UNIVERSITÀ DI PARMA

Stereo Matching

Summary

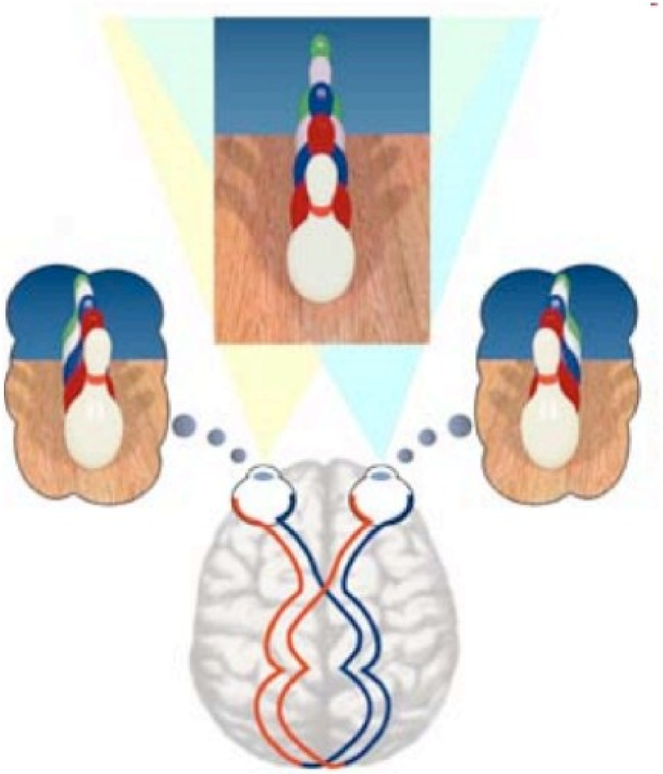
UNIVERSITÀ DI PARMA

- Small recap about:
  - Epipolar geometry
  - Stereo Images Rectification
- Disparity
- Correspondences search
  - NCC
  - SAD

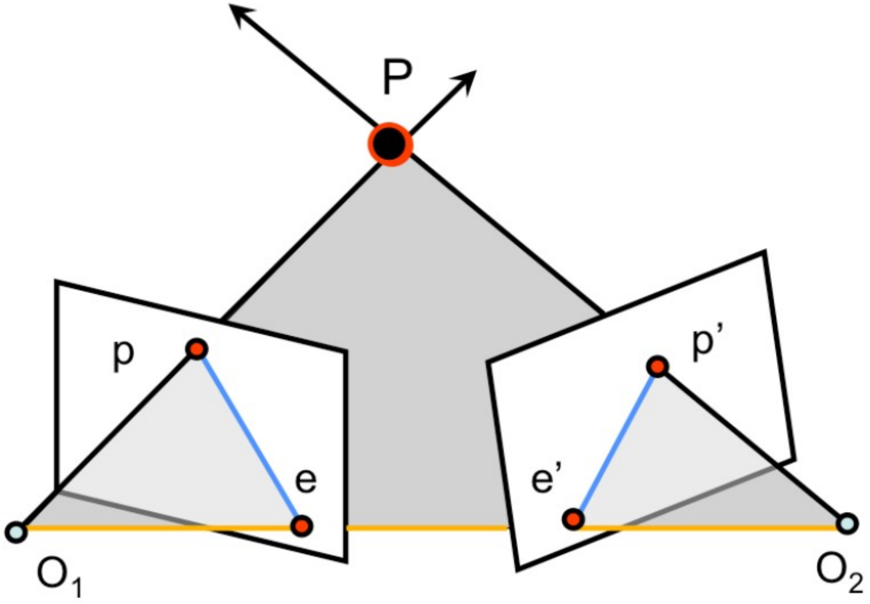
Credits

- [FP] D. A. Forsyth and J. Ponce. Computer Vision: A Modern Approach (2nd Edition). Prentice Hall, 2011.
- [HZ] R. Hartley and A. Zisserman. Multiple View Geometry in Computer Vision. Cambridge University Press, 2003.
- CS231A · Computer Vision: from 3D reconstruction to recognition
  - Prof. Silvio Savarese – Stanford University
- 15-463, 15-663, 15-862, Computational Photography, Fall 2021
  - Prof. Ioannis Gkioulekas – Carniage Mellon Graphics Lab

Stereo vision wins

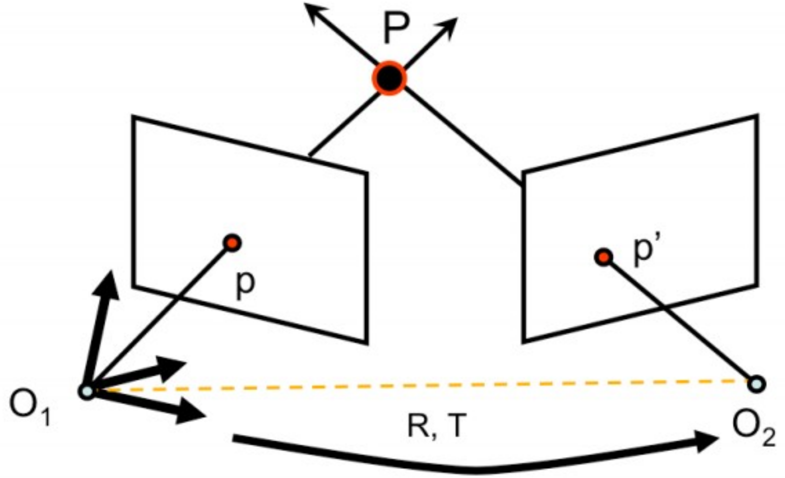


Epipolar geometry



- O1-O2-P: epipolar plane
- O1-O2: baseline
- pe & p’e’: epipolar lines (i.e. they meet!)
- e & e’: epipoles
  - Intersection of baseline with image planes
  - Projection of O1&O2

Essential Matrix



$$p^T \cdot [T \times (R\, p')] = 0 \rightarrow p^T \cdot \boxed{[T_\times] \cdot R}\, p' = 0$$

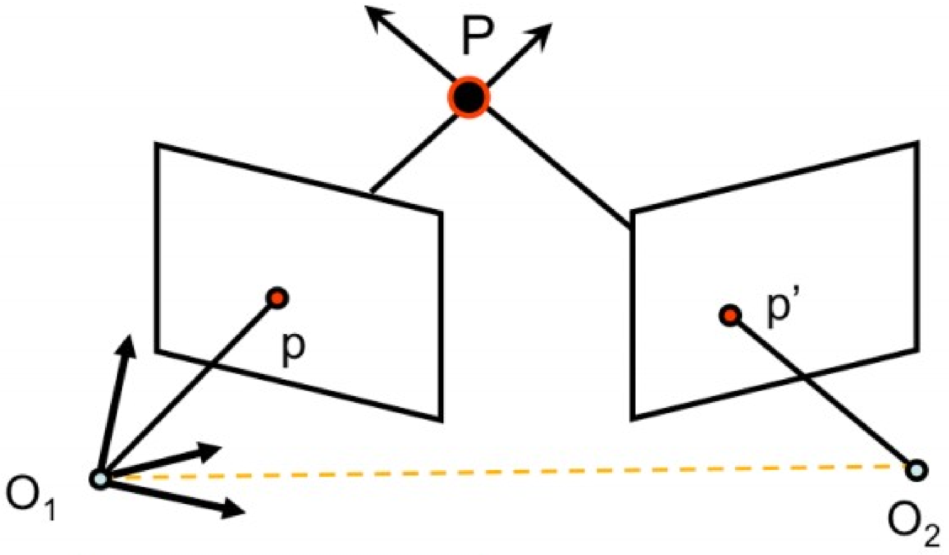
[Eq. 8] [Eq. 9]

E = Essential matrix

(Longuet-Higgins, 1981)

- E is the **Essential Matrix**

Fundamental Matrix



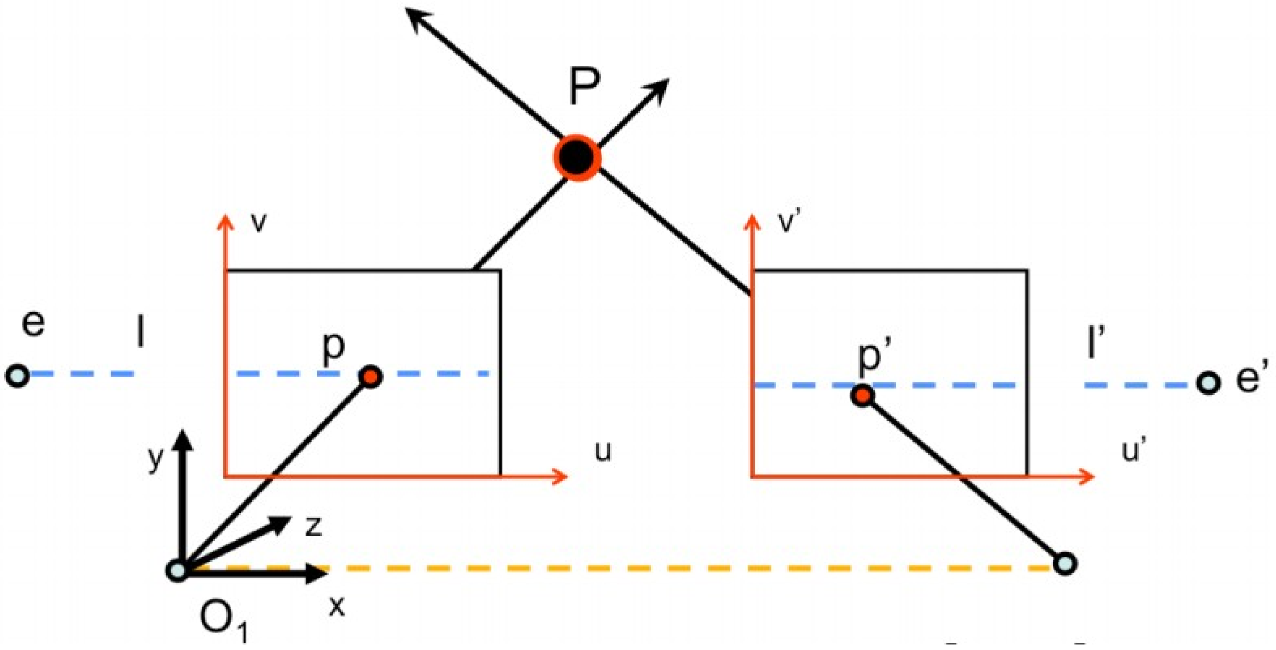
[Eq. 13] $\boxed{p^T F p' = 0}$

$$F = K^{-T} \cdot [T_\times] \cdot R K'^{-1}$$

[Eq. 14]

