3D Computer Vision

Radim Šára   Martin Matoušek

Center for Machine Perception
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague

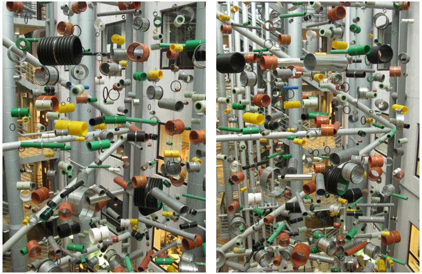
https://cw.fel.cvut.cz/wiki/courses/tdv/start

http://cmp.felk.cvut.cz
mailto:sara@cmp.felk.cvut.cz
phone ext. 7203

rev. December 12, 2023

Open Informatics Master's Course

How Difficult Is Stereo?



Centrum för teknikstudier at Malmö Högskola, Sweden

The Vyšehrad Fortress, Prague

- top: easy interpretation from even a single image
- bottom left: we have no help from image interpretation
- bottom right: ambiguous interpretation due to a combination of missing texture and occlusion

A Summary of Our Observations and an Outlook

1. simple matching algorithms do not work
   - the success of a model-free stereo matching is unlikely →158
   - without scene recognition or use high-level constraints the problem seems difficult
2. stereopsis requires image interpretation in sufficiently complex scenes — or another-modality measurement

we have a tradeoff: model strength ↔ universality

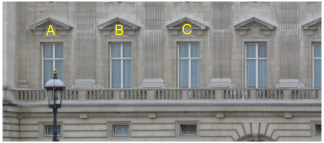
Outlook:

1. represent the occlusion constraint: — correspondences are not independent due to occlusions
   - disparity in rectified images
   - uniqueness as an occlusion constraint
2. represent piecewise continuity — the weakest of interpretations; piecewise: object boundaries
   - ordering as a weak continuity model
3. use a consistent framework
   - finding the most probable solution (MAP)

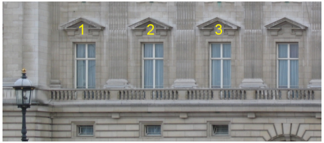
Structural Ambiguity in Stereovision

- suppose we can recognize local matches independently but have no scene model
- lack of an occlusion model
- lack of a continuity model

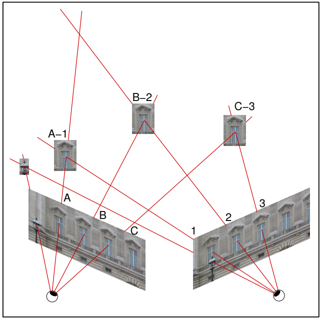
$\Rightarrow$ structural ambiguity in the presence of repetitions (or lack of texture)



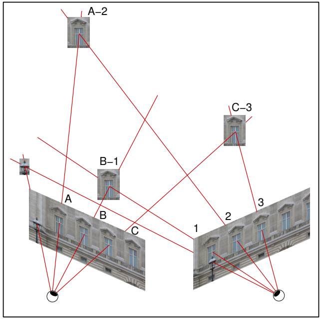
left image



right image



matching/interpretation 1

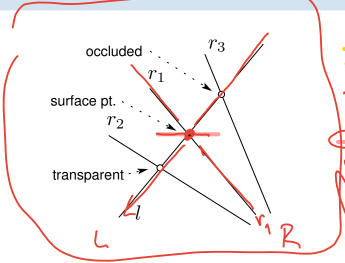


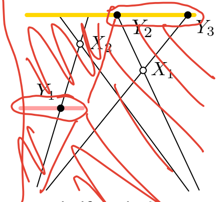
interpretation 2

- Illustration of the problem
- Keypoints: Window detections

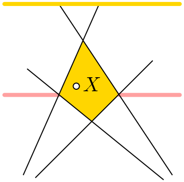
- Repetitive keypoints $\Rightarrow$ non-unique matching
- Cameras are not canonical; constant-depth surface is not a plane

