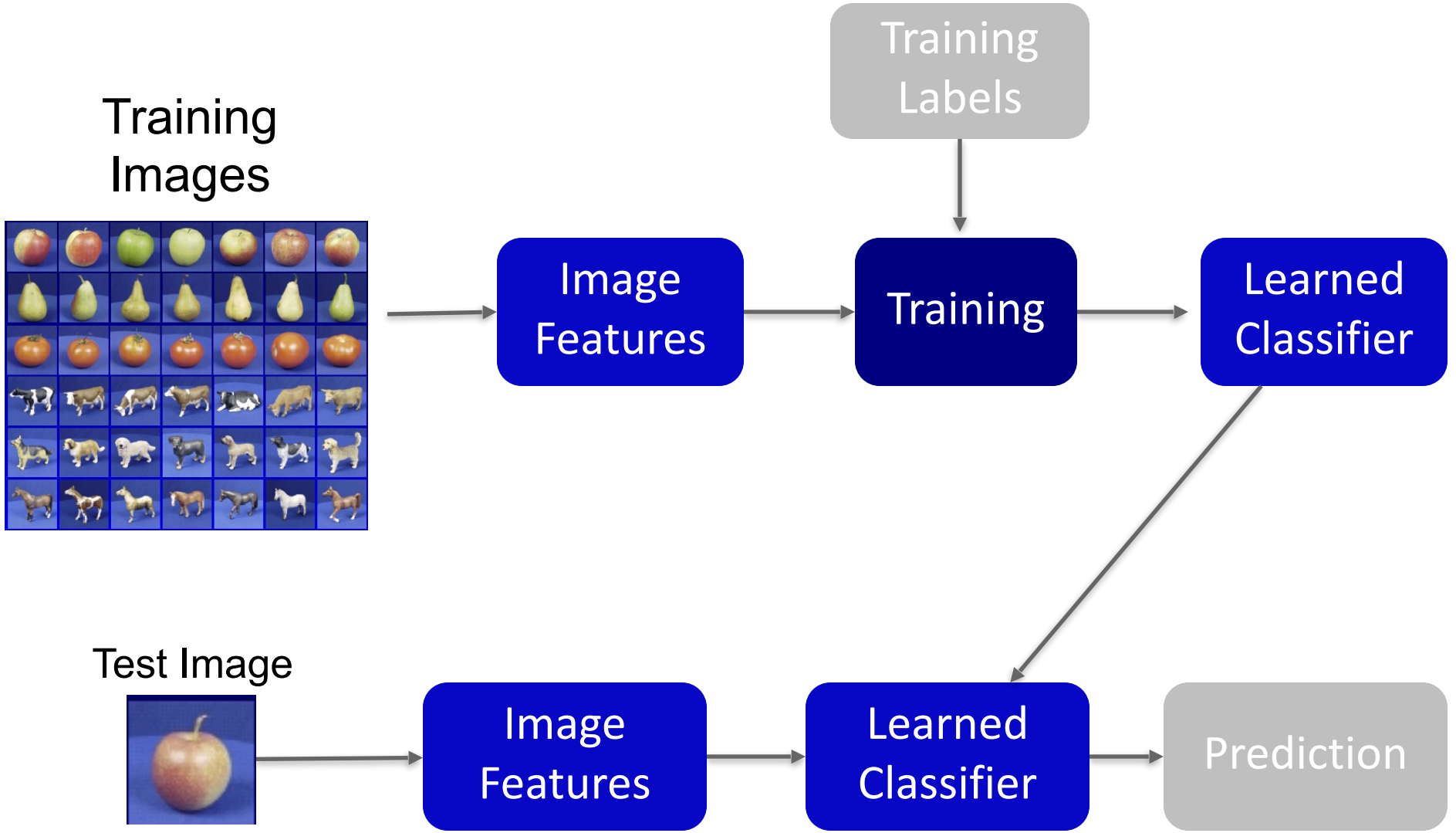
Lecture 17

Object Detection

So far: A simple recognition pipeline



Today’s agenda

- Object detection
  - Task and evaluation
- A simple detector
- Deformable parts model

