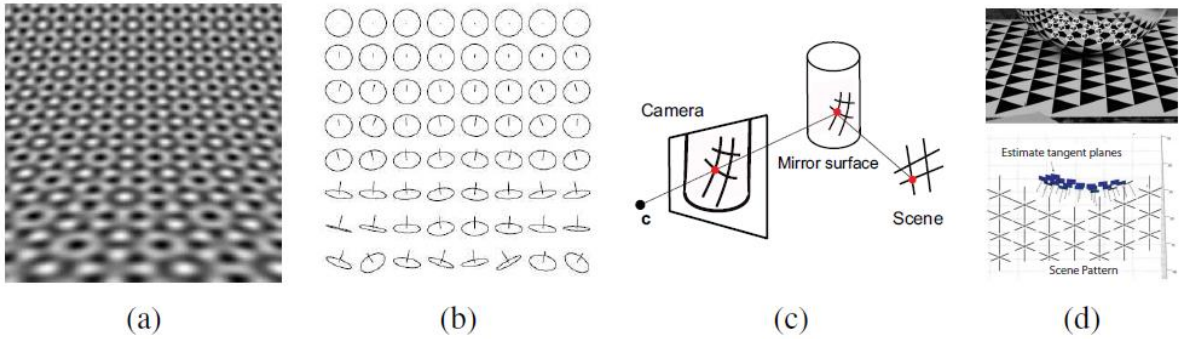
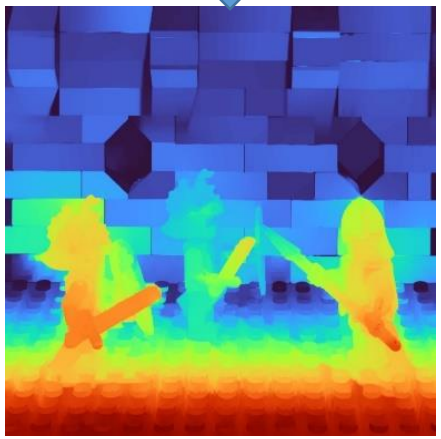


Figure 13.5 *Synthetic shape from texture (Gårding 1992) © 1992 Springer: (a) regular texture wrapped onto a curved surface and (b) the corresponding surface normal estimates. Shape from mirror reflections (Savarese, Chen, and Perona 2005) © 2005 Springer: (c) a regular pattern reflecting off a curved mirror gives rise to (d) curved lines, from which 3D point locations and normals can be inferred.*

