PS3 Review Session

Kuan Fang
CS231A
02/16/2018

Overview

Space carving

Single Object Recognition via SIFT

Histogram of Oriented Gradients (HOG)

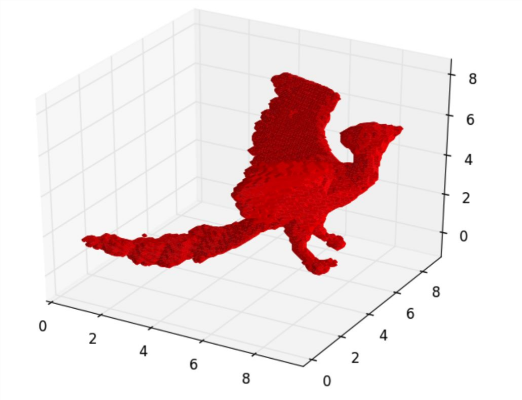
Space Carving

Objective:

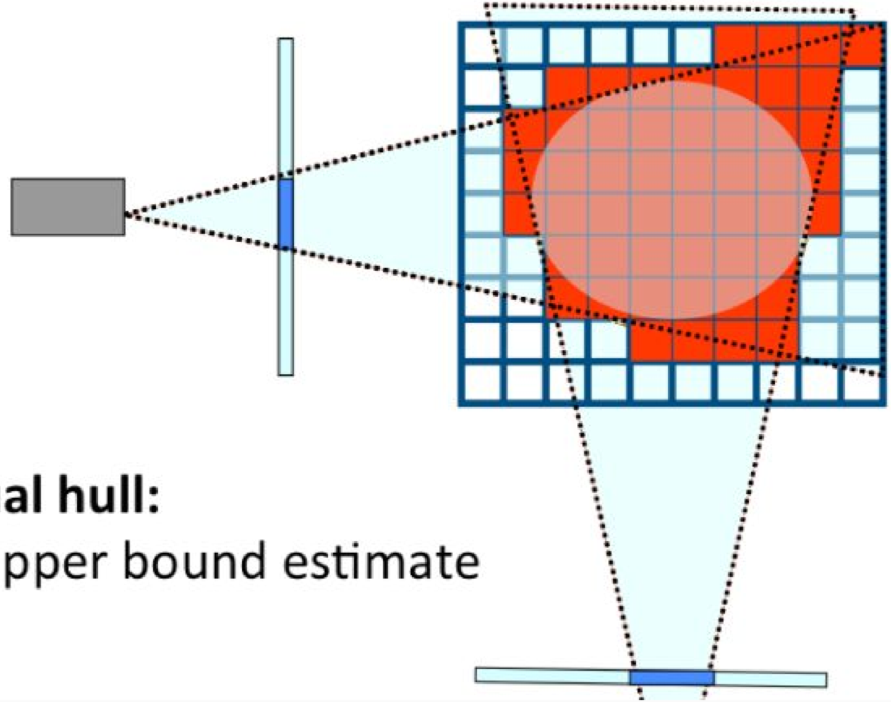
- Implement the process of space carving.

Lectures:

- Active Stereo & Volumetric Stereo

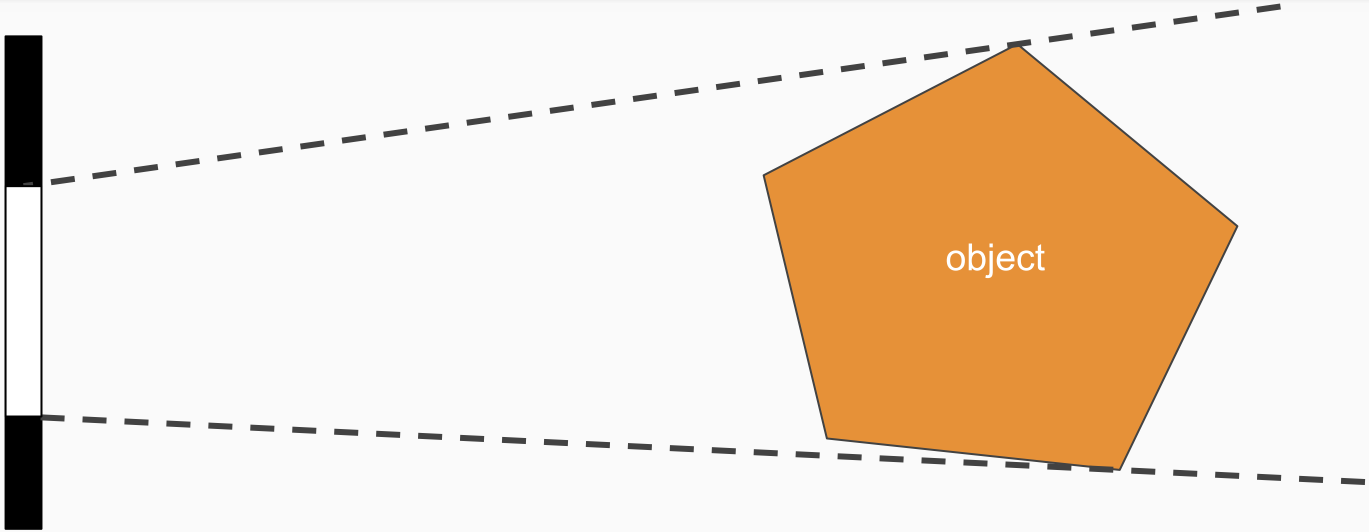


Review: Space Carving

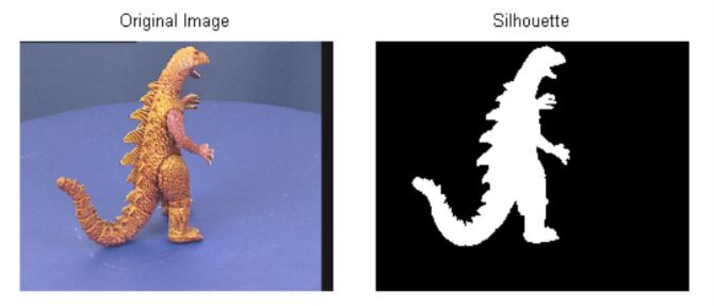


Visual hull:
an upper bound estimate

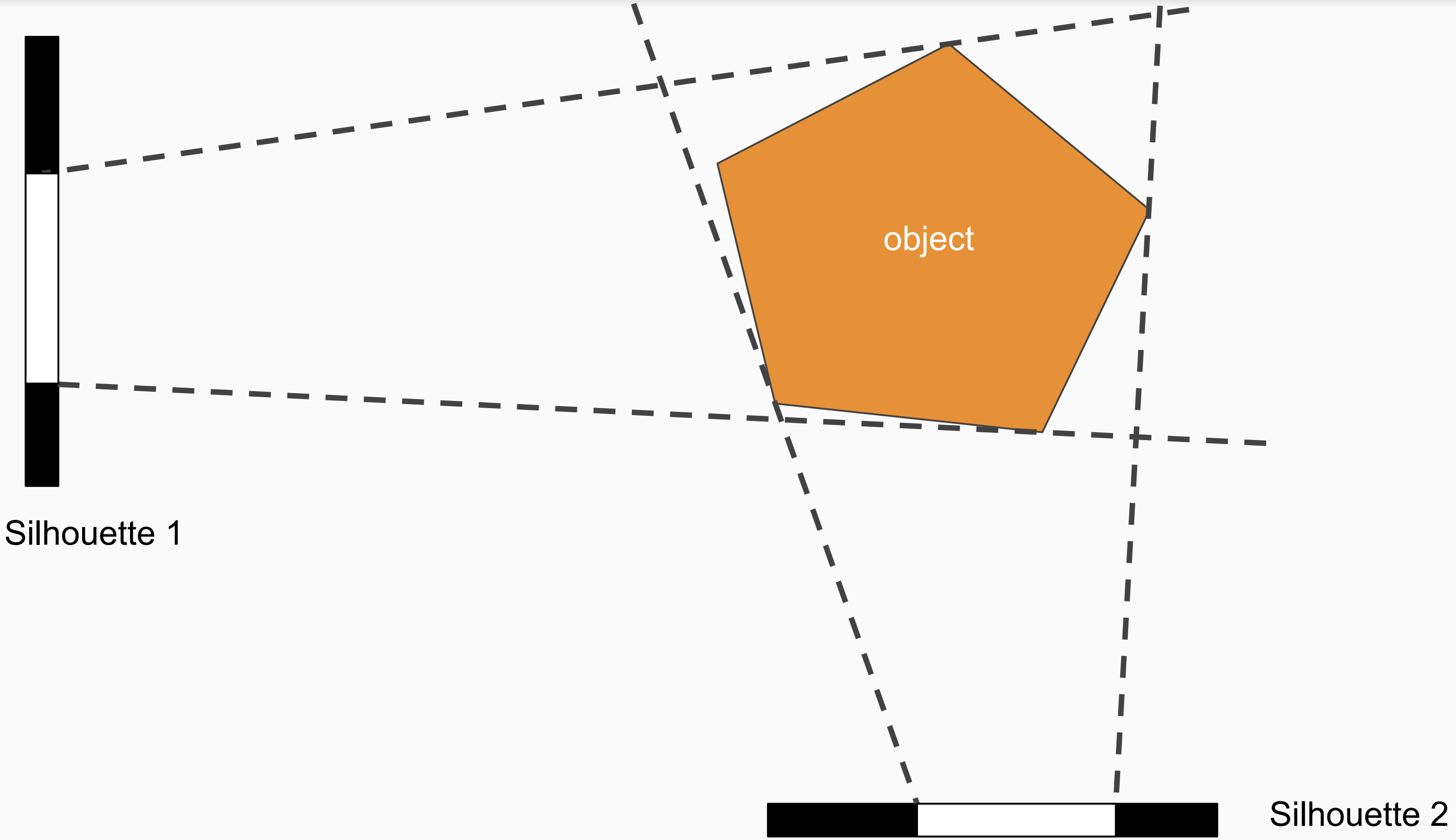
Review: Space Carving



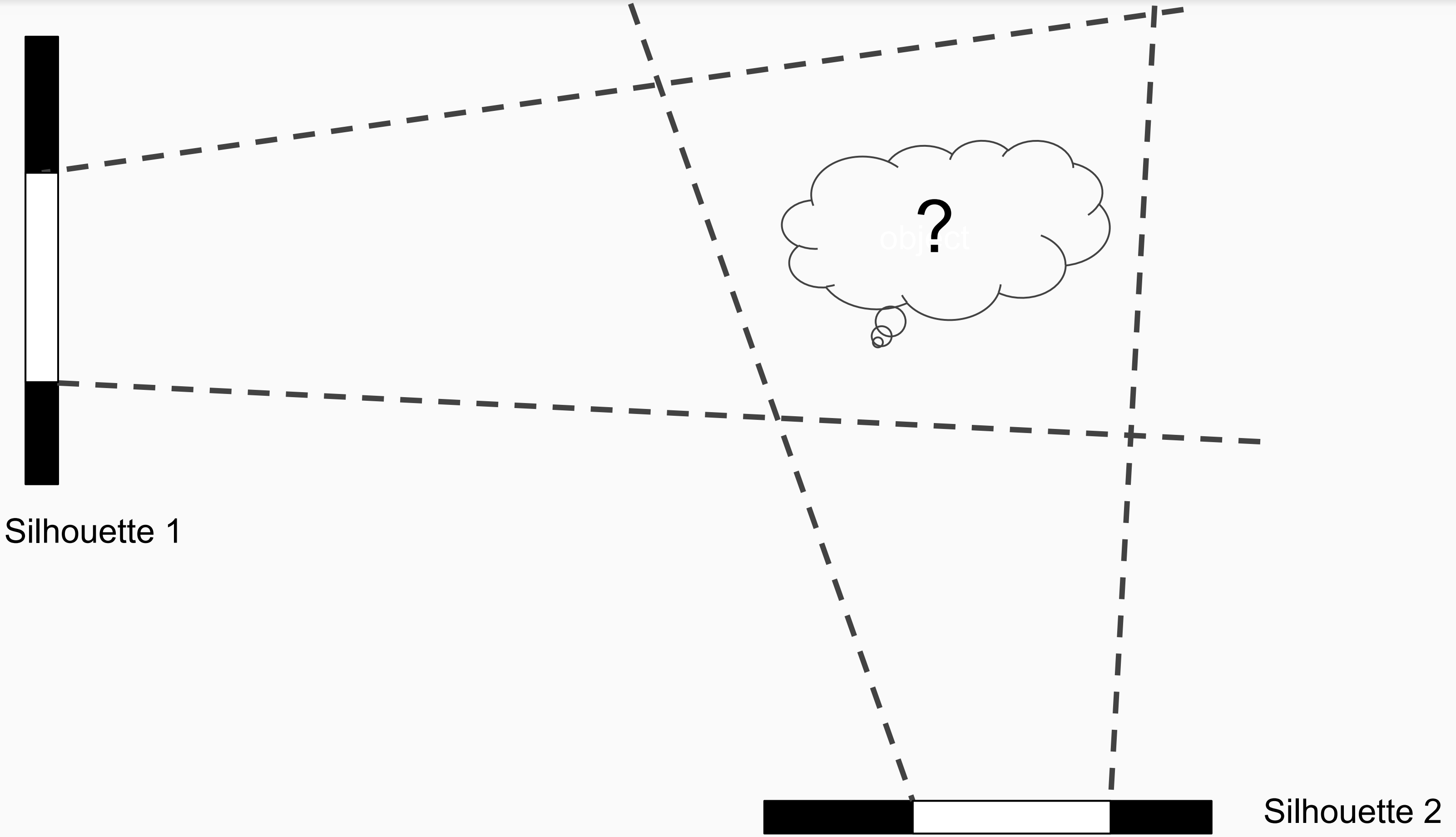
Silhouette 1



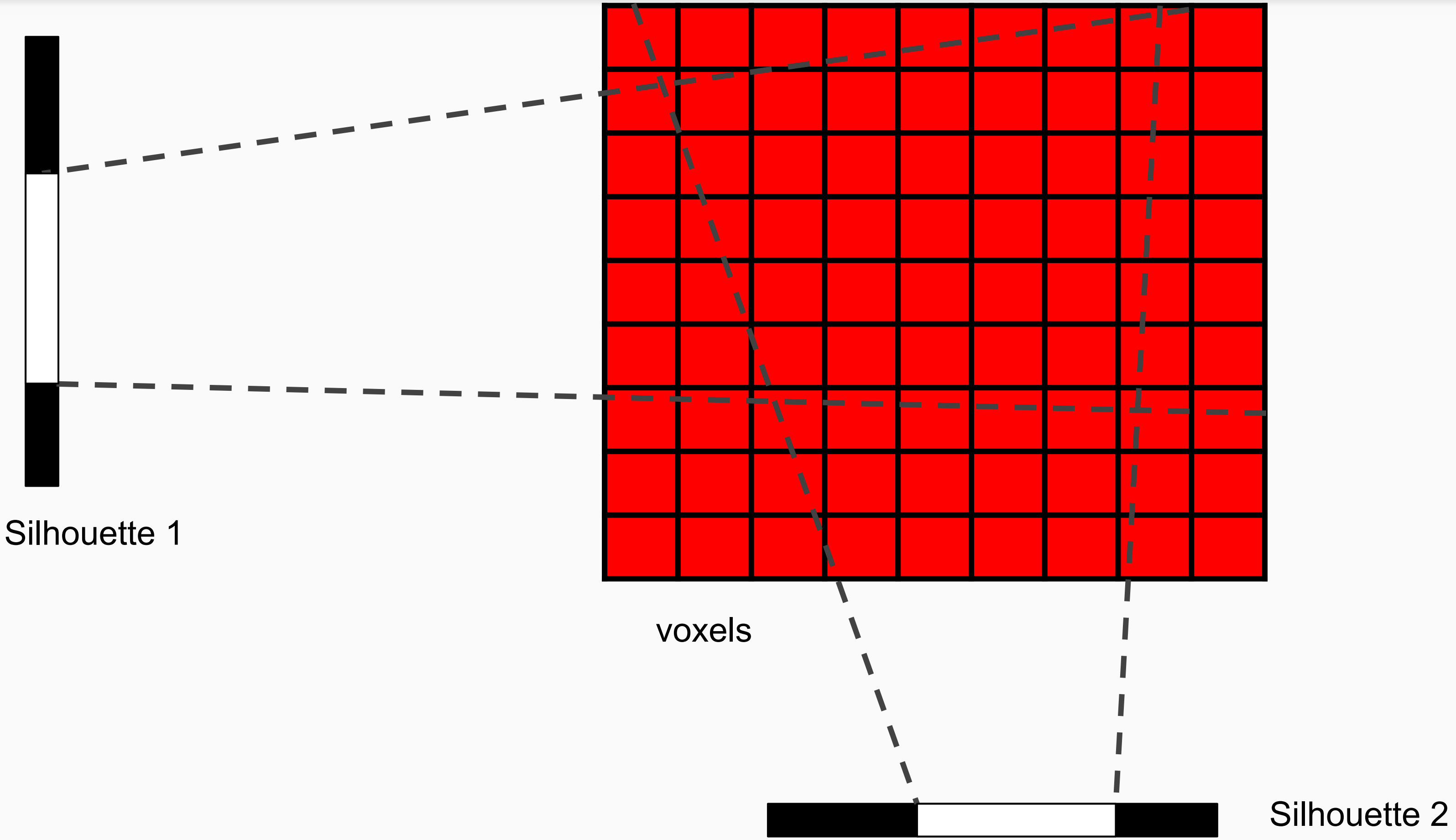
Review: Space Carving



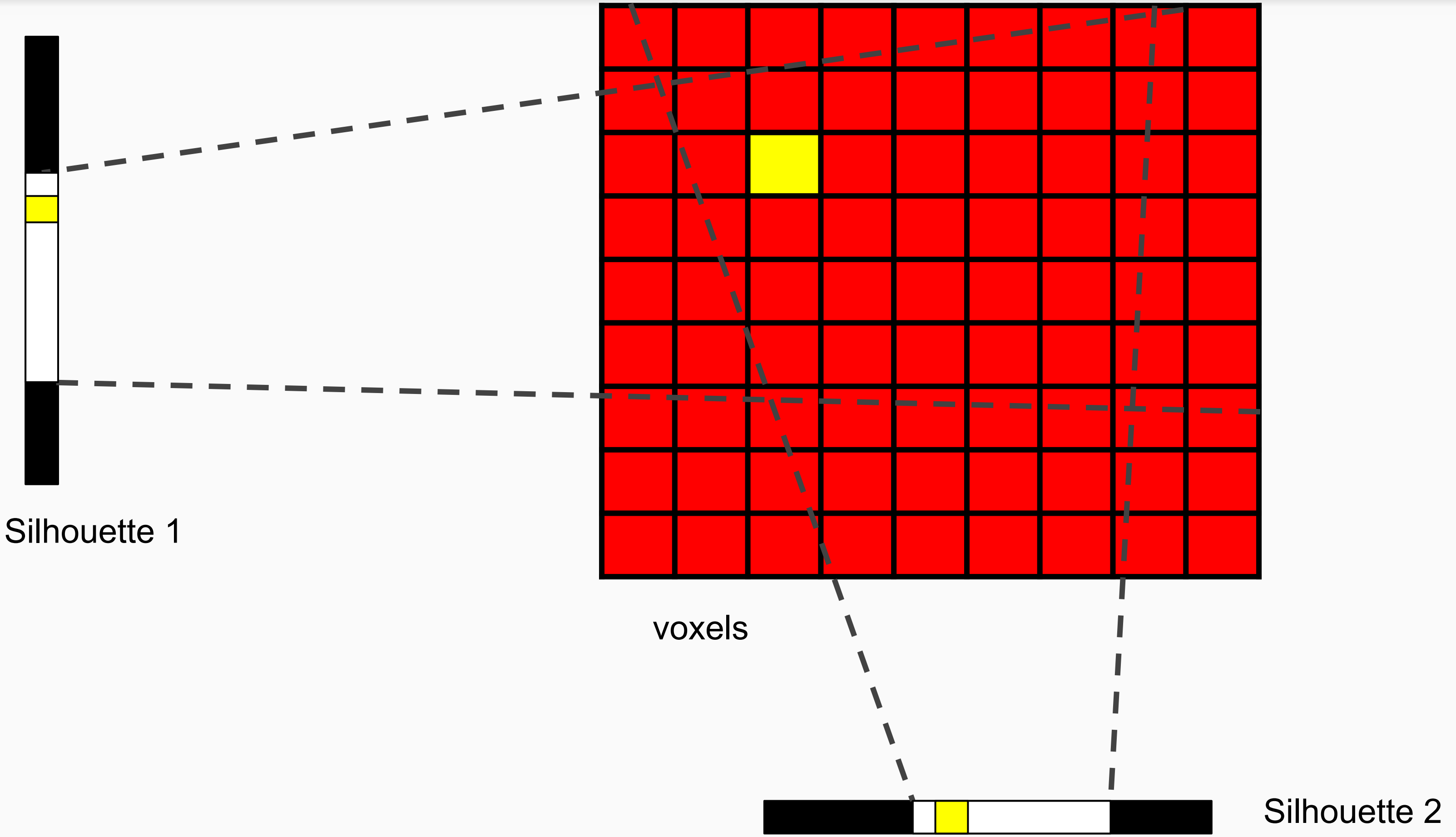