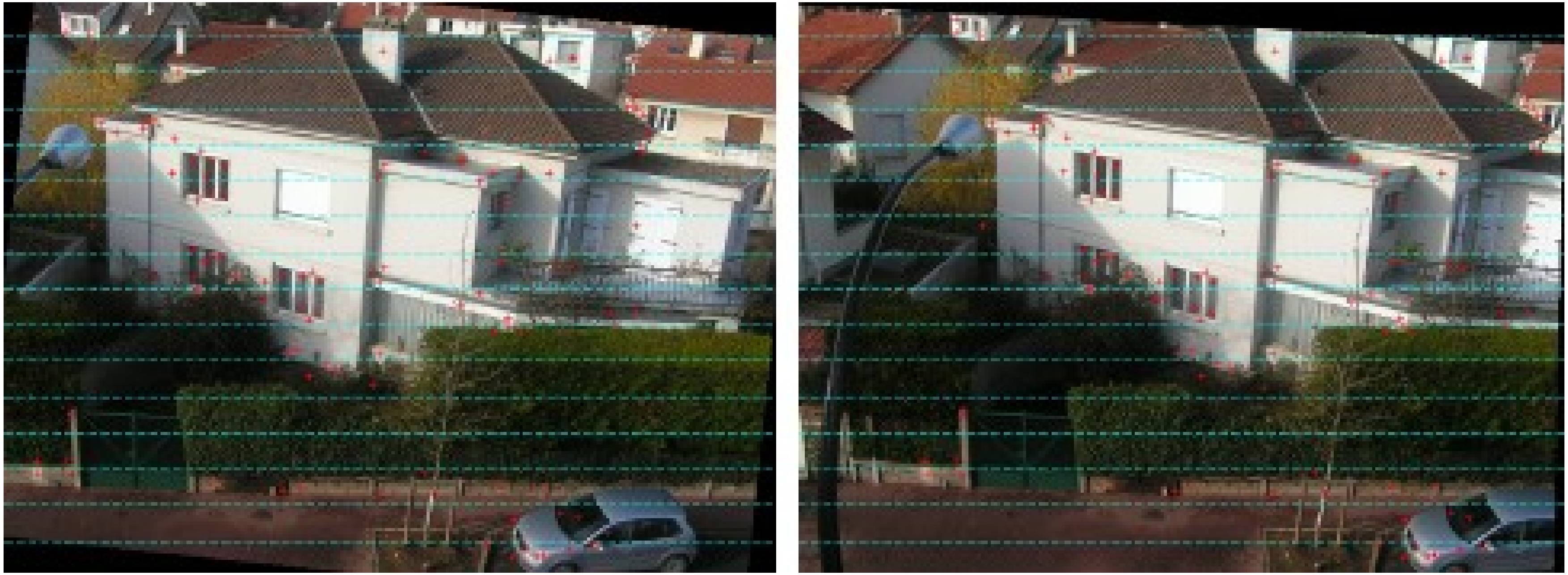
- F ⟶ Fundamental Matrix (Faugeras and Luong 1992)

Special case: parallel image planes



- v=v' means the same f**g line!
- The search for p' is simpler…

Special case: parallel image planes



- Example of two images whose planes are parallel to each other
- Epipolar lines are parallel to each other

Key problems

- **Correspondances:** given a p in one image find p’ in the other image
- ~~**Camera geometry:** given a set of correspondences find camera parameters~~
- ~~**Scene Geometry:** given a set of correspondences and parameters of both cameras reconstruct the 3D scene~~

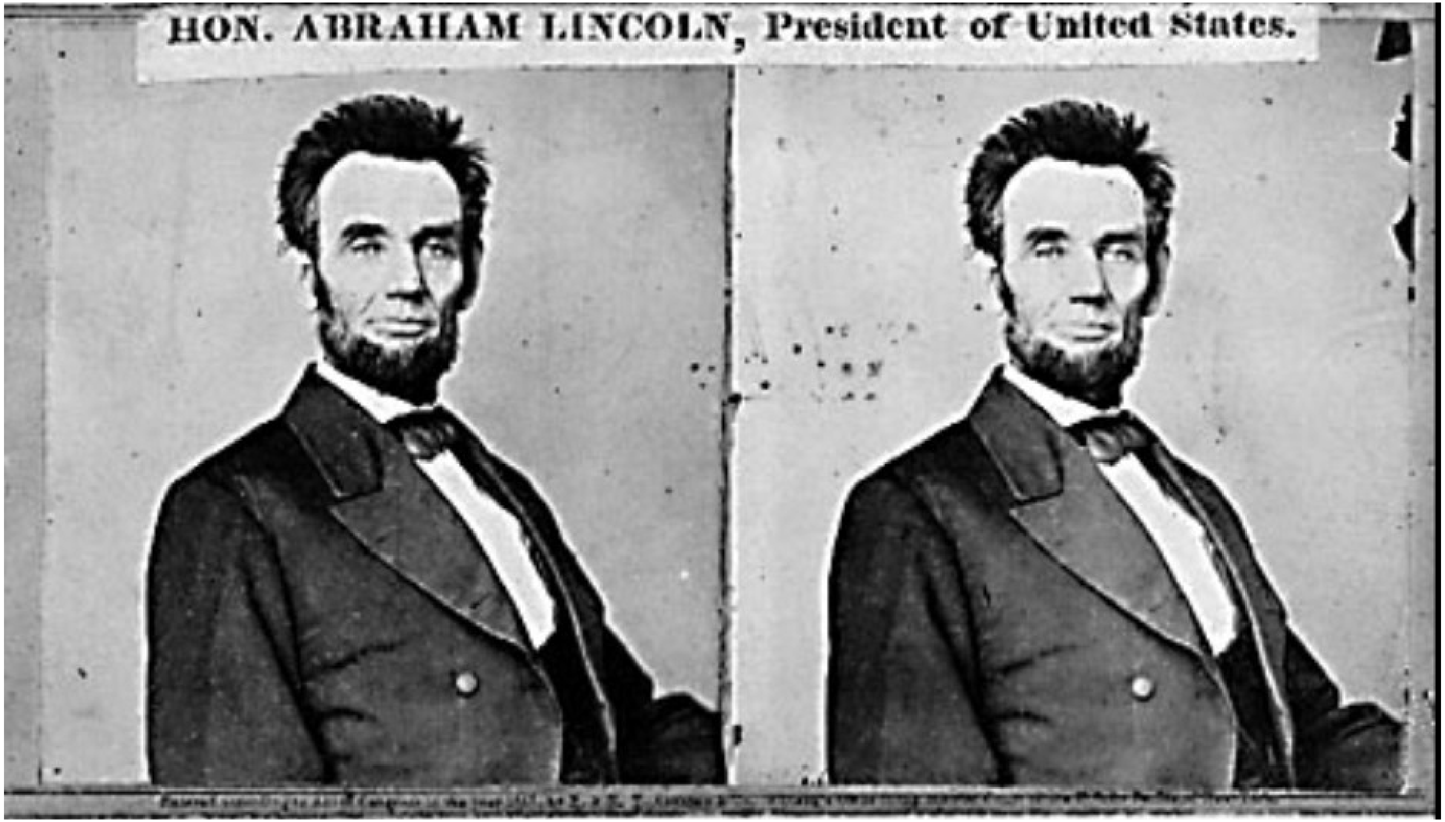
Disparity

- What is the difference between those images?

Disparity

Disparity is useful?

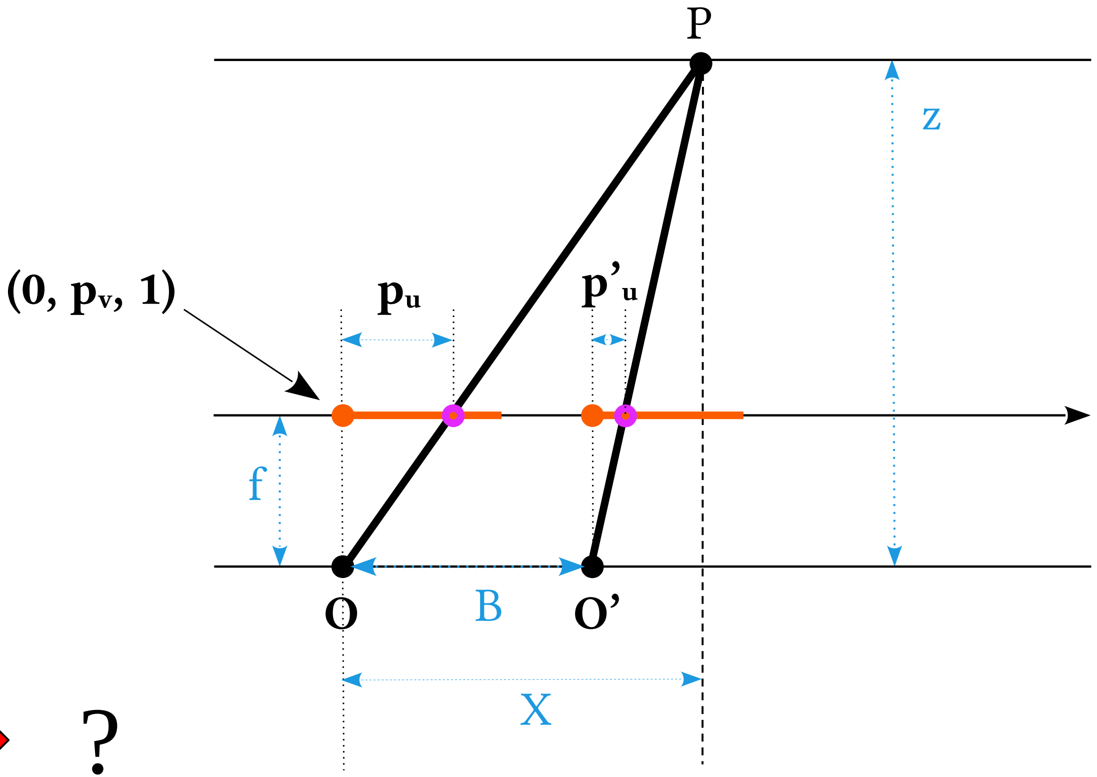
Disparity is useful?



Disparity is useful?

Disparity

Disparity



$p_u - p'_u$ ➡ ?