▶Understanding Basic Occlusion Types





half occlusion



mutual occlusion

mutual occlusion in the wall hole

- surface point at the intersection of rays l and r_1 occludes a world point at the intersection (l, r_3) and implies the world point (l, r_2) is transparent, therefore

(l, r_3) and (l, r_2) are excluded by (l, r_1)

- in half-occlusion, every 3D point such as X_1 or X_2 is excluded by a binocularly visible surface point such as Y_1, Y_2, Y_3

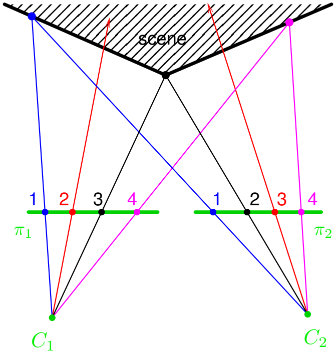
  $\Rightarrow$ decisions on correspondences are not independent

- in mutual occlusion this is no longer the case: any X in the yellow zone above is not excluded

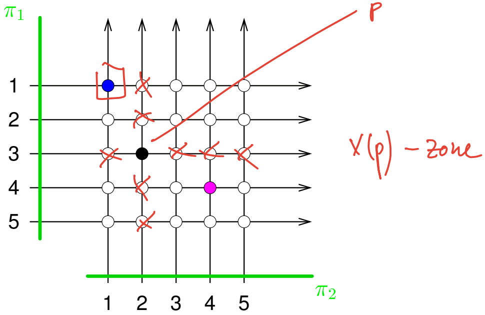
  $\Rightarrow$ decisions inside the zone are independent on the rest

▶Matching Table

Based on scene opacity and the observation on mutual exclusion we expect each pixel to match at most once.



rays in epipolar plane



matching table T

matching table

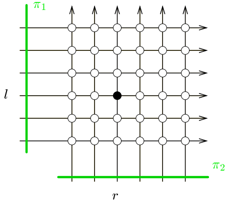
- rows and columns represent optical rays
- nodes: possible correspondence pairs
- full nodes: matches
- numerical values associated with nodes: descriptor similarities

see next

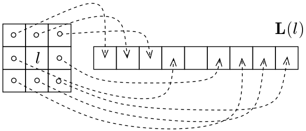
►Constructing An Image Similarity Cost

- let $p_i = (l, r)$ and $\mathbf{L}(l)$, $\mathbf{R}(r)$ be (left, right) image descriptors (vectors) constructed from local image neighborhood windows

in matching table T:



'block' in the left image $\mapsto$ 'a set of random-variable samples':



αL , R

- a simple block (dis-)similarity is $\operatorname{SAD}(l, r) = \|\mathbf{L}(l) - \mathbf{R}(r)\|_1$ — L_1 metric (sum of absolute differences; smaller is better)
- a scaled-descriptor (dis-)similarity is $\operatorname{sim}(l, r) = \dfrac{\|\mathbf{L}(l) - \mathbf{R}(r)\|^2}{\sigma_I^2(l, r)}$ — smaller is better
- σ_I^2 – the difference scale; a suitable (plug-in) estimate is $\frac{1}{2}\left[\operatorname{var}\big(\mathbf{L}(l)\big) + \operatorname{var}\big(\mathbf{R}(r)\big)\right]$, giving

$$\operatorname{sim}(l, r) = 1 - \underbrace{\frac{2\,\operatorname{cov}\big(\mathbf{L}(l), \mathbf{R}(r)\big)}{\operatorname{var}\big(\mathbf{L}(l)\big) + \operatorname{var}\big(\mathbf{R}(r)\big)}}_{\rho\left(\mathbf{L}(l), \mathbf{R}(r)\right)} \tag{38}$$

$\operatorname{var}(\cdot)$, $\operatorname{cov}(\cdot)$ is sample (co-)variance, not invariant to scale difference