Goal of Space Carving



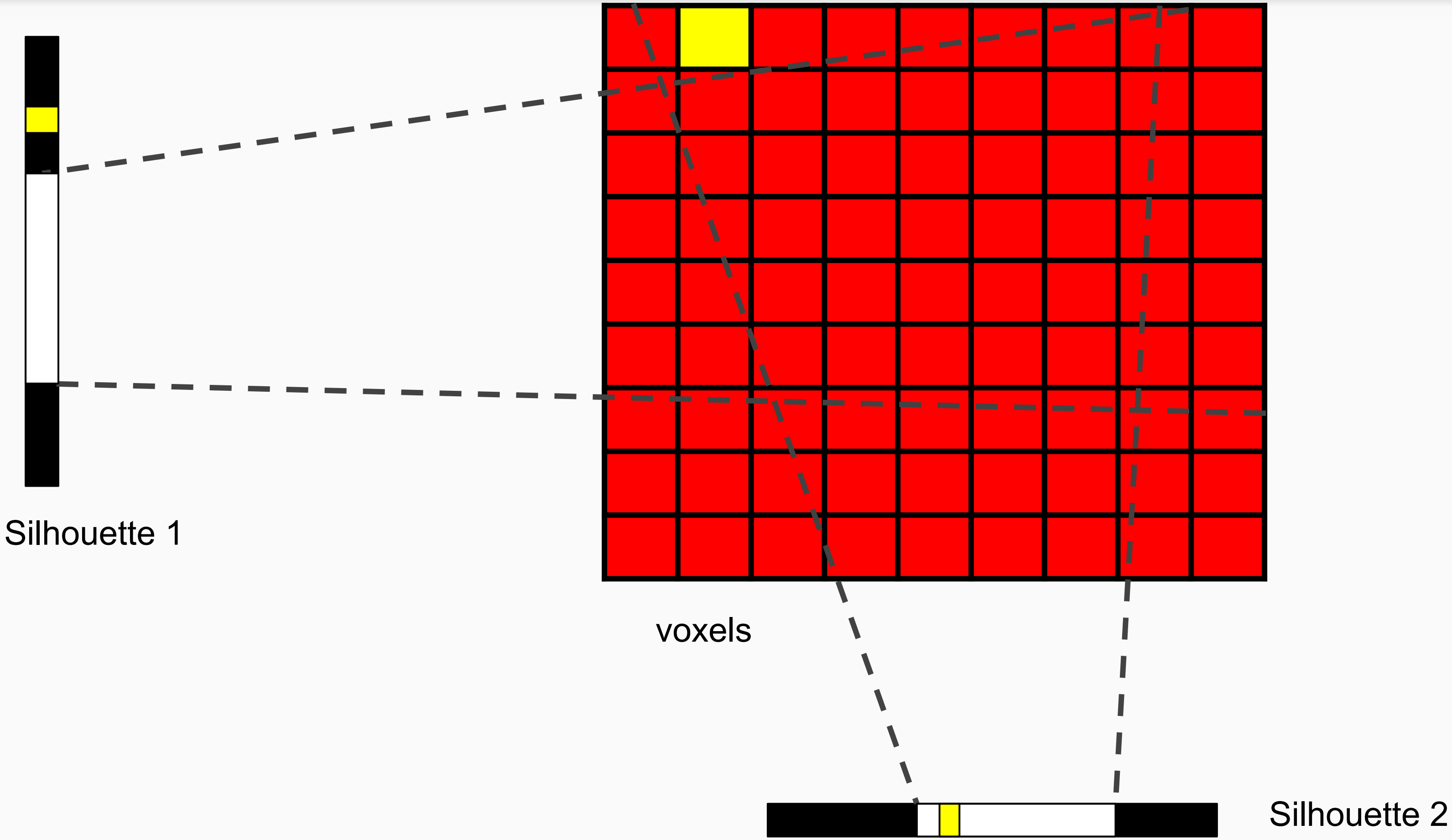
Review: Space Carving



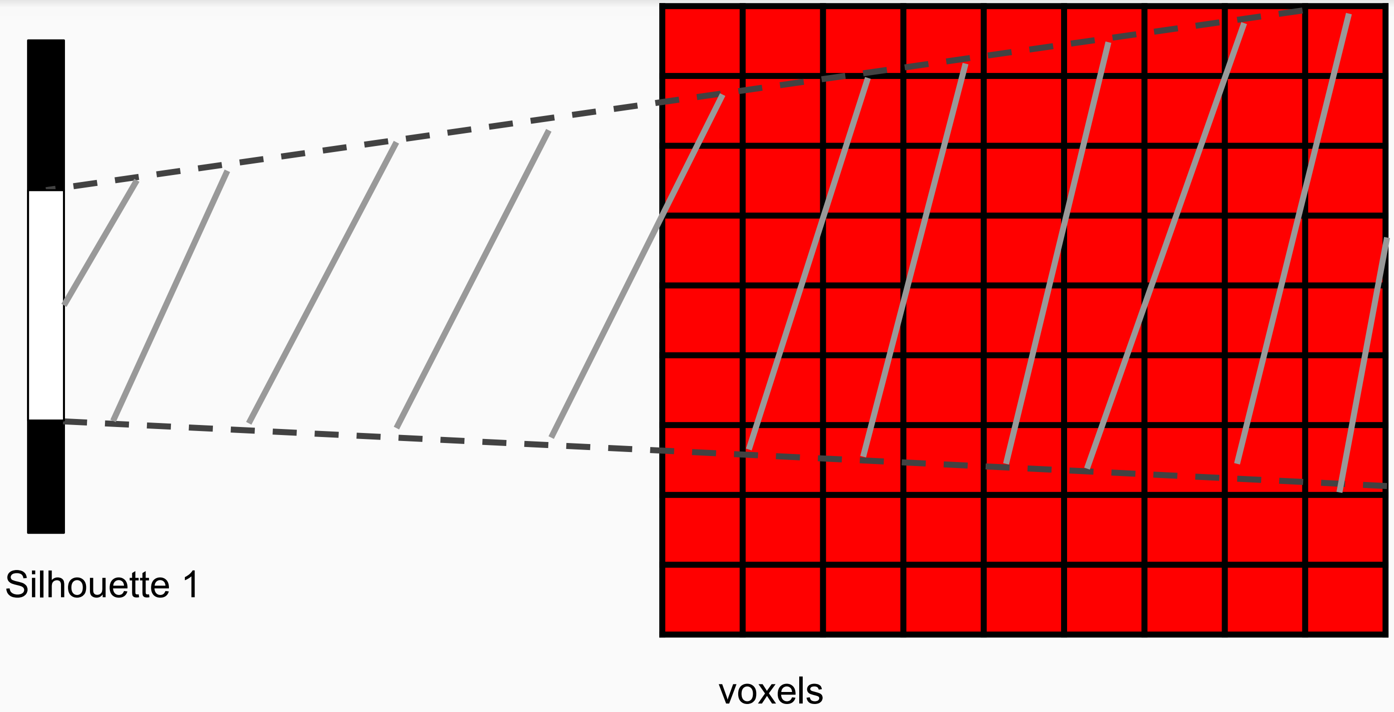
Review: Space Carving



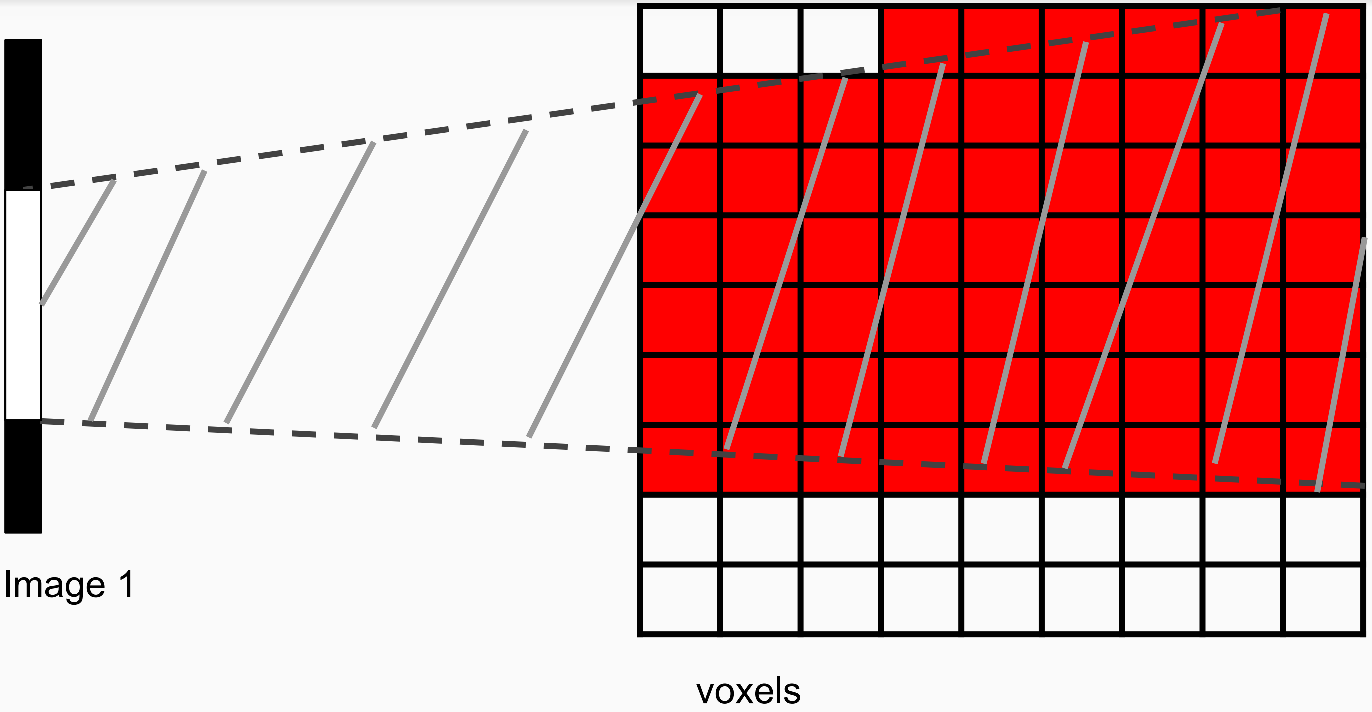
Review: Space Carving



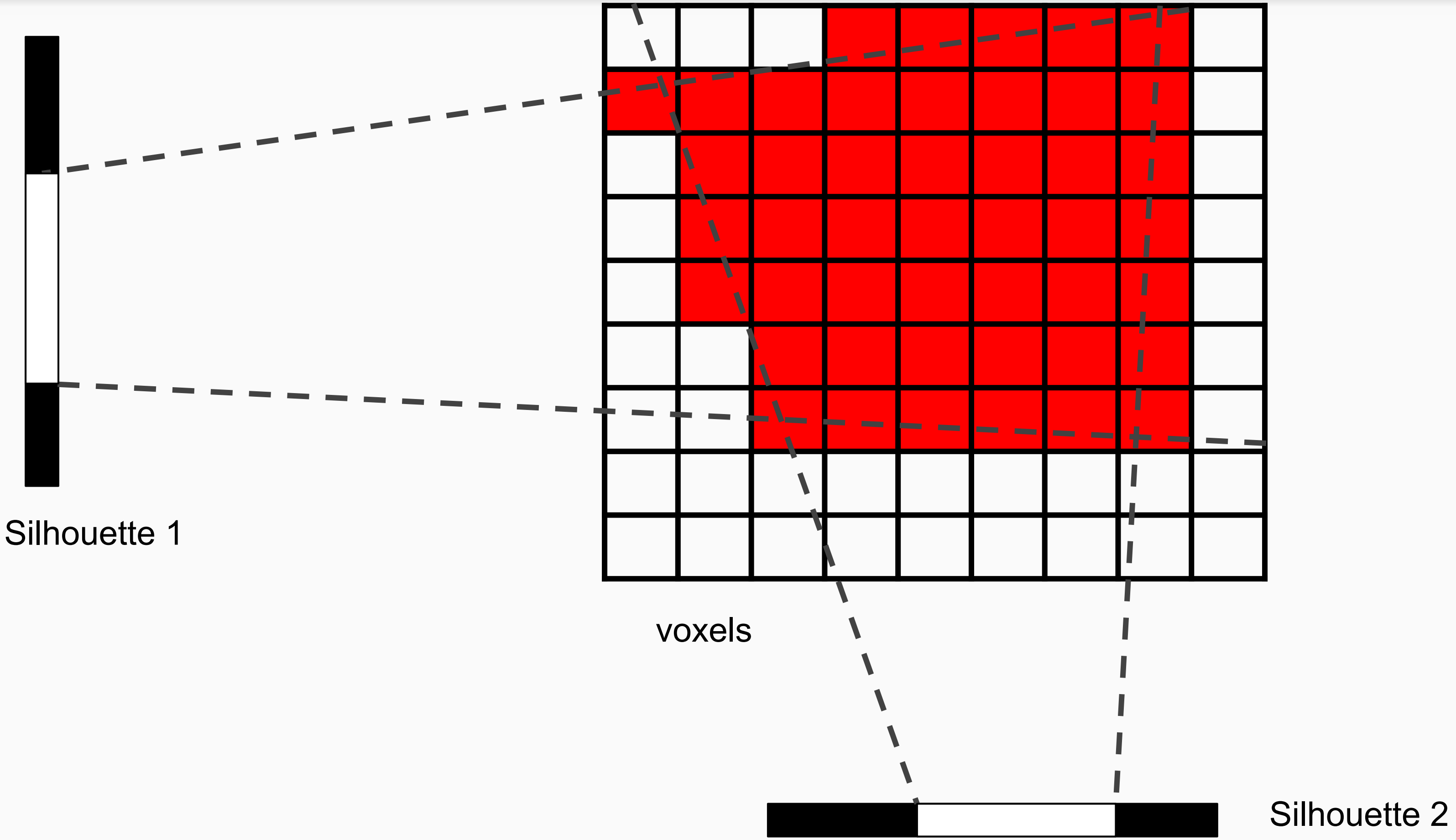
Review: Space Carving



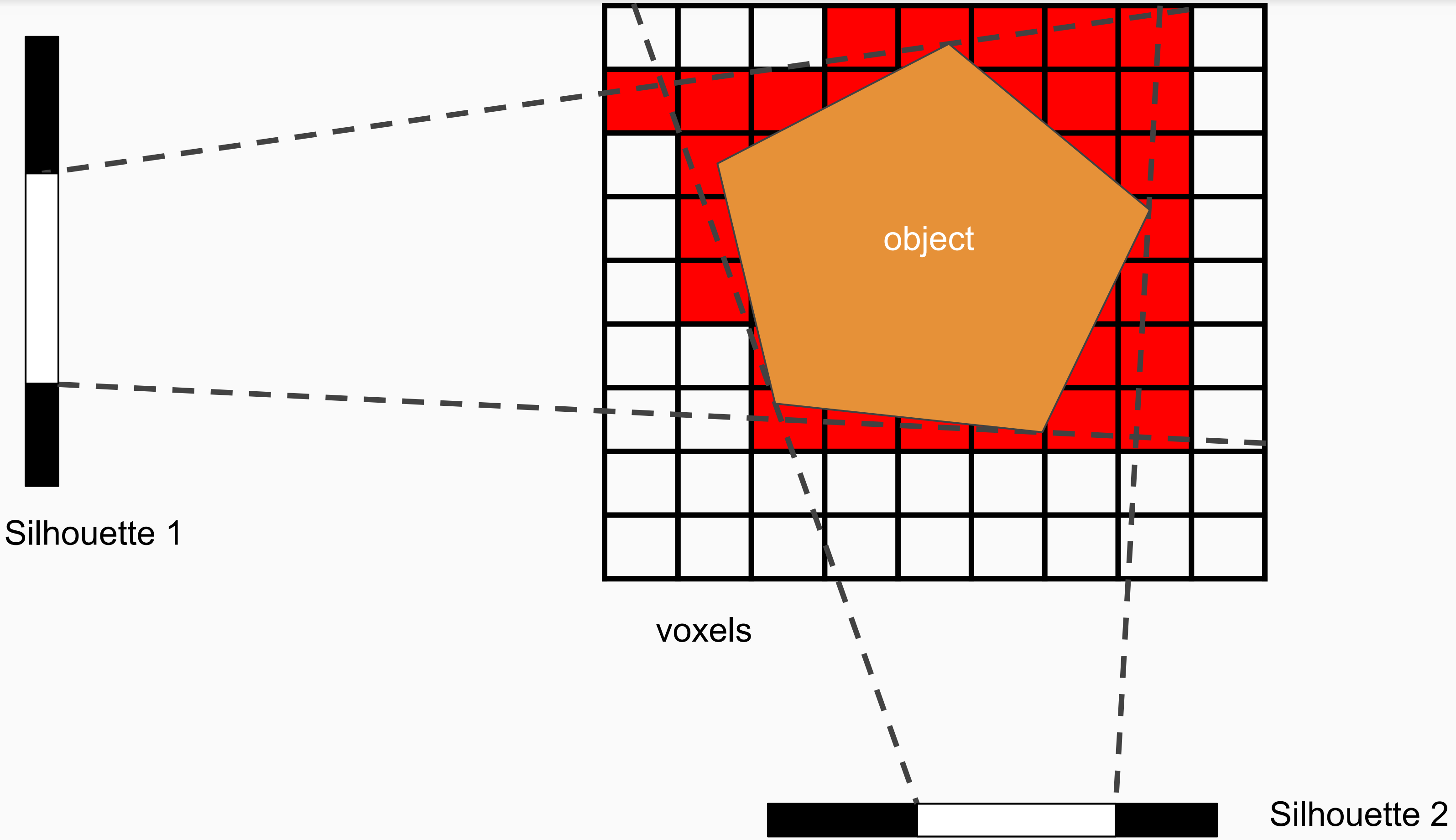
Review: Space Carving



Review: Space Carving