Disparity

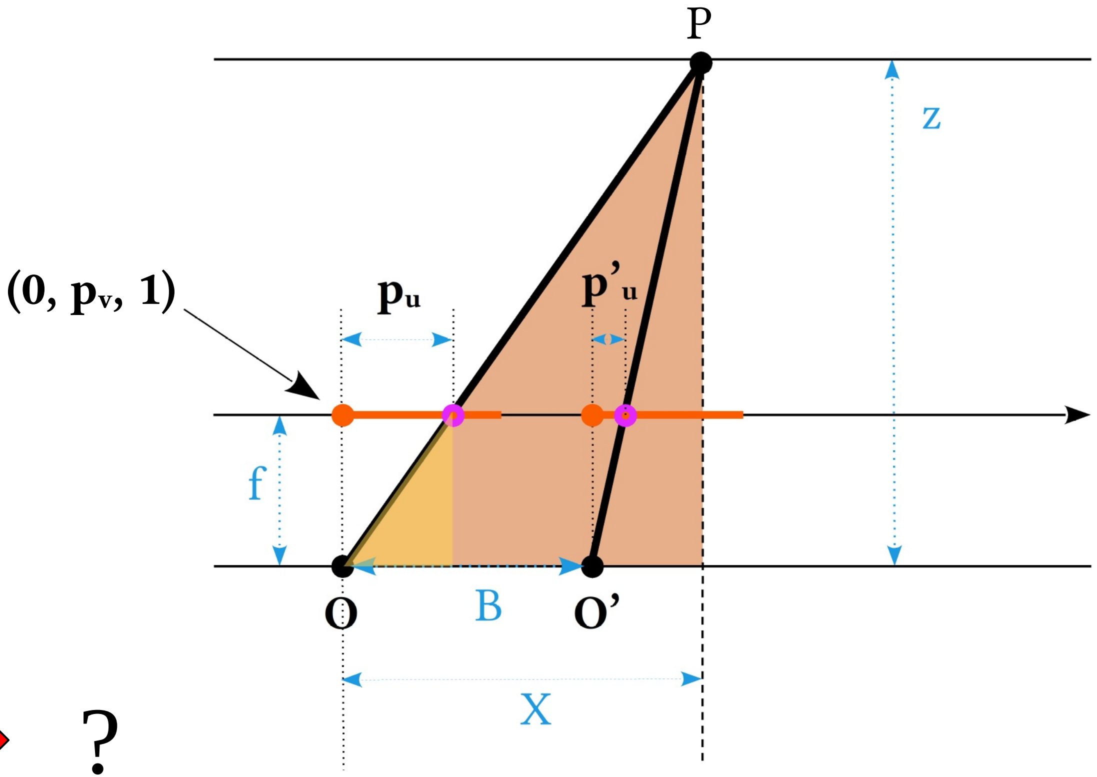
UNIVERSITÀ DI PARMA

The yellow right triangle features f and p_u as sides

The brown one features Z and X

$$p_u = f\,\frac{X}{Z}$$

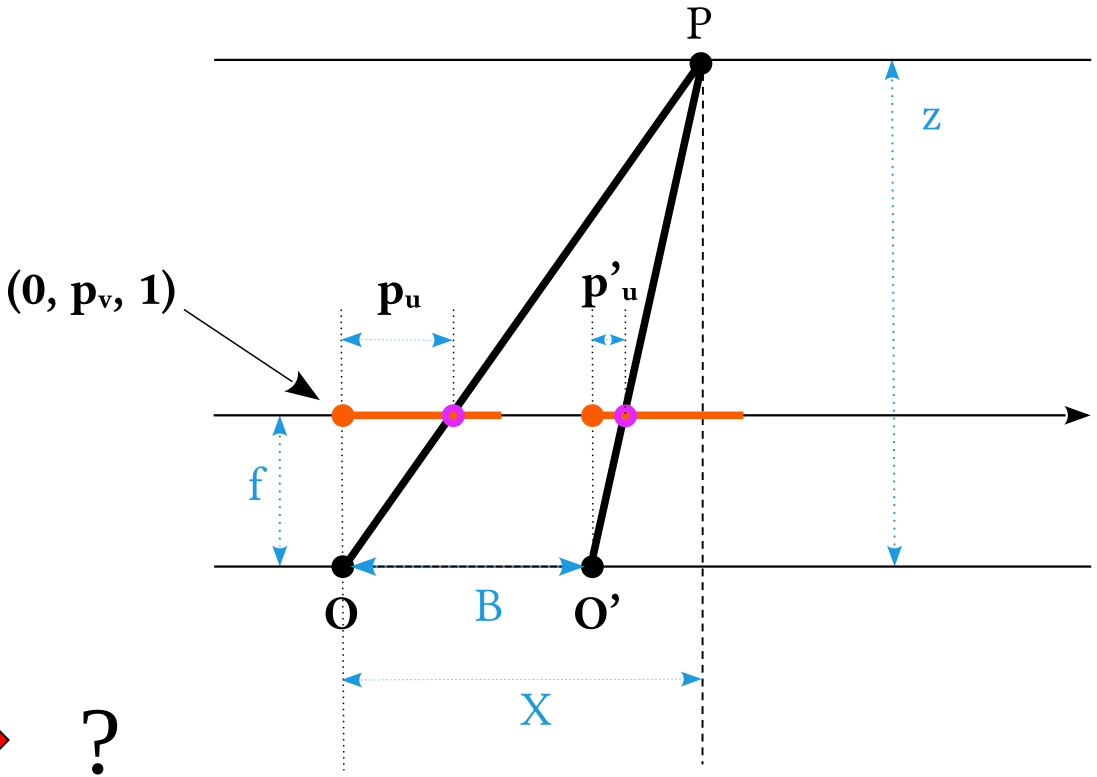
$$p_u - p'_u \quad \Rightarrow \quad ?$$



Disparity

UNIVERSITÀ DI PARMA

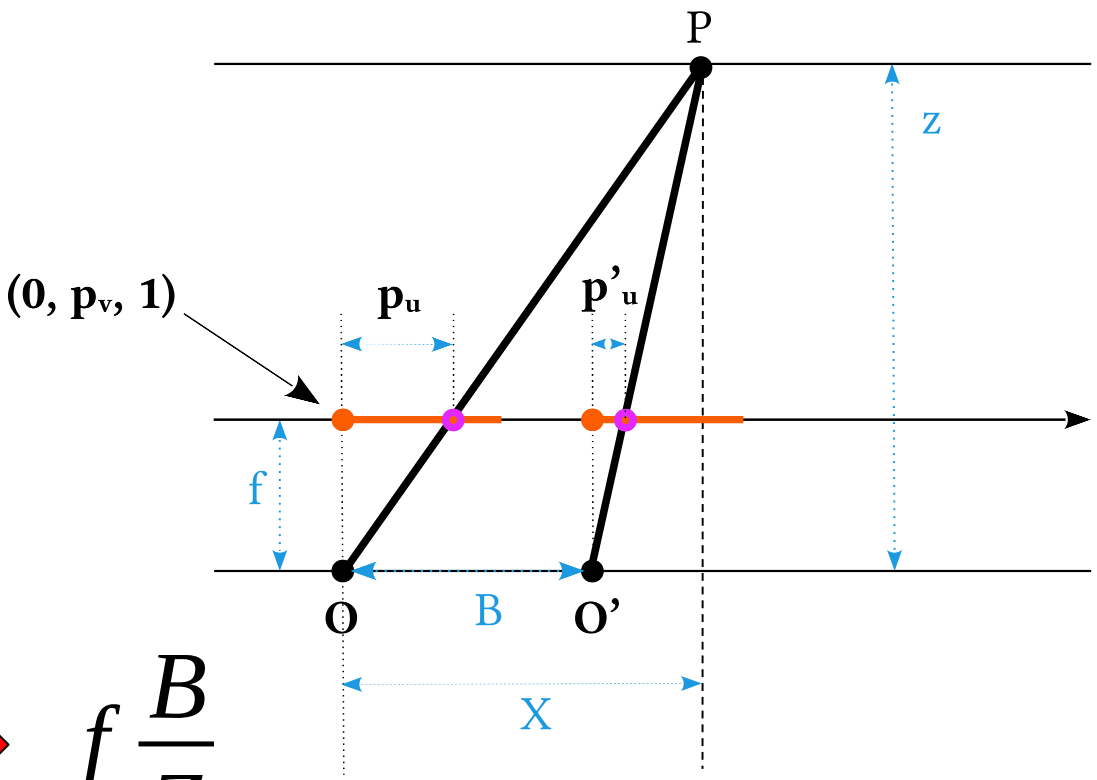
$$\begin{cases} p_u &= f\dfrac{X}{Z} \\ p'_u &= f\dfrac{X-B}{Z} \end{cases}$$



$p_u - p'_u$ ➡ ?

Disparity

$$\begin{cases} p_u = f\dfrac{X}{Z} \\ p'_u = f\dfrac{X-B}{Z} \end{cases}$$



$$p_u - p'_u \Rightarrow f\frac{B}{Z}$$

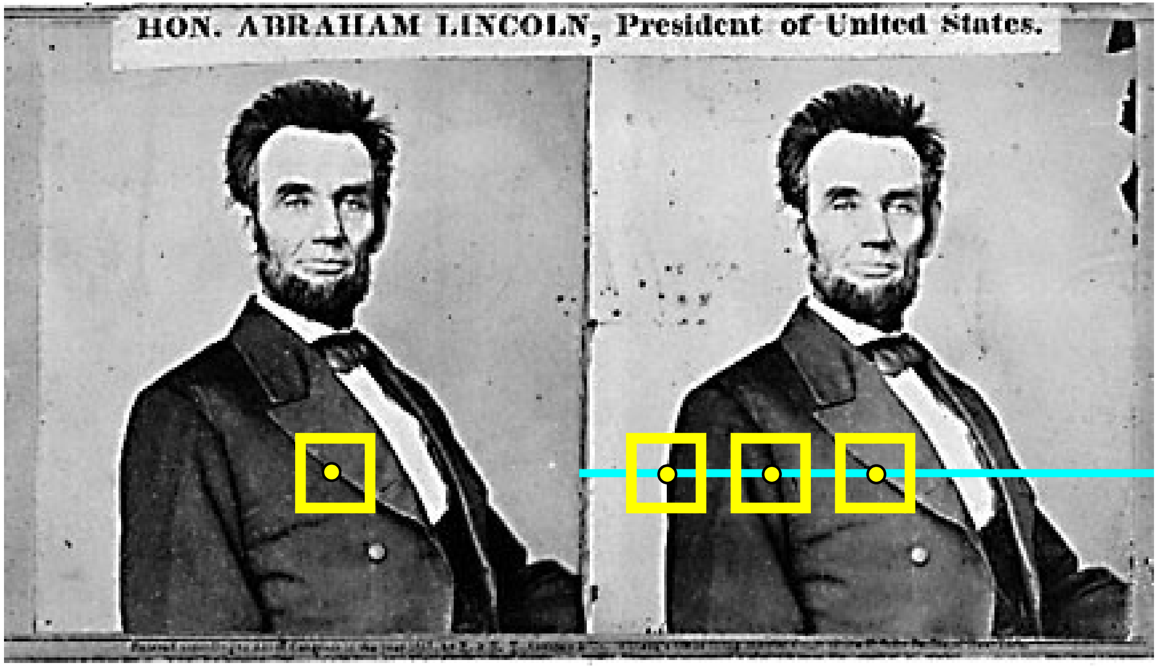
Disparity

- The p_u-p'_u "difference" is the disparity d
- Disparity is inversely proportional to Z
- Yes it can give us information about depth!

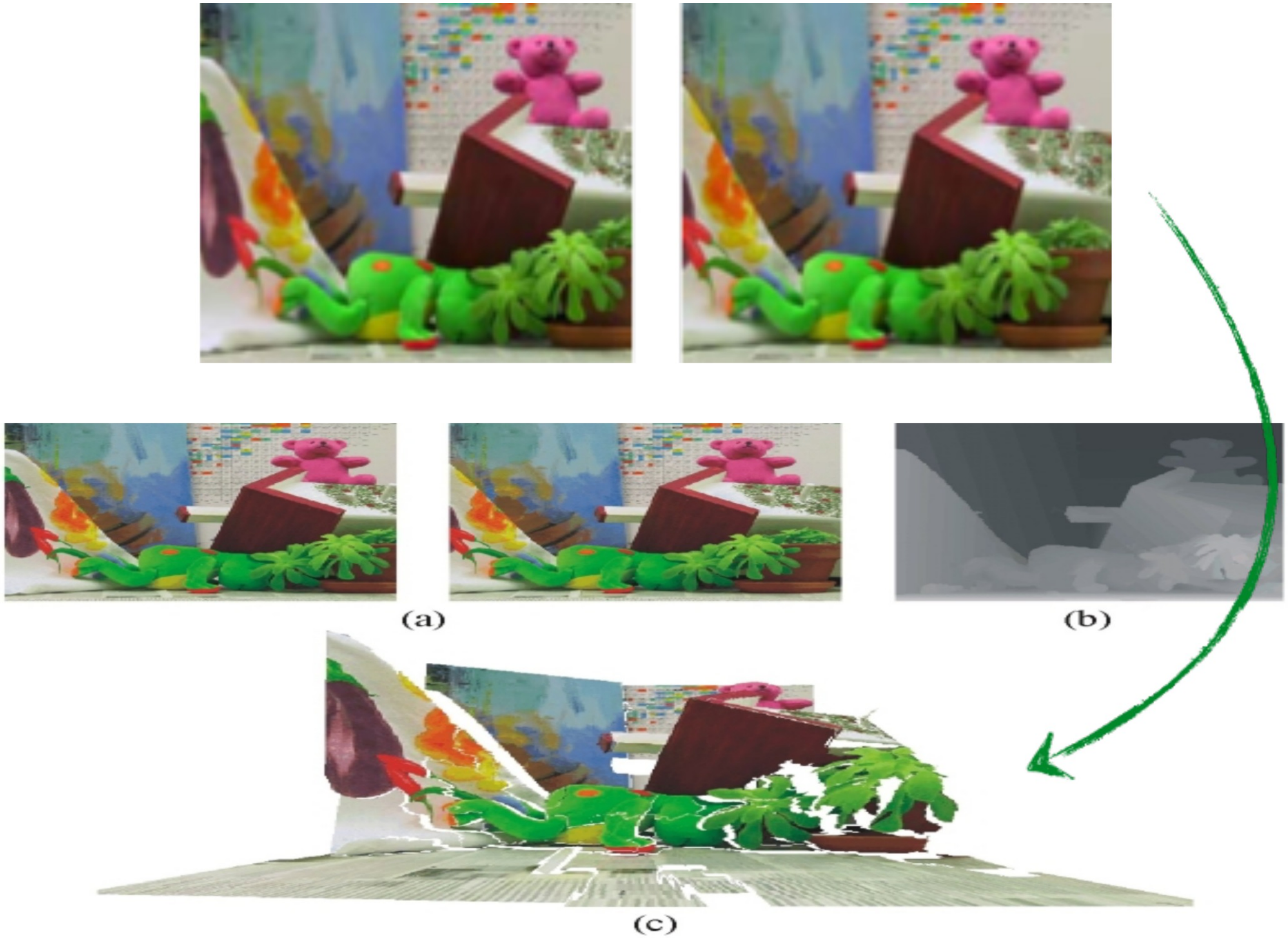
$$p_u - p'_u \quad \Rightarrow \quad d = f\frac{B}{Z}$$