Today’s agenda

- Object detection
  - Task and evaluation
- A simple detector
- Deformable parts model

Object Detection

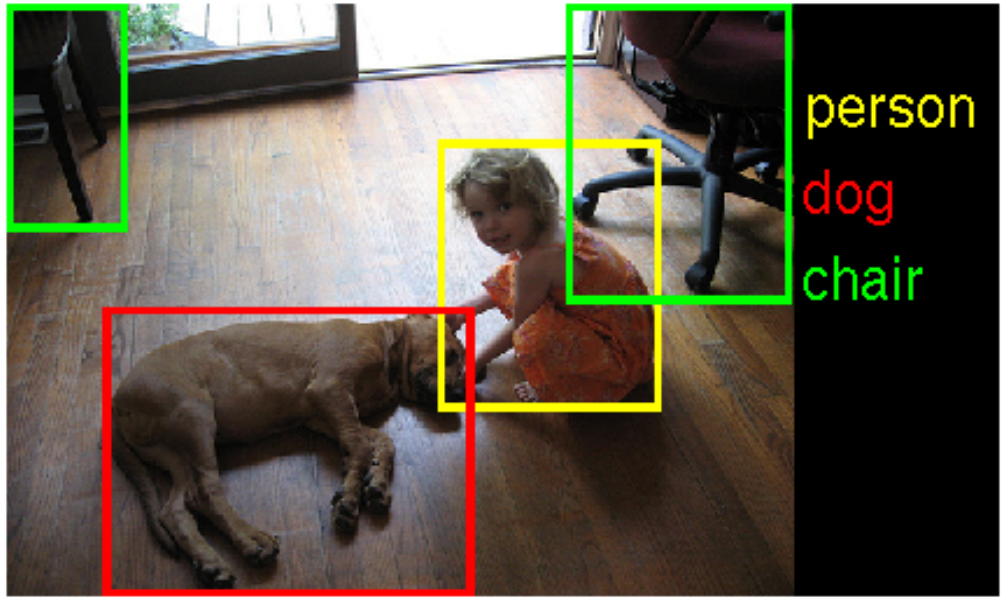
Credit: Flickr user neilalderney123

- What do you see in the image?

Object Detection

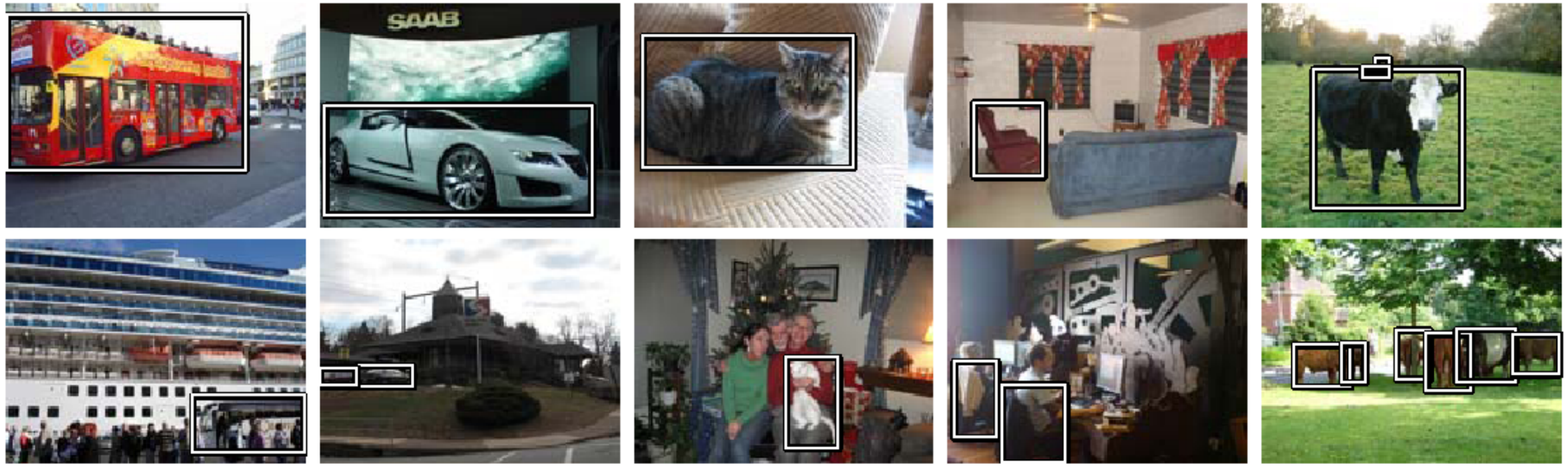
- **Problem**: Detecting and localizing objects from various categories, such as cars, people, etc.