Review: Space Carving



Space carving - overview

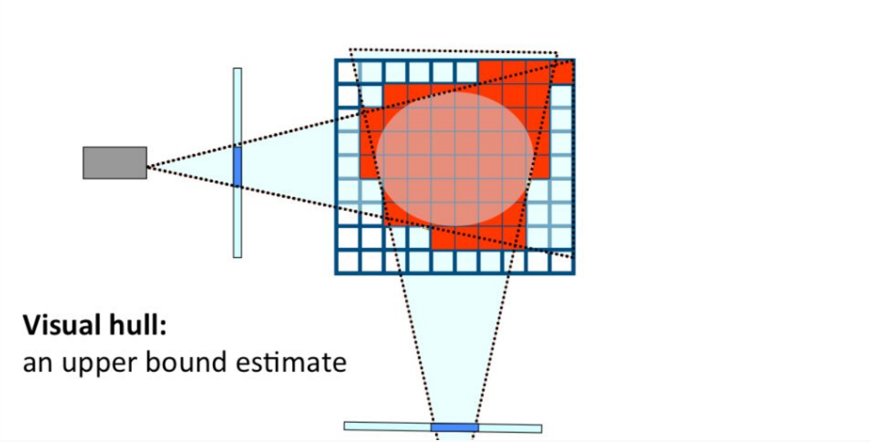
Steps:

- Estimate silhouettes of images (could be based on some heuristics, e.g. color)
- Form the initial voxels as a cuboid
- Iterate over cameras and remove the voxels which project to the dark part of each silhouette

Space carving - (a) (b) (c)

Steps:

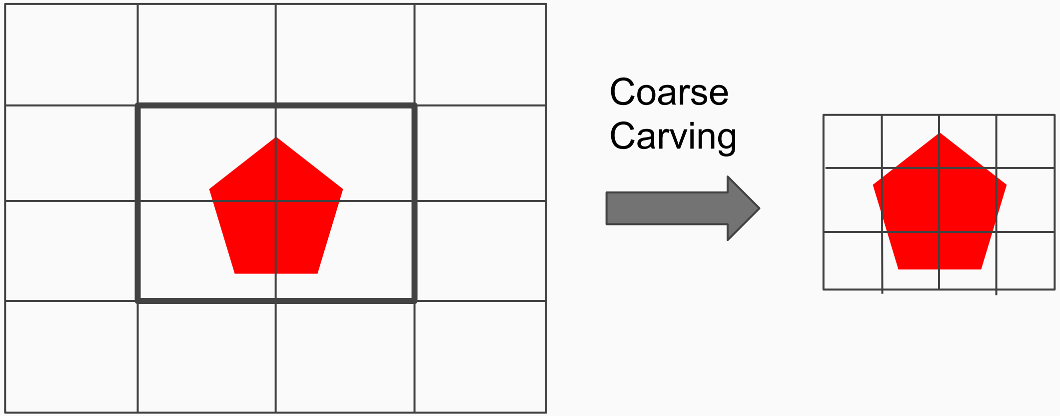