- ρ – MNCC – Moravec's Normalized Cross-Correlation similarity — bigger is better [Moravec 1977]

$$\rho^2 \in [0, 1], \qquad \operatorname{sign}\rho \sim \text{'phase'}$$

- another successful (dis-)similarity is the Hamming Distance over the Census Transform — related to local binary patterns

Census Transform (CT)

- CT: Per-pixel binarization, given reference value (e.g the window center)
- For a grayscale image:

$$\varkappa(x_{ij};\, r) = \begin{cases} 0 & x_{ij} \le r \\ 1 & x_{ij} > r \end{cases}$$

189	235	181
217	185	228
231	61	254

input window

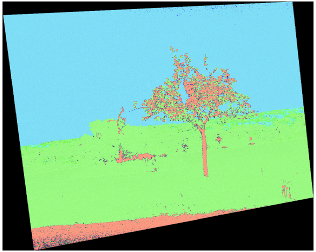
$\xrightarrow{\varkappa(\cdot;\, 185)}$

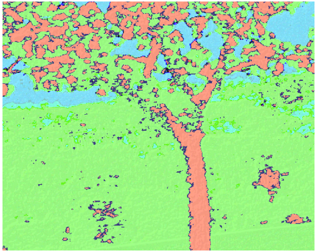
1	1	0
1	0	1
1	0	1

bits

$= 483$ (new value for the central pixel)

input image



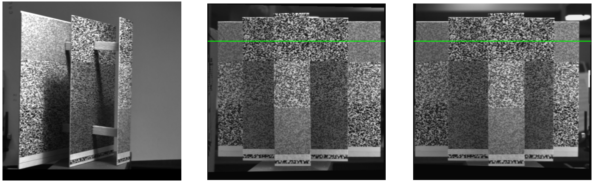


$\varkappa$: RGB CT, $3 \times 2 = 6$-bit per pixel, 3×3 window $= 48$ bit/px

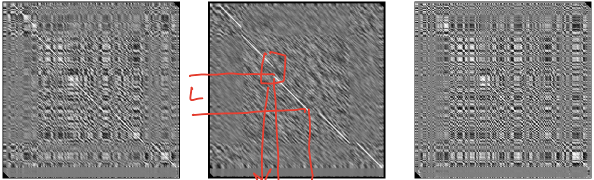
- preserves sharp boundaries
- may or may not use windowing (cost aggregation)

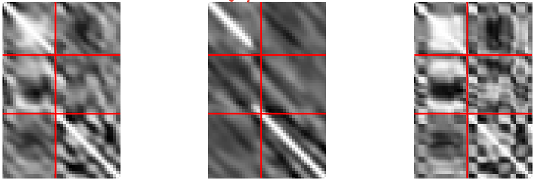
How A Scene Looks in The Filled-In Matching Table

object | left image | right image



5×5 window | 11×11 window | 3×3 window





a good tradeoff | occlusion artefacts | undiscriminable

- MNCC ρ used ($\alpha = 1.5,\ \beta = 1$) →182
- high-similarity structures correspond to scene objects

Things to notice:

constant disparity

- a diagonal in the matching table
- zero disparity is the main diagonal

assuming standard stereopair

depth discontinuity

- horizontal or vertical jump in matching table

large image window

- similarity values have better discriminability
- worse occlusion localization

repeated texture

- horizontal and vertical block repetition

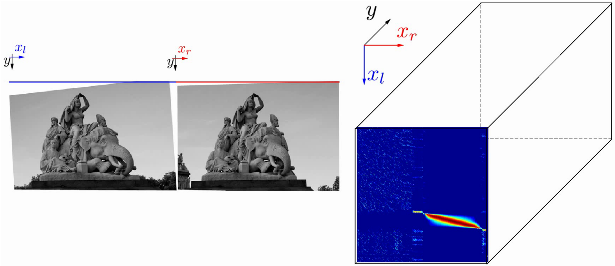
Descriptors: Image points are tagged by their (viewpoint-invariant) physical properties:

- texture window [Moravec 77]
- Census Transform [Zabih & Woodfill 94]
- a descriptor like DAISY [Tola et al. 2010]
- learned descriptors
- reflectance profile under a moving illuminant
- (pixelwise) photometric ratios [Wolff & Angelopoulou 93-94]
- dual photometric stereo [Ikeuchi 87]
- (pixelwise) polarization signature
- . . .