- Challenges:
  - Illumination,
  - viewpoint,
  - deformations,
  - Intra-class variability



Object Detection Benchmarks

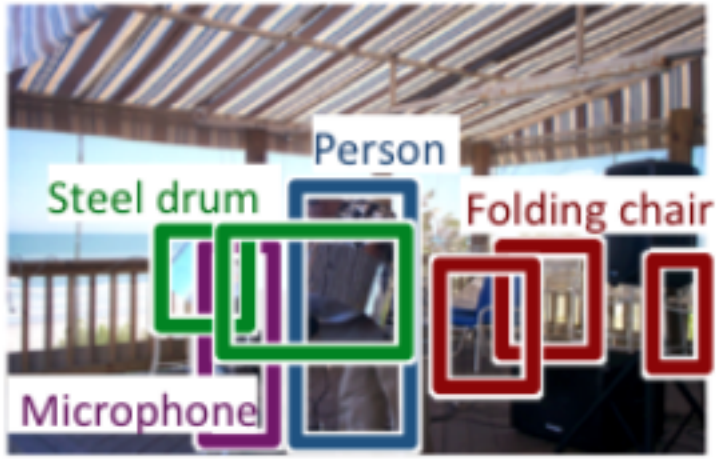
- PASCAL VOC Challenge



- 20 categories
- Annual classification, detection, segmentation, … challenges

Object Detection Benchmarks

- PASCAL VOC Challenge
- ImageNet Large Scale Visual Recognition Challenge (ILSVRC)
  - 200 Categories for detection



Object Detection Benchmarks

- PASCAL VOC Challenge
- ImageNet Large Scale Visual Recognition Challenge (ILSVR)
- Common Objects in Context (COCO)
  - 80 Object categories

How do we evaluate object detection?

predictions

ground truth

Defining what is a good versus bad detection

IoU is a metric used to decide good from bad predictions.

Given a predicted box and and ground truth box:

IoU = intersection between the two boxes over (divided by) the union of the two

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$

Defining what is a good versus bad detection

We say a prediction was good if it has IoU > 0.5 with any of the ground truth boxes

0.5 is a threshold that is generally accepted as a good heuristic.

How do we evaluate object detection?

predictions

ground truth

True positive:
- The overlap of the prediction with the ground truth is MORE than 0.5

How do we evaluate object detection?