Depth estimation

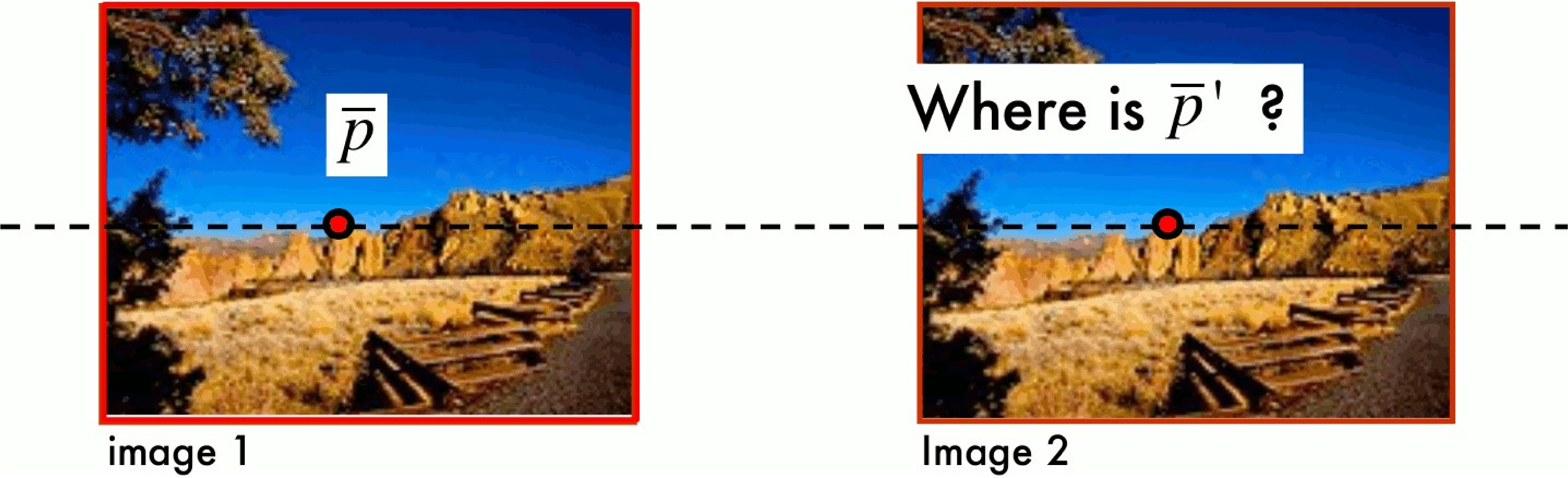
- Simple steps
  - Rectify images (strictly speaking not mandatory, but foul to not do that)
  - For each pixel
    - (Find epipolar line)
    - Find best match
    - Estimate Z



Depth estimation

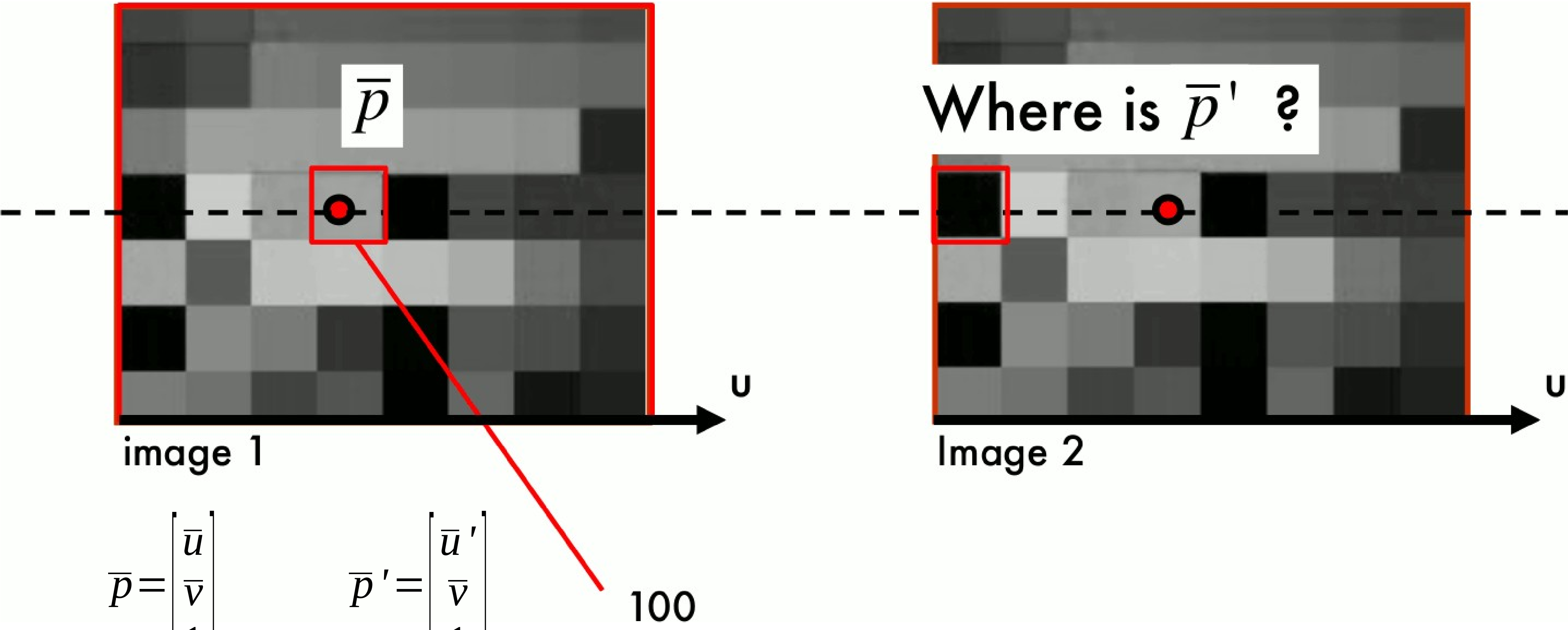


Find best match



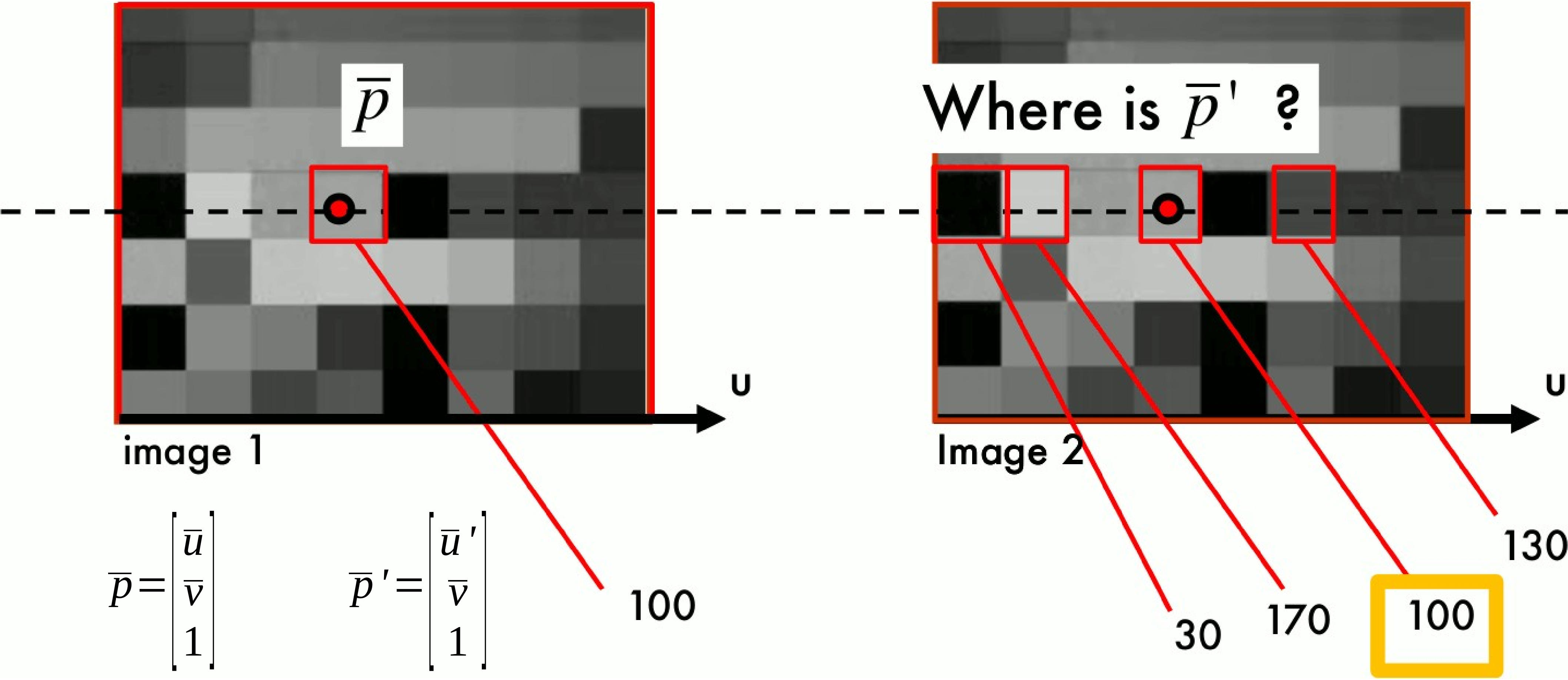
$$\bar{p} = \begin{bmatrix} \bar{u} \\ \bar{v} \\ 1 \end{bmatrix} \qquad \bar{p}' = \begin{bmatrix} \bar{u}' \\ \bar{v} \\ 1 \end{bmatrix}$$