- Estimate silhouettes of images (could be based on some heuristics, e.g. color)
- Form the initial voxels as a cuboid
  - *You may find these functions useful: np.meshgrid, np.repeat, np.tile*
- Iterate over cameras and remove the voxels which project to the dark part of each silhouette
  - *Question: What will the voxels look like after the first, second, … iteration?*



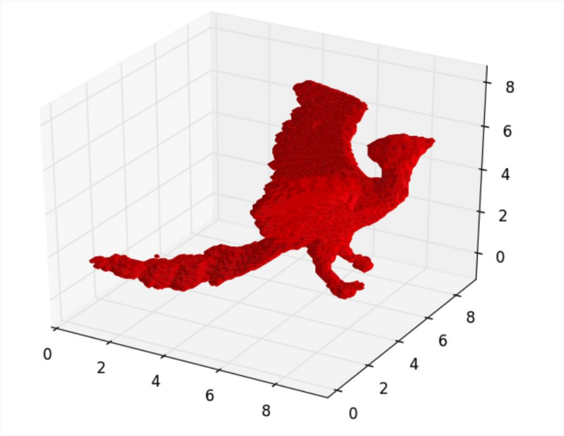
Space carving - (d)

Steps:

- Estimate silhouettes of images (could be based on some heuristics, e.g. color)
- Form the initial voxels as a cuboid
  - Question: What will the cuboid look like after each iteration?
- Iterate over cameras and remove the voxels which project to the dark part of each silhouette



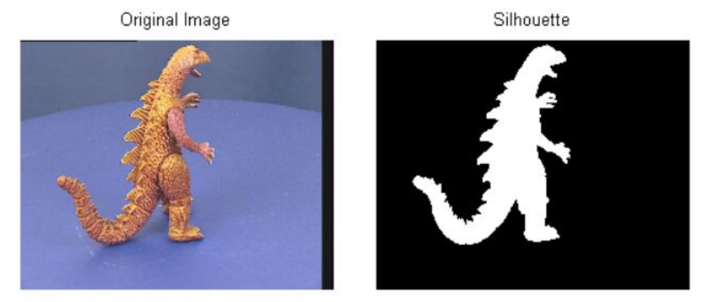
Final Output



Space carving - (e)