BDRF

- similar points are more likely to match
- image similarity values for all 'match candidates' give the 3D matching table — also called: 'disparity volume'

10^9

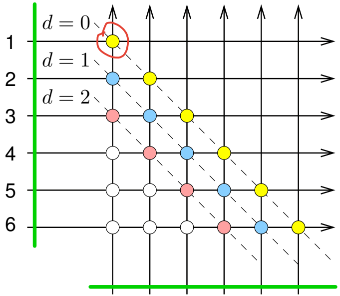


click for video

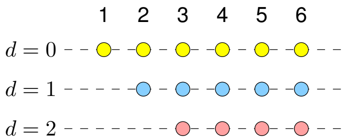
►Marroquin's Winner Take All (WTA) Matching Algorithm

Alg: Per left-image pixel: The most SAD-similar pixel along the right epipolar line →174

1. select disparity range — this is a critical weak point
2. represent the matching table diagonals in a compact form

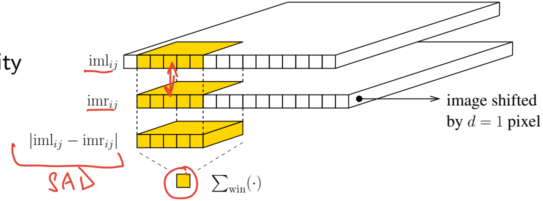


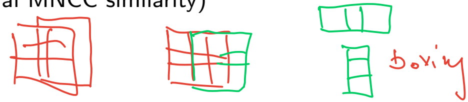
left img pixel



pixels

3. use the 'image sliding & cost aggregation algorithm'
4. take the maximum over disparities d
5. threshold results by the maximal allowed SAD dissimilarity (or minimal MNCC similarity)



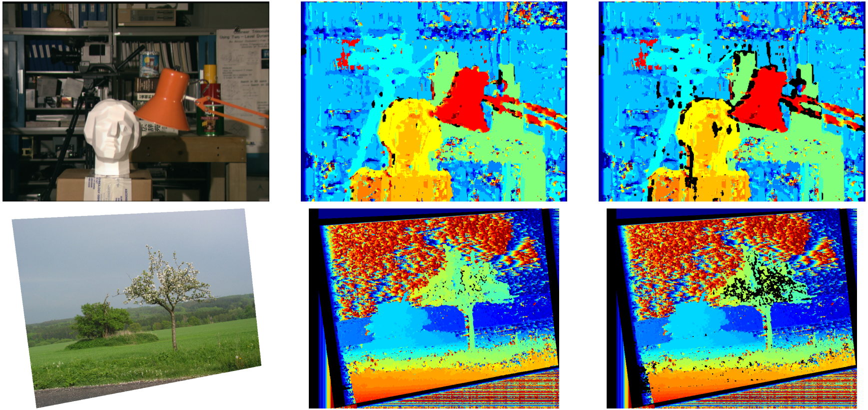


A Matlab Code for WTA

```
function dmap = marroquin(iml, imr, disparityRange)
%       iml, imr - rectified gray-scale images
% disparityRange - non-negative disparity range

% (c) Radim Sara (sara@cmp.felk.cvut.cz) FEE CTU Prague, 10 Dec 12

 thr = 20;                                       % bad match rejection threshold
 r = 2;
 winsize = 2*r+[1 1];                            % 5x5 window (neighborhood) for r=2
 N = boxing(ones(size(iml)), winsize);           % the size of each local patch is
                                                 % N = (2r+1)^2 except for boundary pixels

 % --- compute dissimilarity per pixel and disparity --->

 for d = 0:disparityRange                        % cycle over all disparities
  slice = abs(imr(:,1:end-d) - iml(:,d+1:end));  % pixelwise dissimilarity (unscaled SAD)   aggregation
  V(:,d+1:end,d+1) = boxing(slice, winsize)./N;  % window aggregation
 end

 % --- collect winners, threshold, output disparity map --->

 [cmap,dmap] = min(V,[],3);   <- WTA            % collect winners and their dissimilarities
 dmap(cmap > thr) = NaN;                         % mask-out high dissimilarity pixels
end % of marroquin

function c = boxing(im, wsz)
 % if the mex is not found, run this slow version:
 c = conv2(ones(1,wsz(1)), ones(wsz(2),1), im, 'same');
end % of boxing
```