Find best match



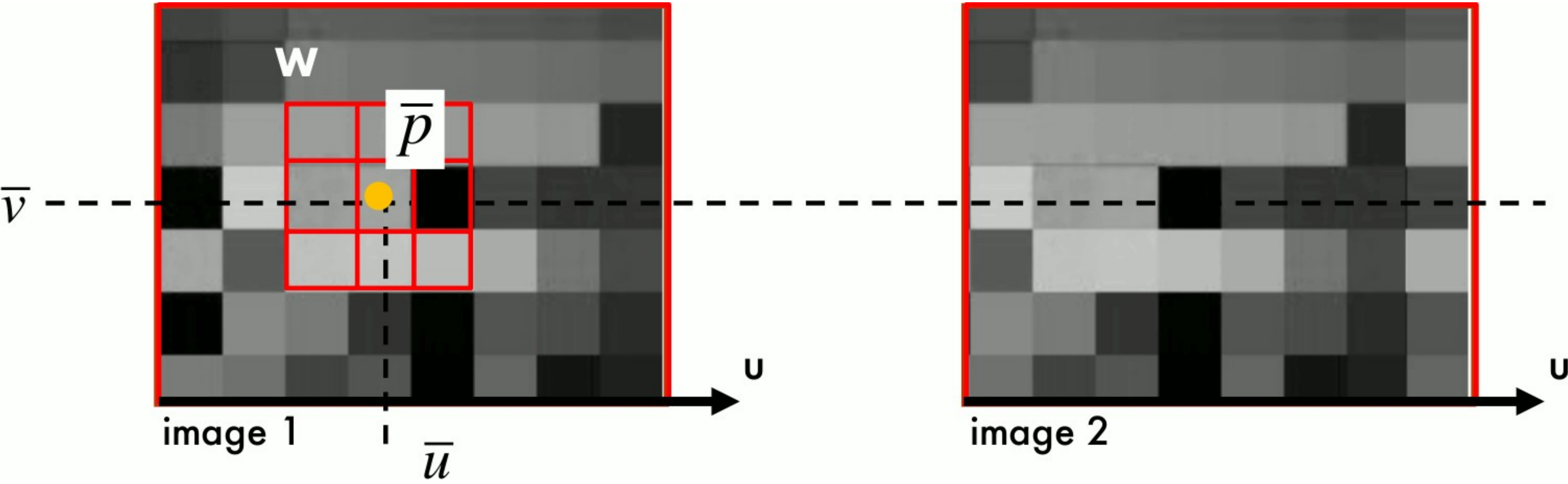
$$\overline{p} = \begin{bmatrix} \overline{u} \\ \overline{v} \\ 1 \end{bmatrix} \qquad \overline{p}' = \begin{bmatrix} \overline{u}' \\ \overline{v} \\ 1 \end{bmatrix}$$

Find best match



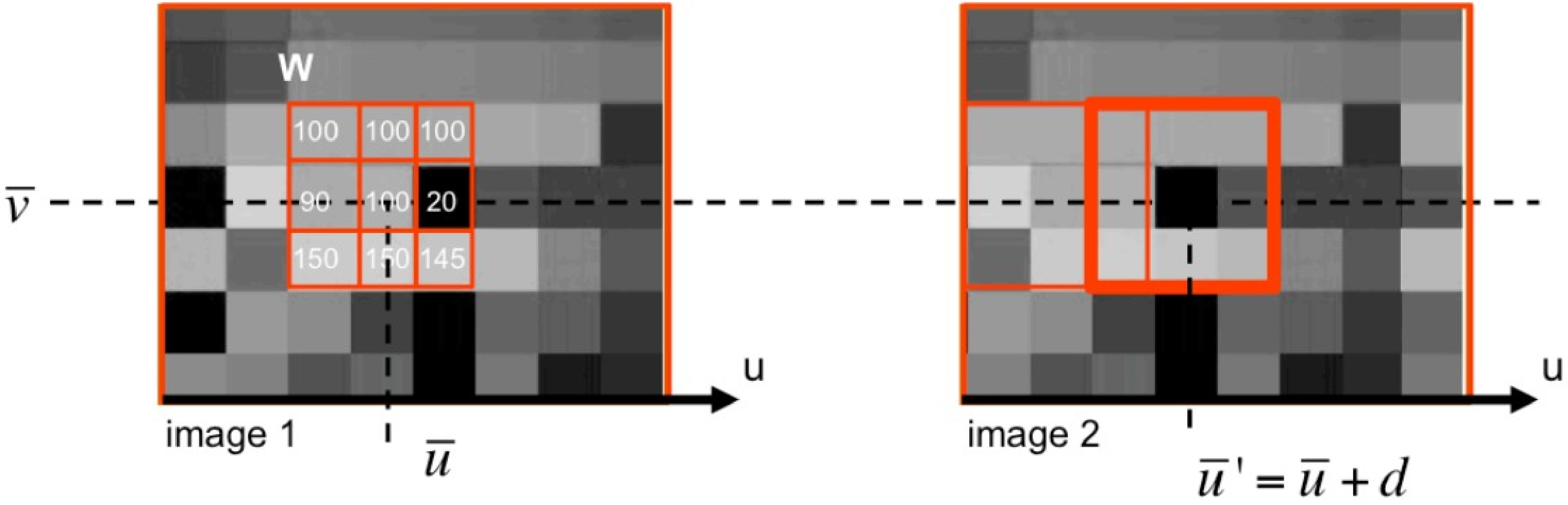
$$\bar{p} = \begin{bmatrix} \bar{u} \\ \bar{v} \\ 1 \end{bmatrix} \qquad \bar{p}' = \begin{bmatrix} \bar{u}' \\ \bar{v} \\ 1 \end{bmatrix}$$

Find best match



- We use a W window around $\bar{p}=(\bar{u},\bar{v})$
- Search for the same block in the other image

Find best match



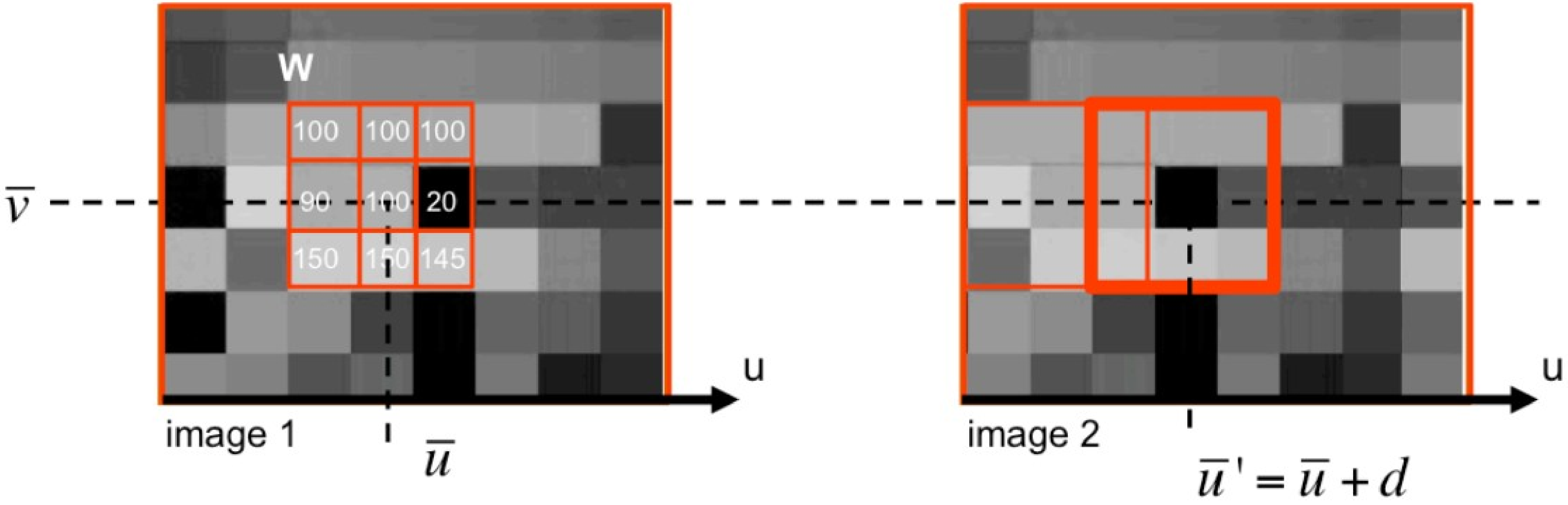
Example: **W** is a 3x3 window in red

w is a 9x1 vector

w = $[100, 100, 100, 90, 100, 20, 150, 150, 145]^T$

- Run along the same line $v=\bar{v}$ and compute W' for each position
- Match W vs W' → how we can do that?

Cross Correlation



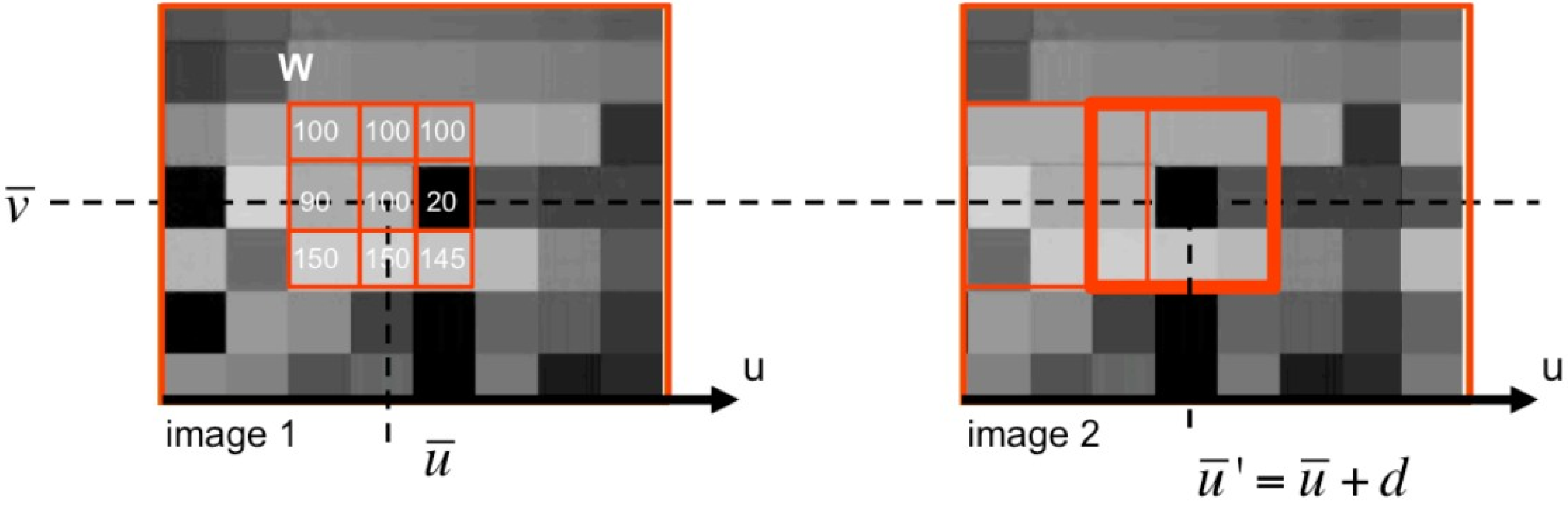
Example: **W** is a 3x3 window in red

w is a 9x1 vector

w = $[100, 100, 100, 90, 100, 20, 150, 150, 145]^T$

- Compute dot product $w^T \cdot w(u)$ for each u and compute max
  - Max correlation (considering w and w’ as vectors from W and W’)