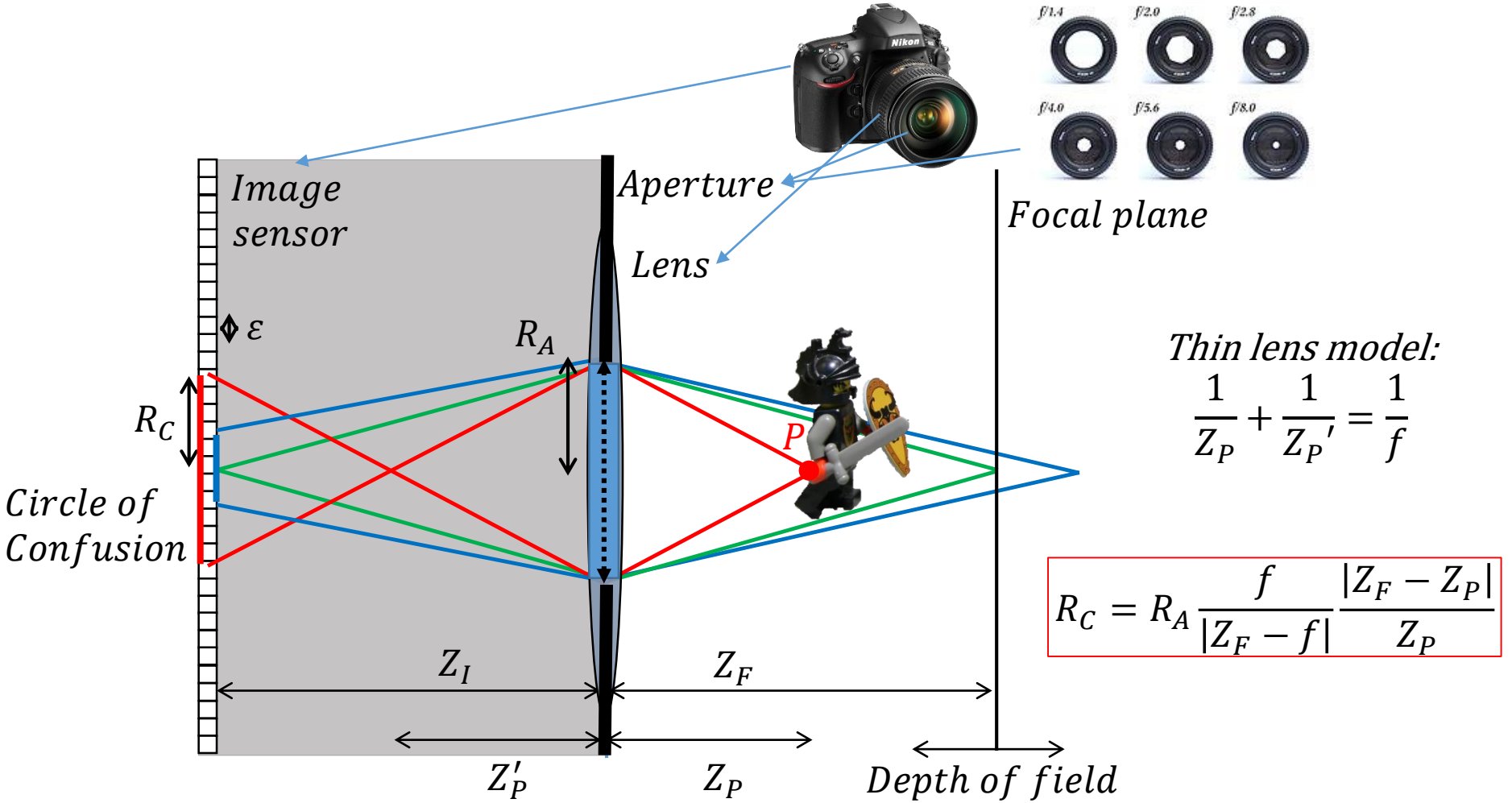
Source: https://szeliski.org/Book

Shape from defocus

Input image



Focus/defocus map



Thin lens model:

$$\frac{1}{Z_P} + \frac{1}{Z_P'} = \frac{1}{f}$$

$$R_C = R_A \frac{f}{|Z_F - f|} \frac{|Z_F - Z_P|}{Z_P}$$

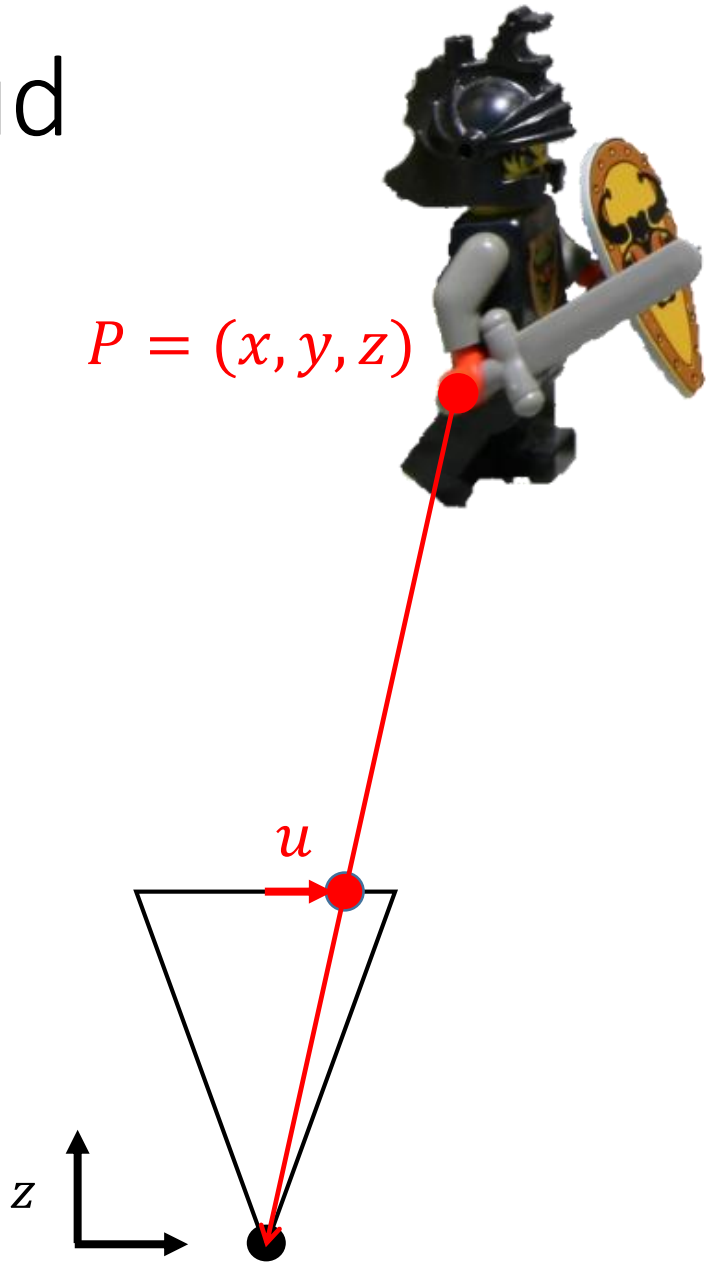
Application – Point cloud

Projection from scene to camera:

$$K[R \mid t]\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \Rightarrow \begin{bmatrix} u/w \\ v/w \\ 1 \end{bmatrix}$$