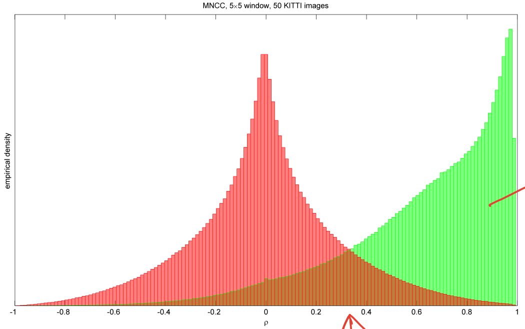
WTA: Some Results

thr = 20 (weaker) thr = 10 (stronger)



- results are fairly bad
- false matches in textureless image regions and on repetitive structures (book shelf)
- a more restrictive threshold (thr = 10) does not work as expected
- we searched the true disparity range, results get worse if the range is set wider
- chief failure reasons:
  - unnormalized image dissimilarity does not work well
  - no occlusion model (it just ignores the occlusion structure we have discussed $\rightarrow$172)

Empirical Distribution of MNCC ρ for Matches (green) and Non-Matches (red)



- histograms of ρ computed from 5×5 correlation window

ρ: bigger is better

- KITTI dataset
  - $4.2 \cdot 10^6$ ground-truth (LiDAR) matches for $p_1(\rho)$ (green),
  - $4.2 \cdot 10^6$ random non-matches for $p_0(\rho)$ (red)

Obs:

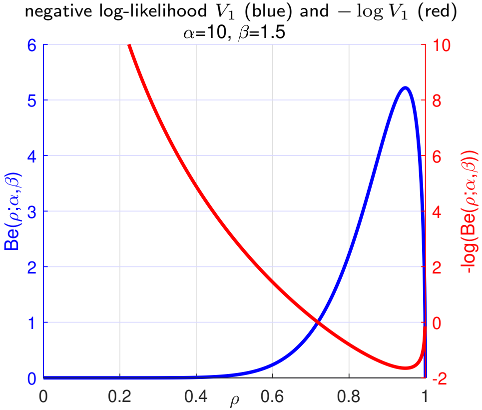
- non-matches (red) may have arbitrarily large ρ
- matches (green) may have arbitrarily low ρ
- $\rho = 1$ is improbable for matches

Match Likelihood

- ρ is just a normalized measurement
- we need a probability distribution on $[0, 1]$
  e.g. the histogram or the Beta distribution:

$$p_1(\rho) = \frac{1}{B(\alpha, \beta)} \, |\rho|^{\alpha-1}(1 - |\rho|)^{\beta-1}$$

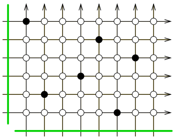
- note that uniform distribution is obtained for $\alpha = \beta = 1$
- when $\alpha = 2$ and $\beta = 1$ then $p_1(\cdot) = 2|\rho|$



- the mode is at $\sqrt{\frac{\alpha-1}{\alpha+\beta-2}} \approx 0.9733$ for $\alpha = 10,\ \beta = 1.5$
- if we chose $\beta = 1$ then the mode was at $\rho = 1$
- perfect similarity is 'suspicious' (depends on expected camera noise level)
- from now on we will work with negative log-likelihood cost

$$V_1\big(\rho(l,r)\big) = -\log p_1\big(\rho(l,r)\big) \qquad \text{smaller is better} \qquad (39)$$