Cross Correlation



Example: **W** is a 3x3 window in red

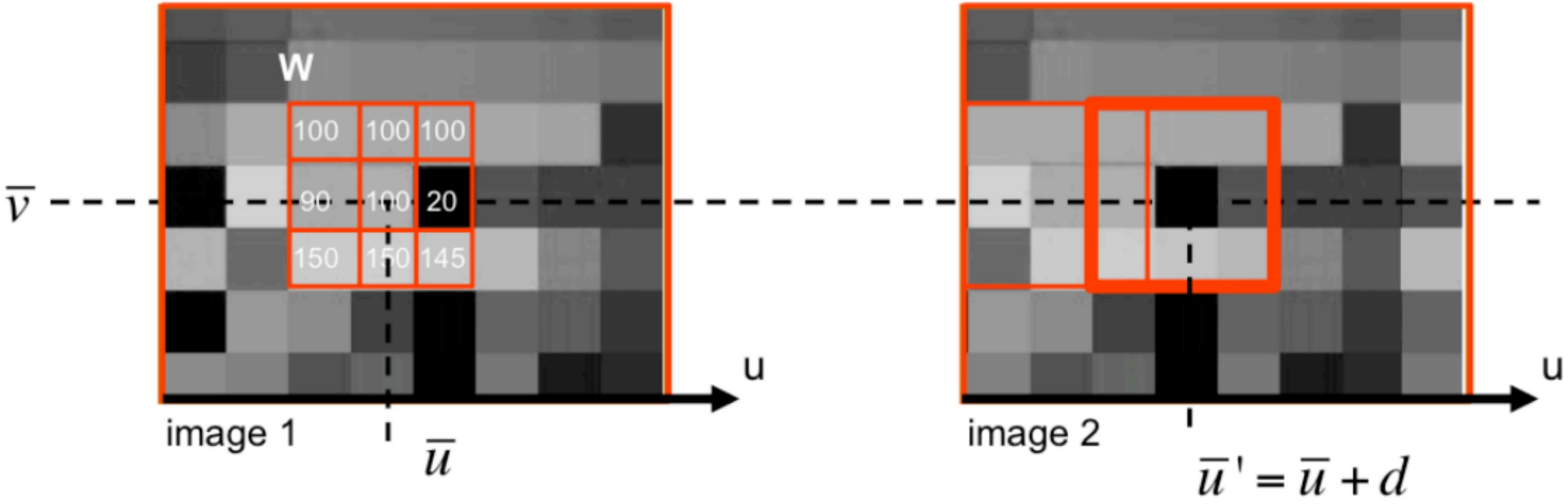
w is a 9x1 vector

w = $[100, 100, 100, 90, 100, 20, 150, 150, 145]^T$

- Highly affected by intensity variations
- Mismatches!

Cross Correlation

NCC: Normalized Cross Correlation



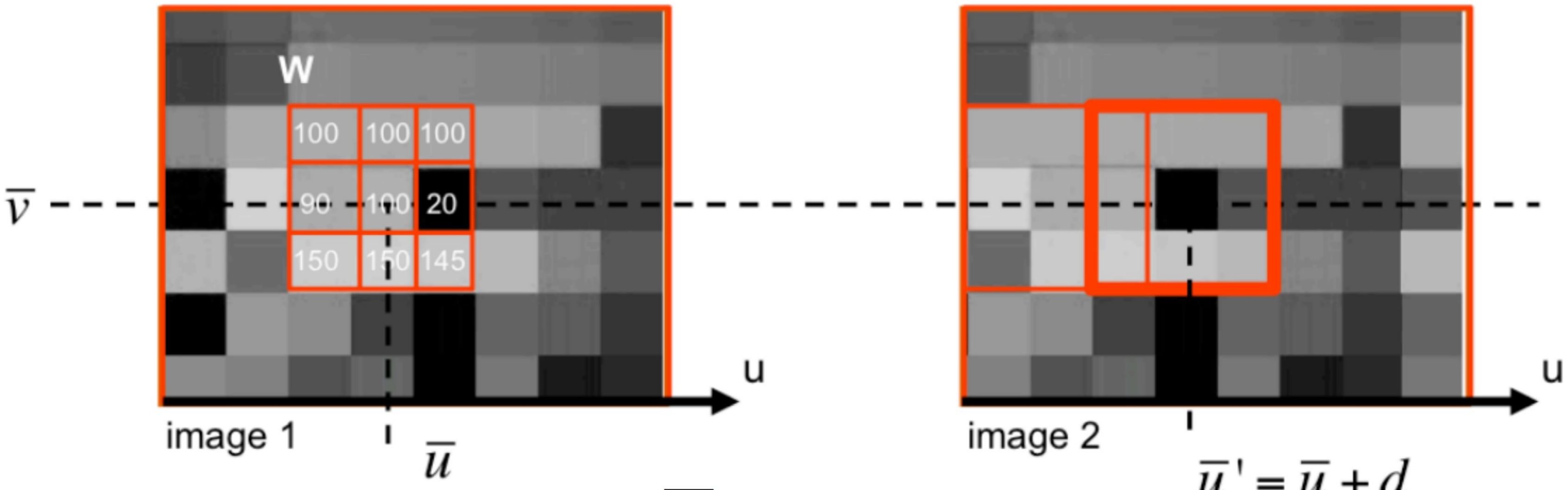
Find u that maximizes:

$$\frac{(w-\bar{w})^t (w'(u)-\bar{w}'(u))}{\|(w-\bar{w})\| \; \|(w'(u)-\bar{w}'(u))\|}$$

$\bar{w} =$ mean value within **W** located at u^{bar} in image 1

$\bar{w}'(u) =$ mean value within **W** located at u in image 2

SAD: Sum of Absolute Distances



$$SAD = \sum |w - w'(u)|$$

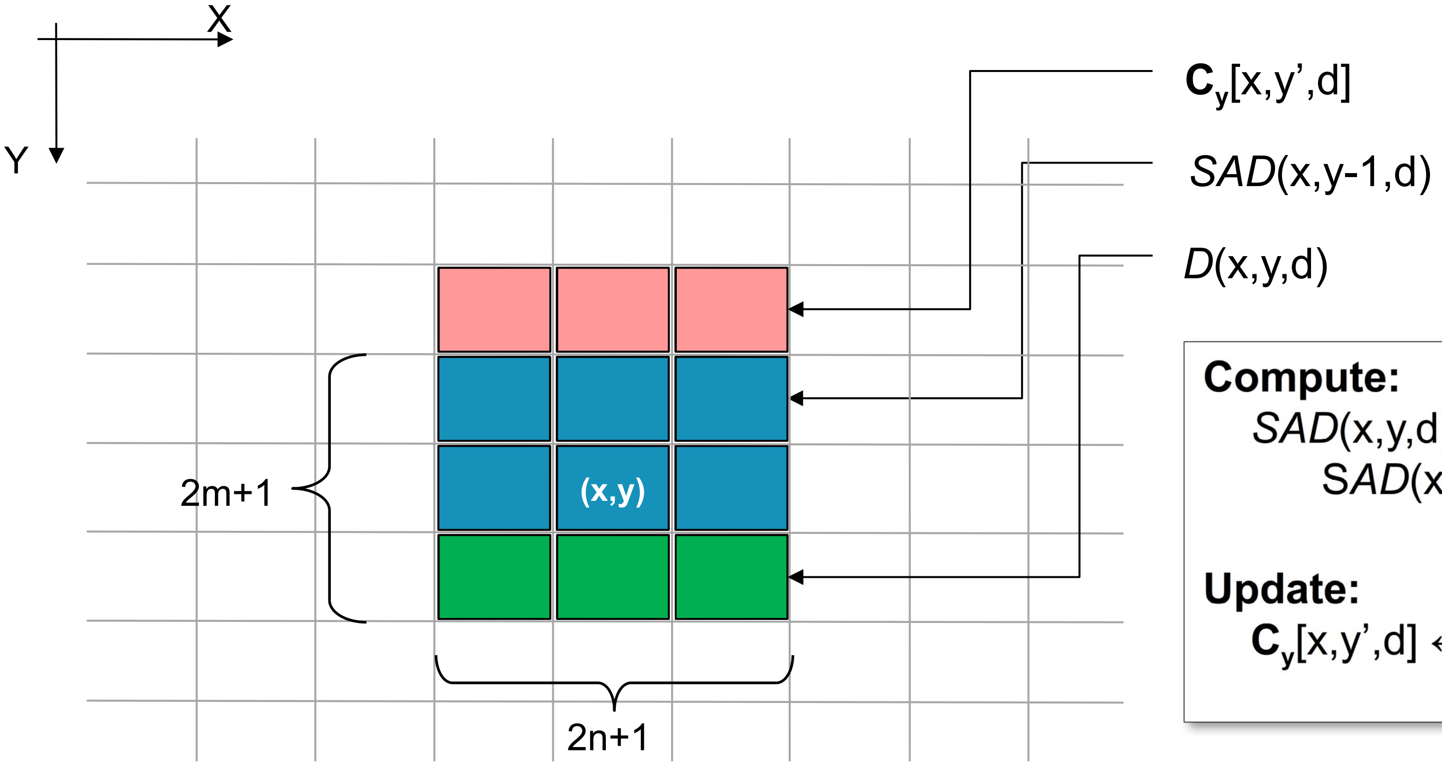
- Sum of absolute differences between W and W’ (or w and w’)
- Winner Takes All ⟶ only get the best match
- Easy to use other windows formats: squares, rectangles, adaptive…

SAD: Sum of Absolute Distances

UNIVERSITÀ DI PARMA

- Complexity: $O(WHDW_{win}H_{win})$
- Memory
  - No buffer required
- Highly dependent on window size
- Sparse maps can be efficiently computed
  - Compute stereo on features only

SAD: Semi-incremental Algo



Compute:

$SAD(x,y,d) =$

$SAD(x,y-1,d) - \mathbf{C}_y[x,y',d] + D(x,y,d)$

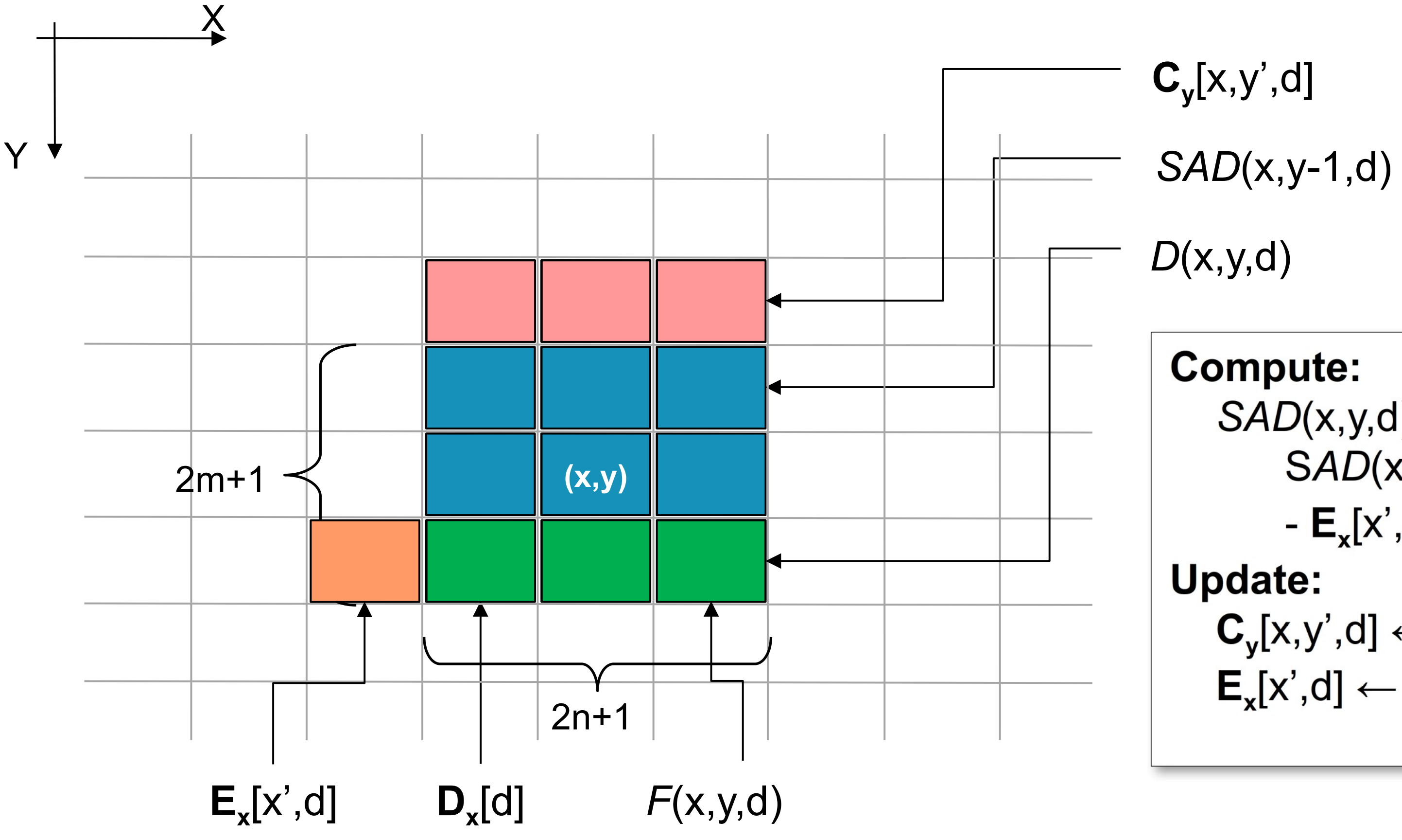
Update:

$\mathbf{C}_y[x,y',d] \leftarrow D(x,y,d)$

SAD: Semi-incremental Algo

- Complexity: $O(WHDW_{win})$
- Memory
  - C_y: WDH_{win} size vector
  - SAD: WD vector
- Sparse maps less efficiently computed

SAD: Full-incremental Algo



Compute:

$$SAD(x,y,d) = SAD(x,y-1,d) - \mathbf{C}_y[x,y',d] + \mathbf{D}_x[d] - \mathbf{E}_x[x',d] + F(x,y,d)$$

Update:

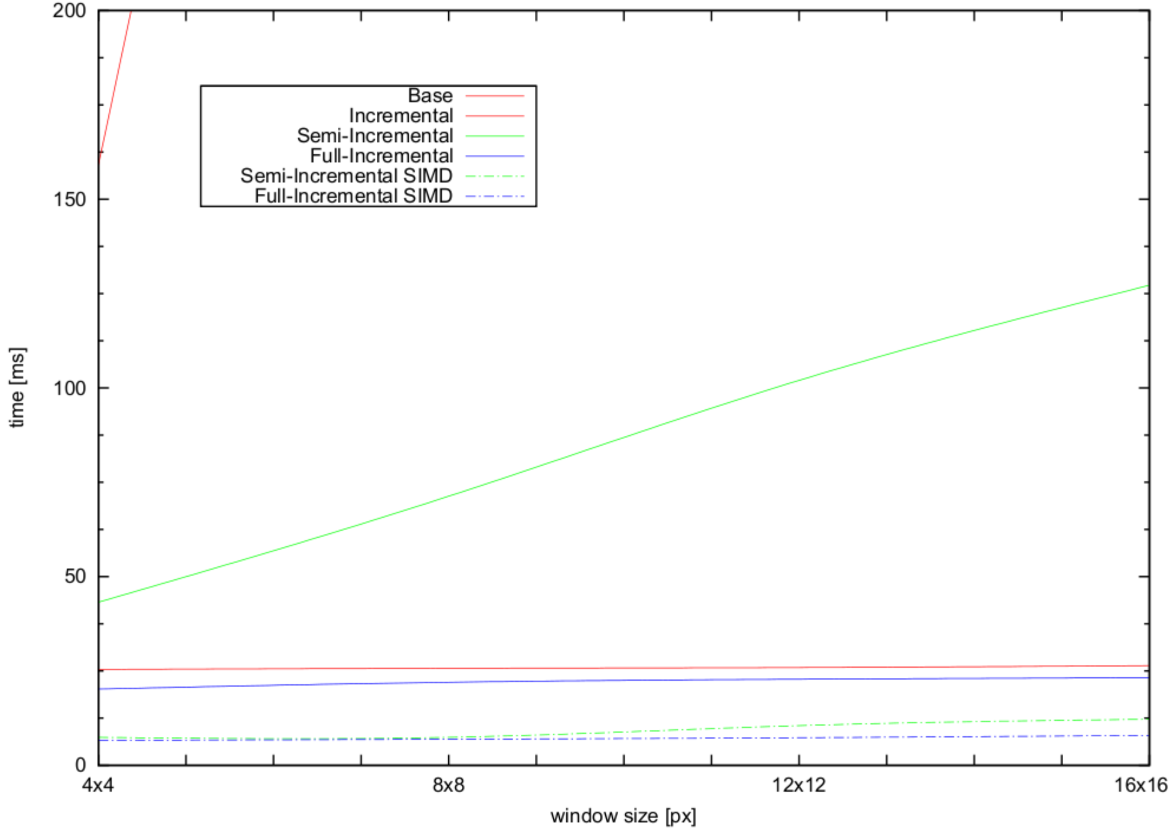
$$\mathbf{C}_y[x,y',d] \leftarrow \mathbf{D}_x[d] - \mathbf{E}_x[x',d] + F(x,y,d)$$

$$\mathbf{E}_x[x',d] \leftarrow F(x,y,d)$$

SAD: Full-incremental Algo

- Complexity: O(WHD)
- Memory
  - C_y: WDH_{win} size vector
  - E_x: W_{win} size vector
  - SAD: WD vector
- Sparse maps can not be efficiently computed
- Border problems

SAD approaches comparison



- 512x384 disparity map on a Intel Core i7@2.66 GHz

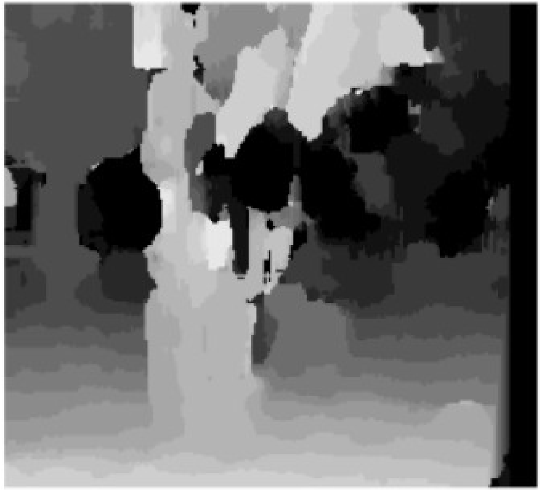
SAD: S

MATCH METRIC	DEFINITION
Normalized Cross-Correlation (NCC)	$\dfrac{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)\cdot\left(I_2(u+d,v)-\bar{I}_2\right)}{\sqrt{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)^2\cdot\left(I_2(u+d,v)-\bar{I}_2\right)^2}}$
Sum of Squared Differences (SSD)	$\sum_{u,v}\left(I_1(u,v)-I_2(u+d,v)\right)^2$
Normalized SSD	$\sum_{u,v}\left(\dfrac{\left(I_1(u,v)-\bar{I}_1\right)}{\sqrt{\sum_{u,v}\left(I_1(u,v)-\bar{I}_1\right)^2}}-\dfrac{\left(I_2(u+d,v)-\bar{I}_2\right)}{\sqrt{\sum_{u,v}\left(I_2(u+d,v)-\bar{I}_2\right)^2}}\right)^2$
Sum of Absolute Differences (SAD)	$\sum_{u,v}\left\|I_1(u,v)-I_2(u+d,v)\right\|$
Rank	$\sum_{u,v}\left(I_1'(u,v)-I_2'(u+d,v)\right)$ $I_k'(u,v)=\sum_{m,n}I_k(m,n)<I_k(u,v)$
Census	$\sum_{u,v}HAMMING\left(I_1'(u,v),I_2'(u+d,v)\right)$ $I_k'(u,v)=BITSTRING_{m,n}\left(I_k(m,n)<I_k(u,v)\right)$

Credits: Brown et al.

Size Matters

Window size = 3



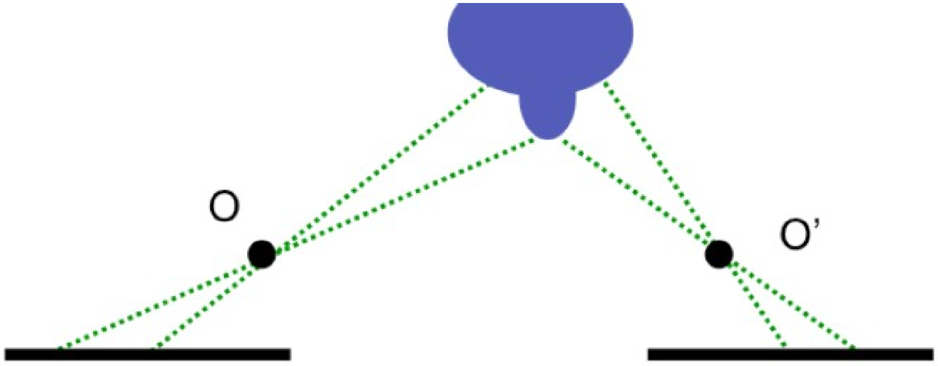
Window size = 20

- Small windows: more details ↔ more noise
- Bigger windows: less details ↔ less noise
    - disparity map is more uniform

Stereo is a difficult task!

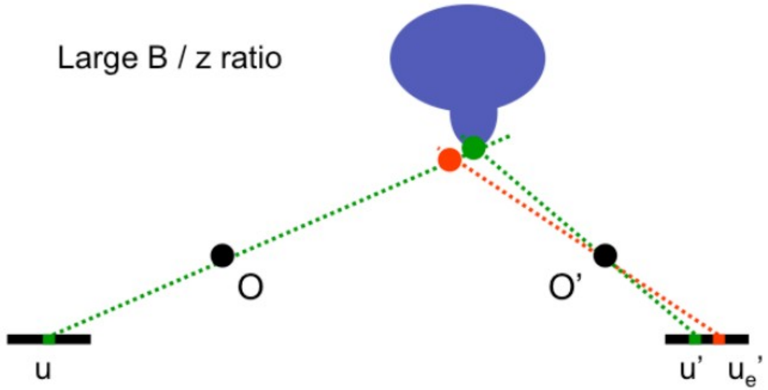
- Fore shortening effects

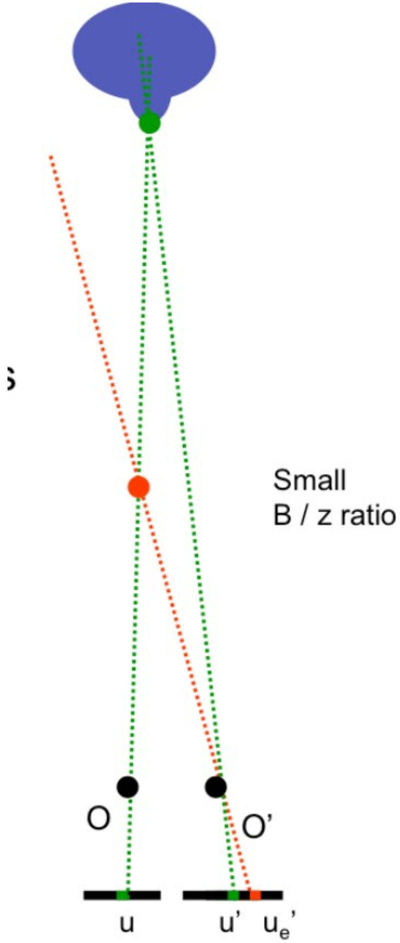
- Occlusions



Stereo is a difficult task!