predictions

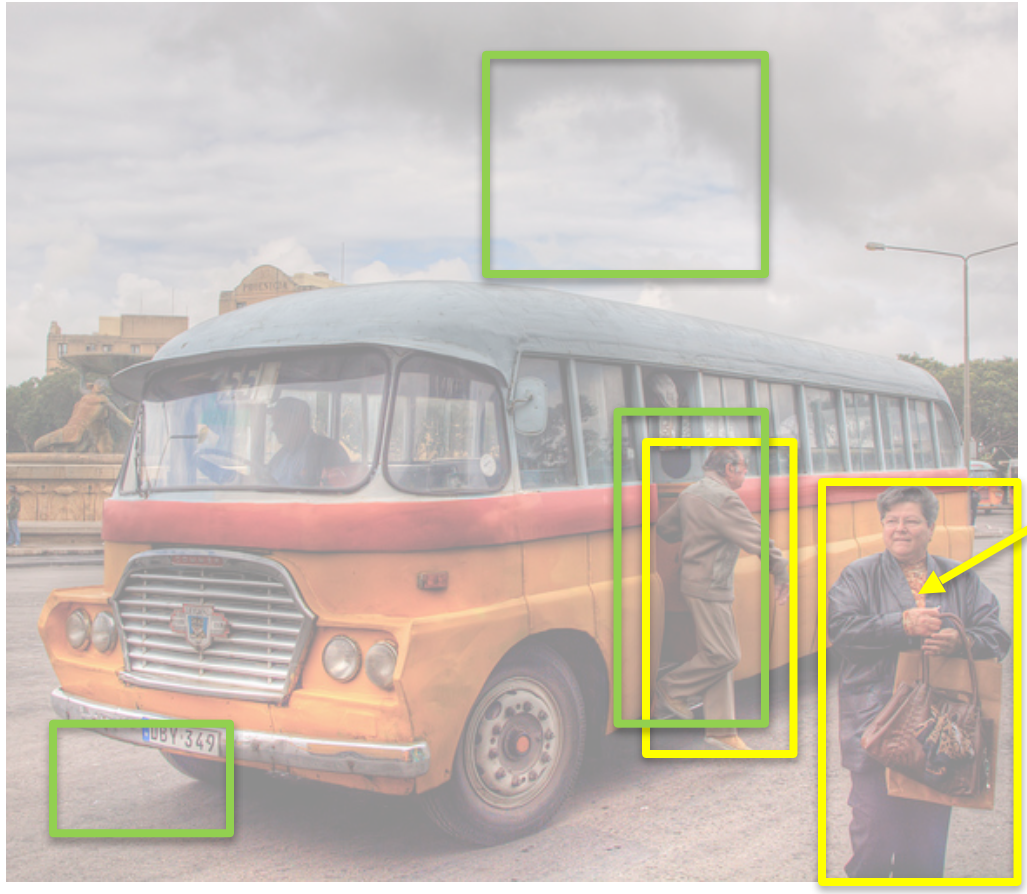
ground truth

True positive:

False positive:

- The overlap of the prediction with the ground truth is LESS than 0.5

How do we evaluate object detection?



predictions

ground truth

True positive:

False positive:

False negative:

- The objects that our model doesn't find

	Predicted 1	Predicted 0
True 1	true positive	false negative
True 0	false positive	true negative

- Precision:
  - how many of the predicted detections are correct?

$$precision = \frac{TP}{TP + FP}$$

- Recall:
  - how many of the ground truth objects are detected?

$$recall = \frac{TP}{TP + FN}$$

How do we evaluate object detection?

predictions

ground truth

True positive: 1
False positive: 2
False negative: 1

Q. What is the precision?

How do we evaluate object detection?

predictions

ground truth

True positive: 1
False positive: 2
False negative: 1

Q. What is the precision? 1/3

How do we evaluate object detection?

predictions

ground truth

True positive: 1
False positive: 2
False negative: 1

Q. What is the precision? 1/3

Q. What is the recall?

How do we evaluate object detection?

predictions

ground truth

True positive: 1
False positive: 2
False negative: 1

Q. What is the precision? 1/3

Q. What is the recall? 1/2

Today’s agenda

- Spatial pyramids
- Object detection
  - Task and evaluation
- A simple detector
- Deformable parts model

Dalal-Triggs method

Sliding window (Convolution)

At every patch as the window slides

1. Convert the image patch into your favorite feature representation
    a. For example:
        i. HoG,
        ii. HoG with PCA,
        iii. Bag of words on RGB
        iv. etc.
2. Use a trained classifier to determine if it is a specific class
    a. e.g. kNN classifier
3. Accumulate the predictions over all the patches

Sliding window + hog features

No person here

- Slide through the image and check if there is an object at every location

Sliding window + hog features

- Slide through the image and check if there is an object at every location

YES!! Person match found

Sliding window + hog features

No bus found

- But what if we were looking for buses?

Sliding window + hog features

No bus found

- We will never find the object if we don’t choose our window size wisely!