- we should also define similarity (and negative log-likelihood $V_0(\rho(l,r))$) for non-matches



- given matching M in table T, what is the likelihood of observed data D?
- data – all cost pairs (V_0, V_1) in the matching table T
- matches – pairs $p_i = (l_i, r_i) \in M \subset T, \quad i = 1, \ldots, n^2$
- matching: partitioning matching table T to matched M and excluded E pairs

$$T = M \cup E, \quad M \cap E = \emptyset$$ partition

- matching cost (negative log-likelihood, smaller is better)

constant number of variables in T

$$\boxed{V(D \mid M, T) = \sum_{p \in M} V_1(D \mid p) + \sum_{p \in T \setminus M} V_0(D \mid p)}$$

comparable across M's

$V_1(D \mid p)$ – negative log-probability of data D at matched pixel p (39)
$V_0(D \mid p)$ – ditto at unmatched pixel p →181 and →182

- matching problem

$$M^* = \arg \min_{M \in \mathcal{M}(T)} V(D \mid M, T)$$ (n!)

$\mathcal{M}(T)$ – the set of all matchings in table T

- symmetric: formulated over pairs, invariant to left ↔ right image swap unlike in WTA

►(cont'd) Log-Likelihood Ratio

- we need to reduce the matching to a standard polynomial-complexity problem

1. convert the matching cost to an 'easier' sum

$$V(D \mid M, T) = \sum_{p \in M} V_1(D \mid p) + \sum_{p \in T \setminus M} V_0(D \mid p) + \overbrace{\sum_{p \in M} V_0(D \mid p) - \sum_{p \in M} V_0(D \mid p)}^{0}$$
$$= \sum_{p \in M} \underbrace{\Big(V_1(D \mid p) - V_0(D \mid p)\Big)}_{-L(D \mid p)} + \underbrace{\sum_{p \in T \setminus M} V_0(D \mid p) + \sum_{p \in M} V_0(D \mid p)}_{\sum_{p \in T} V_0(D \mid p) = \text{const}}$$

2. hence

$$\arg \min_{M \in \mathcal{M}(T)} V(D \mid M) = \arg \max_{M \in \mathcal{M}(T)} \sum_{p \in M} L(D \mid p) \tag{40}$$

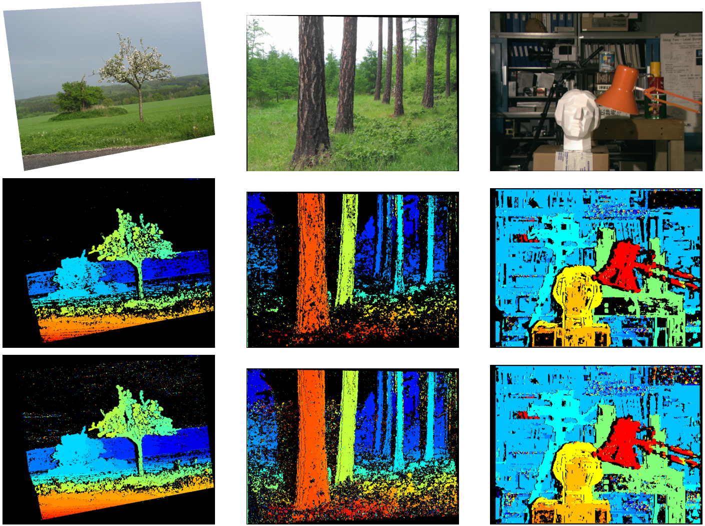
$L(D \mid p)$ – logarithm of matched-to-unmatched likelihood ratio (bigger is better)

why this way: we want to use maximum-likelihood on the entire T

3. (40) is max-cost matching (maximum assignment) for the maximum-likelihood (ML) matching problem
   - use the Hungarian (Munkres) algorithm and threshold the result with τ: $L(D \mid p) > \tau \geq 0$

or approximate the problem by sacrificing symmetry and accuracy to speed and use dynamic programming