- Those effects can be reduced reducing the baseline
  - Actually reducing B/z
- Depth estimation more difficult when B/z ratio is low!
  - Large errors

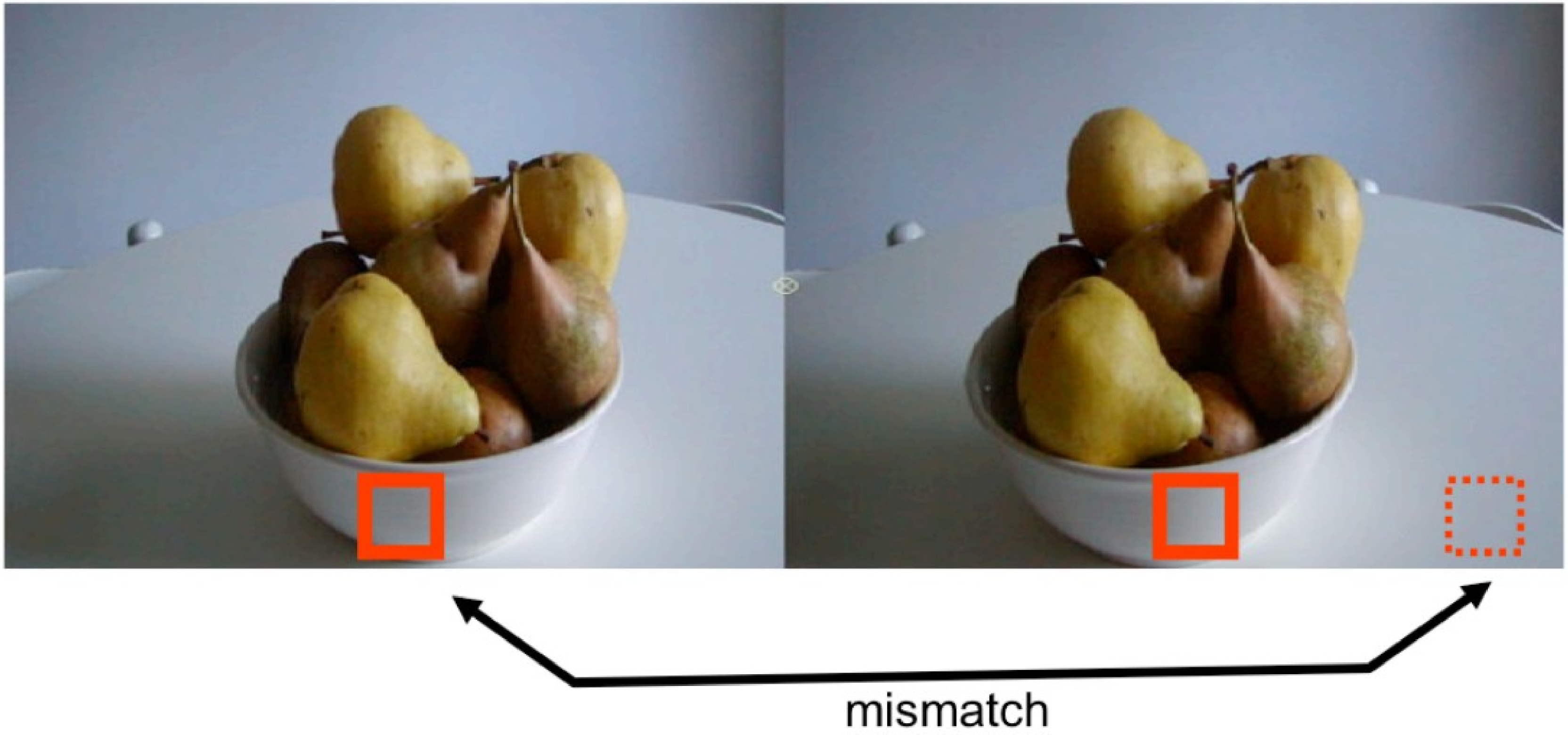




Trinocular systems

Stereo is a difficult task!

- Homogeneous regions



Stereo is a difficult task!

- Repetitive patterns

Stereo is a difficult task!

- Different view points $\longrightarrow$ fore shortening
- Occlusions
- Baseline trade-off
- Homogeneous regions
- Repetitive patterns

- More issues
  - Transparent objects
  - Reflections

Stereo is a difficult task!

UNIVERSITÀ DI PARMA

- How can we cope with those issues?
- Use non local constraints
  - Uniqueness
    - Only one match
  - Ordering
    - Matches should have the same order in both views
  - Continuity
    - Disparity should not change too quicly (except borders)

Depth from disparity

UNIVERSITÀ DI PARMA

- $(p_u, p_v, d) \longrightarrow$ 3D
- Camera reference system

$$d = p_u - p'_u \quad \Longrightarrow \quad z_c = \frac{B \cdot f}{d}$$

$$\begin{matrix} p_u \\ p_v \\ d \end{matrix} \quad \Longrightarrow \quad \begin{matrix} x_c \\ y_c \\ z_c \end{matrix}$$

$$\begin{cases} x' = p_u - u_0 = f \dfrac{x_c}{z_c} \\ y' = p_v - v_0 = f \dfrac{y_c}{z_c} \end{cases} \quad \Longrightarrow \quad \begin{cases} x_c = \dfrac{(p_u - u_0) \cdot B}{d} \\ y_c = \dfrac{(p_v - v_0) \cdot B}{d} \\ z_c = \dfrac{B \cdot f}{d} \end{cases}$$

Depth from disparity

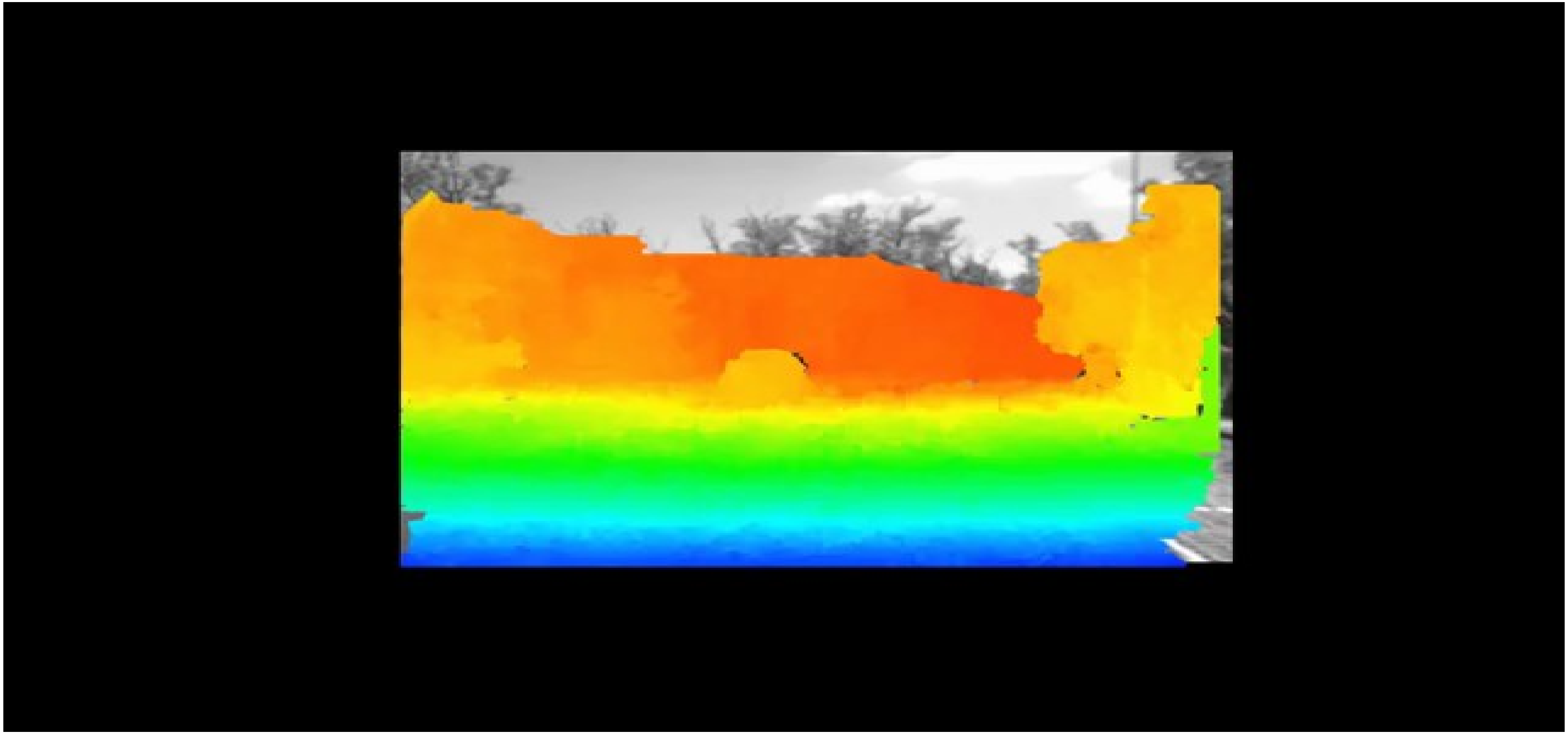
$$\begin{bmatrix} R & T \\ 0 & 1 \end{bmatrix}^T \cdot \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} = \begin{bmatrix} Y_w \\ Y_w \\ Z_w \end{bmatrix}$$

- RT can help us to compute world coordinates

Recap about depth estimation

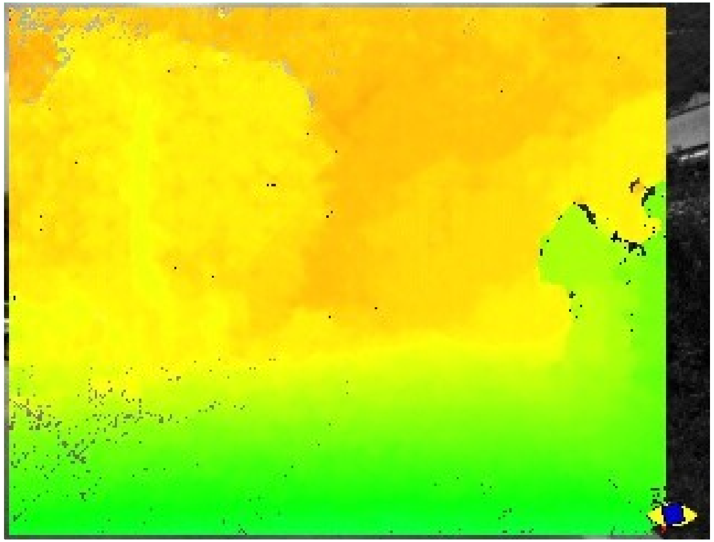
- We need to calibrate both cameras
- Rectify images $\longrightarrow$ parallel image planes
- Search for correspondences
- Transform disparity in world coordinates

Depth from disparity



Semi Global Matching

- SGM allows to obtain **dense** disparity maps
  - Cost function based on disparity directions



UNIVERSITÀ DI PARMA

Stereo Matching

Question time!