Sliding window + hog features

- We need to do multi-scale sliding windows with pyramids

Computationally, we first resize the image to different sizes and then extract features at each size.

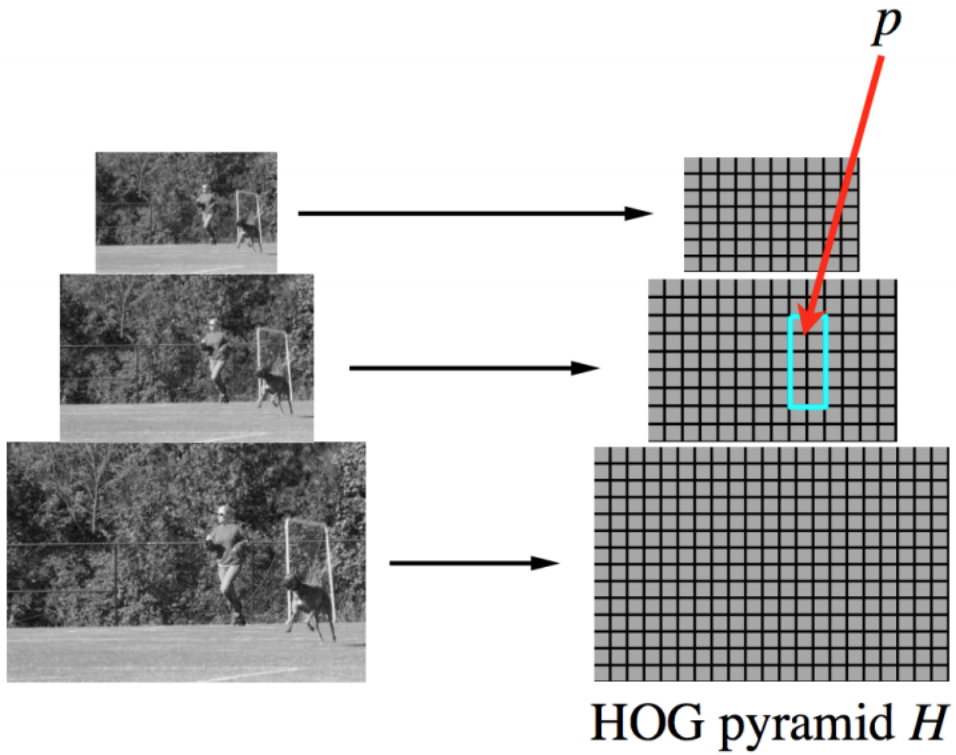


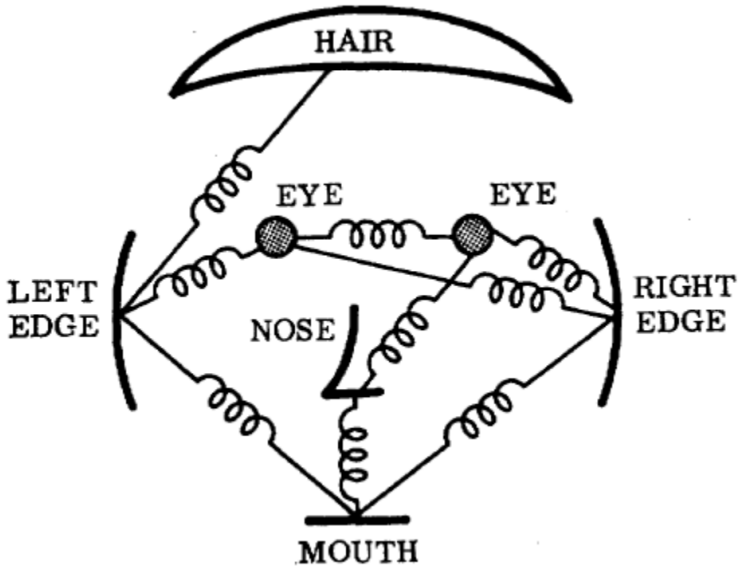
Image Pyramid:
An important idea even as of today!!