Some Results for the Maximum-Likelihood (ML) Matching



left image

3% / 61%

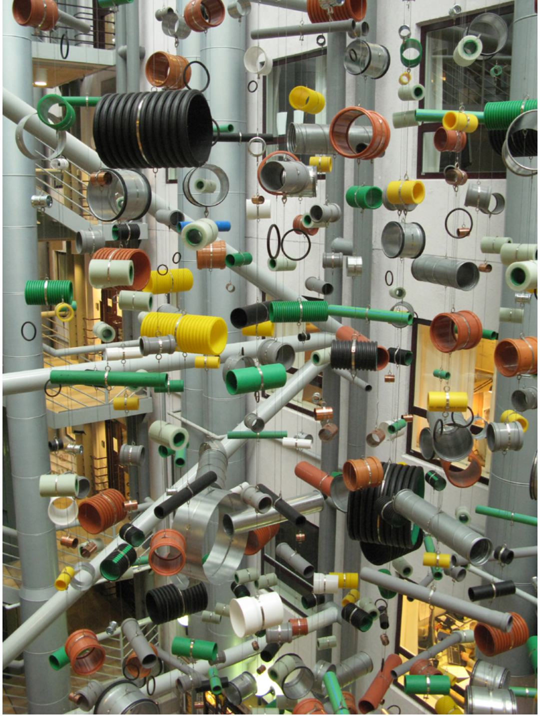
4.3% / 76%

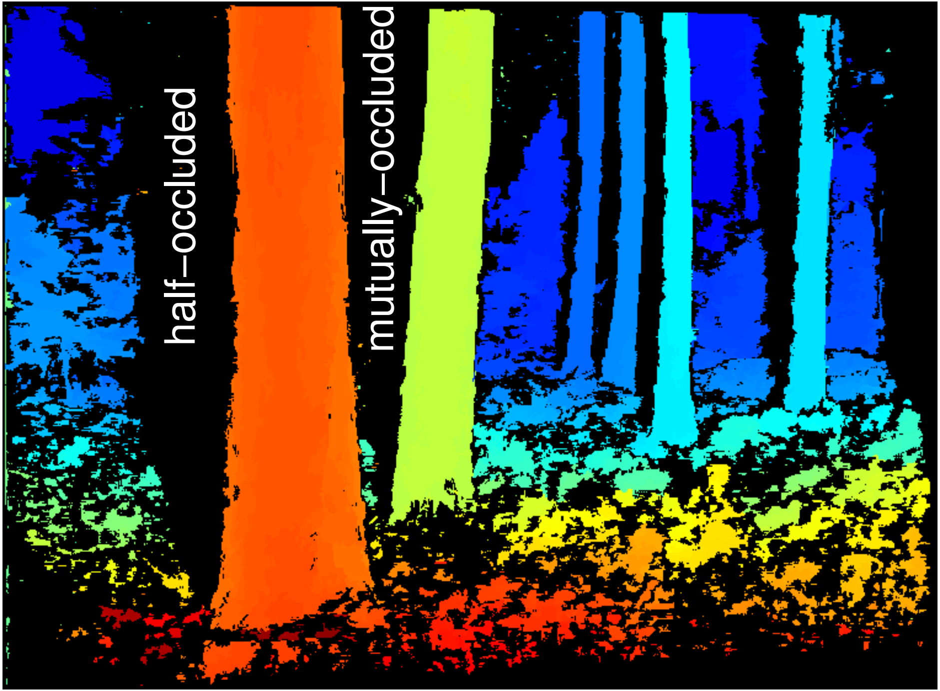
- unlike the WTA we can efficiently control the density/accuracy tradeoff with τ

black = no match

- middle row: threshold τ for $L(D \mid p)$ set to achieve error rate of 3% (and 61% density results)
- bottom row: threshold τ set to achieve density of 76% (and 4.3% error rate results)

Thank You




half–occluded
mutually–occluded