Steps:

- Estimate silhouettes of images (could be based on some heuristics, e.g. color)
  - Problem: The quality of silhouettes is not perfect.
  - The silhouette from each camera is not perfect, but the result is ok. Why?
  - Experiment: Use only a few of the silhouettes.
- Form the initial voxels as a cuboid
- Iterate over cameras and remove the voxels which project to the dark part of each silhouette



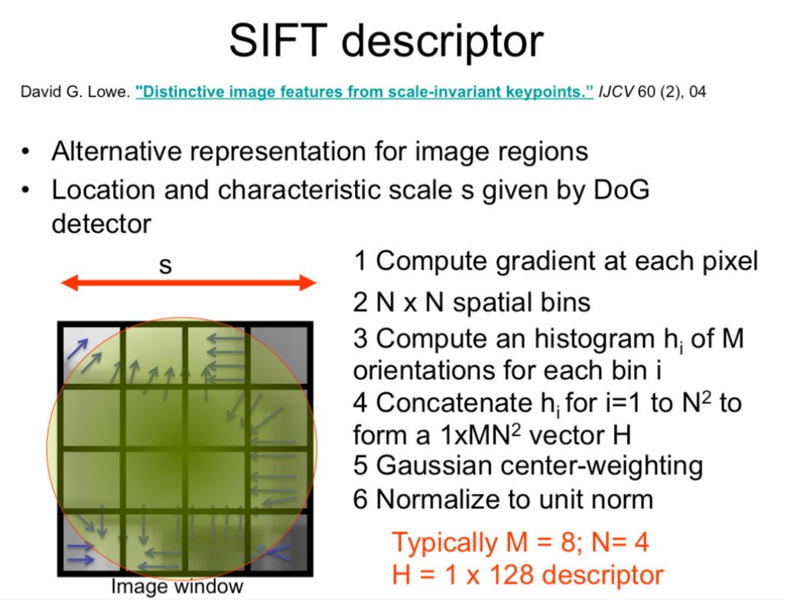
Single Object Recognition Via SIFT - overview

Objective:

- Understand how to use SIFT features for object recognition.
- Implement the RANSAC algorithm.
- Implement the Hough Transform algorithm.
- (Implementation of SIFT is not required.)

Lectures:

- Fitting and Matching
- Detectors and Descriptors

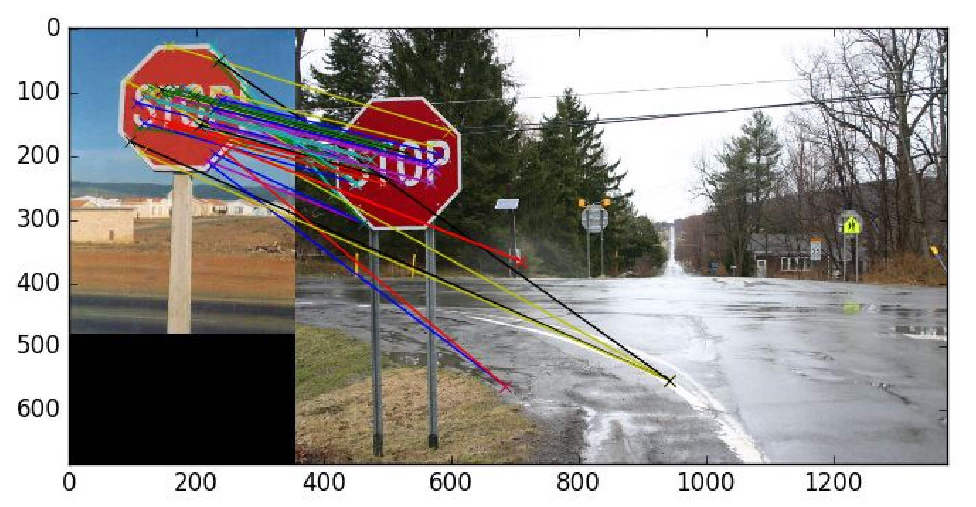


Single Object Recognition Via SIFT - (a)

We've implemented SIFT descriptor for you, and your task is using it for object recognition.

Read: **Section 7. Application to object recognition** in Lowe's SIFT paper
http://www.cs.ubc.ca/~lowe/papers/ijcv04.pdf

Be sure to understand what does the threshold mean and how to use it.

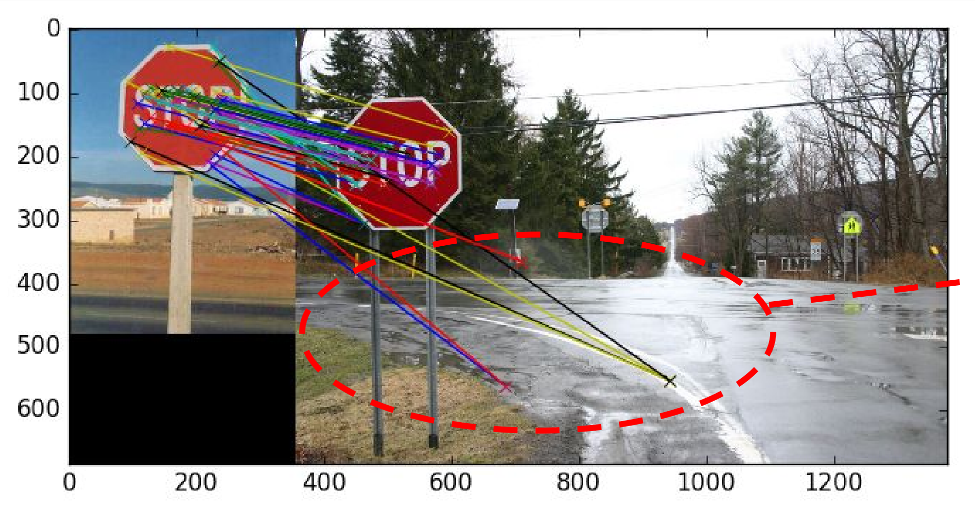


Single Object Recognition Via SIFT - (b)

RANSAC to refine matching

Basic idea: use RANSAC to fit a homography matrix *H* between two set of key points. Only keep the inliers for matching, remove the outliers.

Question: how many pairs of correspondences do we need in each iteration?



Should be removed

Single Object Recognition Via SIFT - (c)

Theoretical properties about RANSAC: see lecture notes / slides

Understand the relations between:

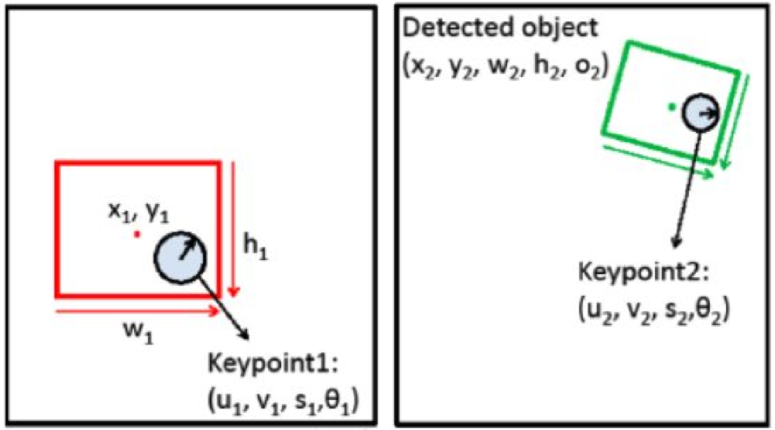
- N: number of samples
- e: outlier ratio
- s: minimum number of data points to fit the model
- p: probability guaranteed

Single Object Recognition Via SIFT - (d) (e)

Find quantitative relation between two bounding boxes

Input (u1, v1, s1, theta1; u2, v2, s2, theta2; x1, y1, w1, h1) Output (x2, y2, w2, h2, o2)

Keypoint1 Keypoint2 Bbx1 Bbx2 & Orientation



(a) Image 1 (b) Image 2

Histogram of Oriented Gradients (HOG) - Overview

Objective:

- Implement HOG features.