Projection from camera to scene:

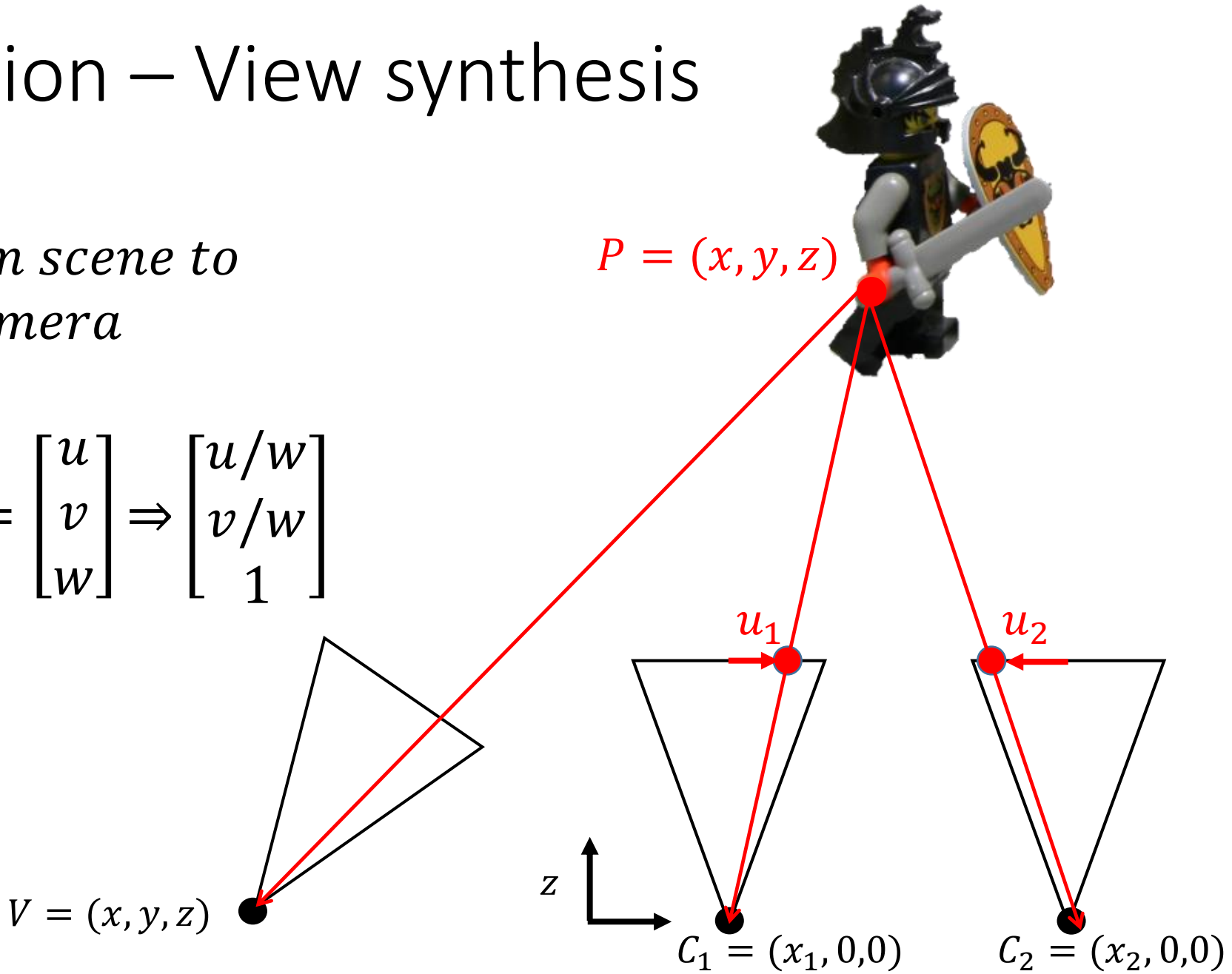
$$(K[R \mid t])^{-1}\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} x/z \\ y/z \\ 1 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$



Application – View synthesis

Projection from scene to virtual camera

$$K_V \begin{bmatrix} R_V & | t_V \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \Rightarrow \begin{bmatrix} u/w \\ v/w \\ 1 \end{bmatrix}$$



Summary

- Depth estimation is a fundamental task of computer vision
- Depth can be estimated from disparity using stereo pair of image
- May other method exist denoted shape-from-X
- Recent state-of-the-art is based on deep learning