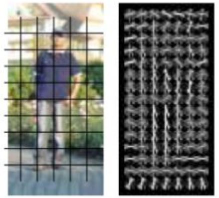
Lectures:

- Detectors and Descriptors

HoG = Histogram of Oriented Gradients

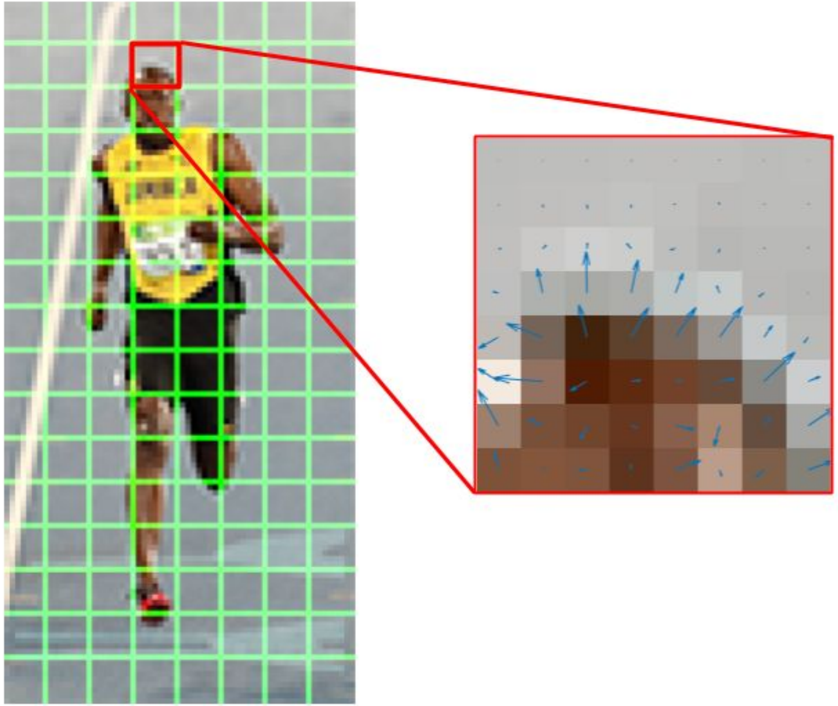
Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05

- Like SIFT, but…
  - Sampled on a dense, regular grid around the object
  - Gradients are contrast normalized in overlapping blocks



Histogram of Oriented Gradients (HOG) - Online Reference

https://www.learnopencv.com/histogram-of-oriented-gradients/



2	3	4	4	3	4	2	2
5	11	17	13	7	9	3	4
11	21	23	27	22	17	4	6
23	99	165	135	85	32	26	2
91	155	133	136	144	152	57	28
98	196	76	38	26	60	170	51
165	60	60	27	77	85	43	136
71	13	34	23	108	27	48	110

Gradient Magnitude

80	36	5	10	0	64	90	73
37	9	9	179	78	27	169	166
87	136	173	39	102	163	152	176
76	13	1	168	159	22	125	143
120	70	14	150	145	144	145	143
58	86	119	98	100	101	133	113
30	65	157	75	78	165	145	124
11	170	91	4	110	17	133	110

Gradient Direction

Histogram of Oriented Gradients (HOG) - Overview

Steps:

- Preprocessing the image.
- Calculate the gradient image.
- Calculate the gradient histogram of each cell.
- Normalize each block.
- Calculate the HOG feature vector.

Histogram of Oriented Gradients (HOG) - (a)

Steps:

- Preprocessing the image.
- Calculate the gradient image.
- Calculate the gradient histogram of each cell.
- Normalize each block.
- Calculate the HOG feature vector.

Histogram of Oriented Gradients (HOG) - (a)

Compute the gradients of image (angles, magnitudes)

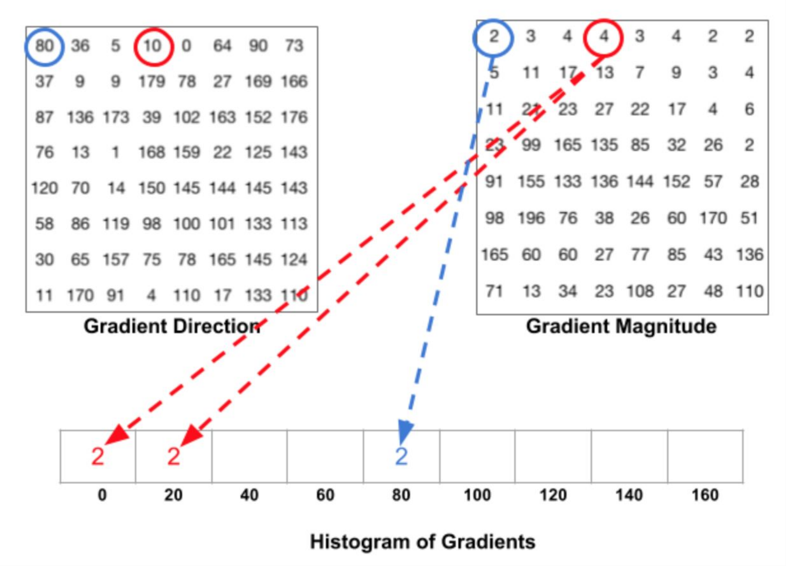
```
The way that the angles and magnitude per pixel are computed as follows:
Given the following pixel grid

    P1 P2 P3
    P4 P5 P6
    P7 P8 P9

We compute the angle on P5 as arctan(dy/dx) = arctan(P2-P8 / P4-P6).
```

```
The magnitude is simply sqrt((P4-P6)^2 + (P2-P8)^2)
```

Histogram of Oriented Gradients (HOG) - (b)

Steps:

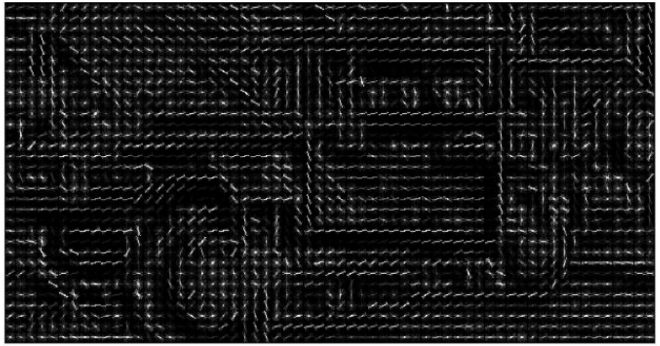
- Preprocessing the image.
- Calculate the gradient image.
- Calculate the gradient histogram of each cell.
    - The output shape is (nbins,).
- Normalize each block.
- Calculate the HOG feature vector.



Histogram of Oriented Gradients (HOG) - (c)

Steps:

- Preprocessing the image.
- Calculate the gradient image.
- Calculate the gradient histogram of each cell.
- Normalize each block.
  - We use a stride of 50% of the block size.
- Calculate the HOG feature vector.



Thank You