Today’s agenda

- Spatial pyramids
- Object detection
  - Task and evaluation
- A simple detector
- Deformable parts model

Deformable Parts Model

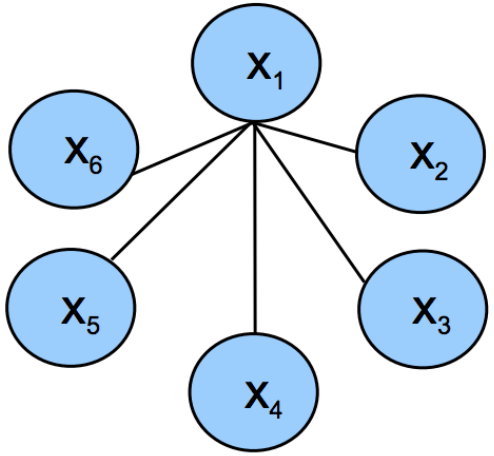
- Represents an object as a "collection of parts"
- Each part represents local appearances
- Make prediction jointly



Fischler and Elschlager, Pictoral Structures, 1973

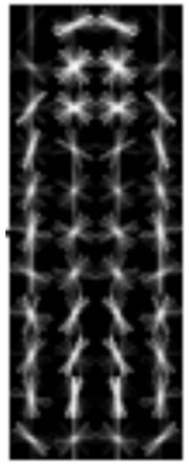
Detecting a person with their parts

- Star model: every part is defined relative to a root.
- Example: a person can be modelled as having a head, left arm, right arm, etc.
- All parts can be modelled relative to the global person detector, which acts as the root.



Deformable parts model

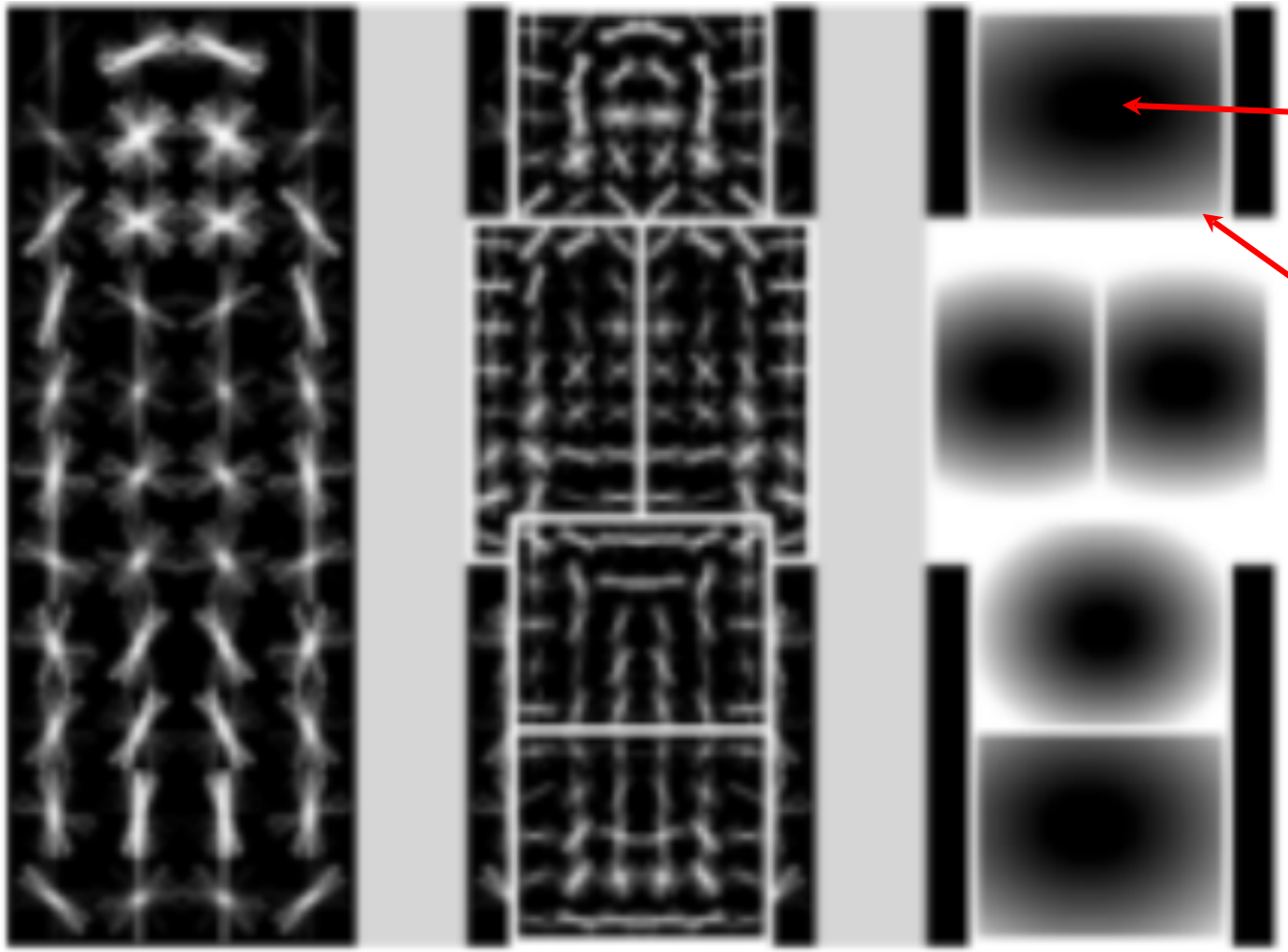
- Each model will have a global model. And a set of part models. Here is an example of a global person HoG filter with it's 'head' part filter:



Global/root filter

Part filter

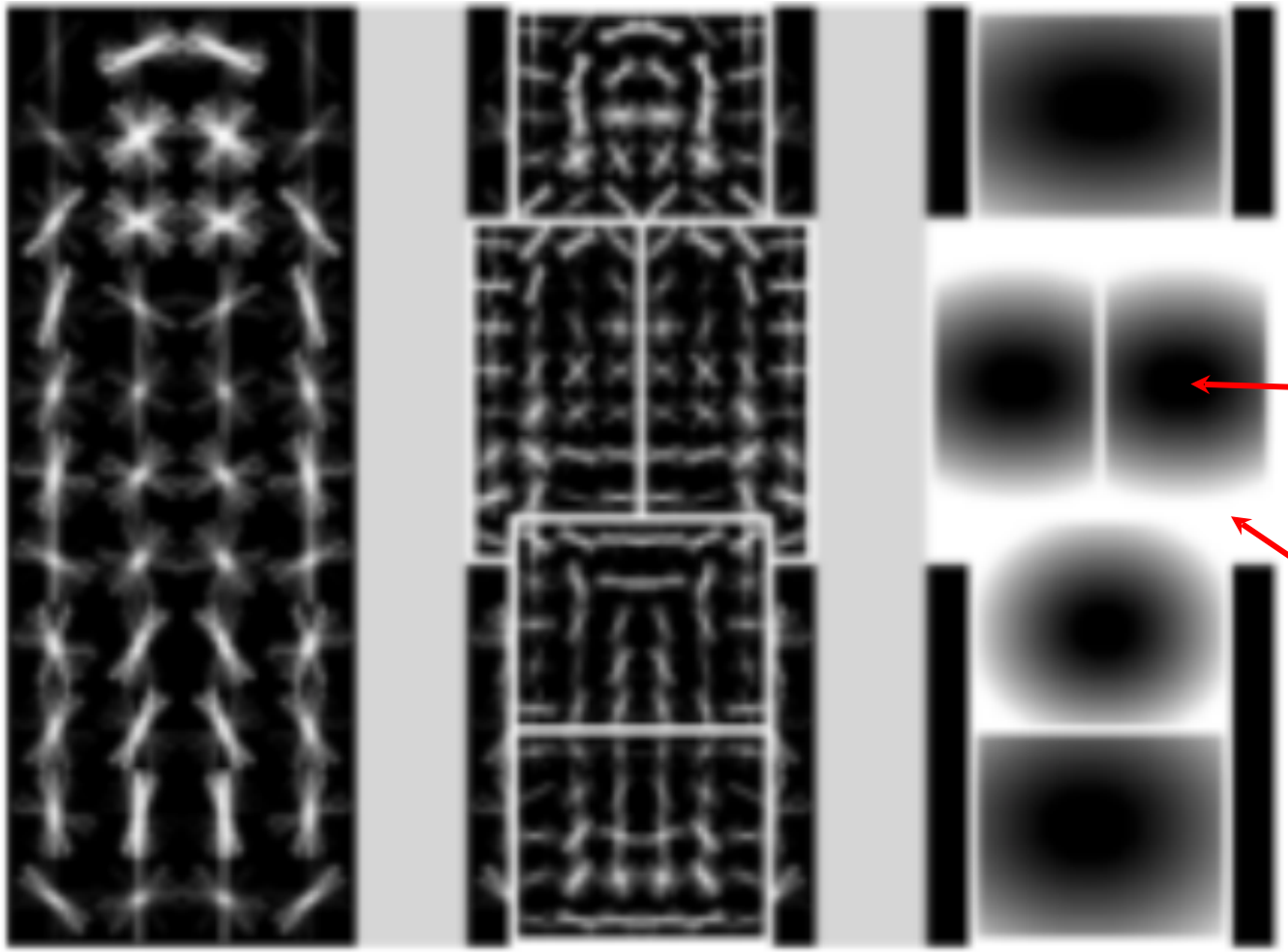
DPM for person model with 5 parts



If the head is here,
the score is high

If the head is here,
the score is low

DPM for person model with 5 parts



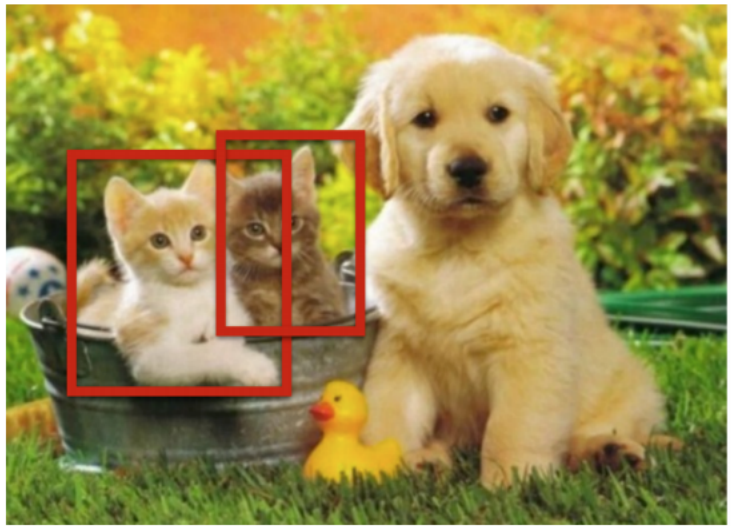
If the arm is here,
the score is high

If the arm is here,
the score is low

How do we use the parts to make a detection?

Intuition:

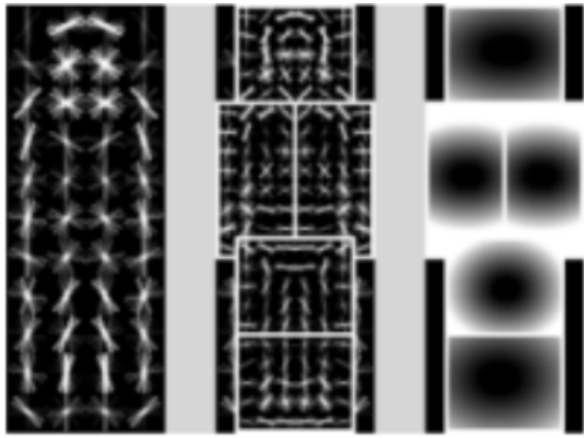
1. First, use the sliding windows at different pyramid scales to detect each part (and the root).
2. Each part gives you a score for where the person might be
3. Accumulate the global and part scores (and penalize the deformation of the parts.)



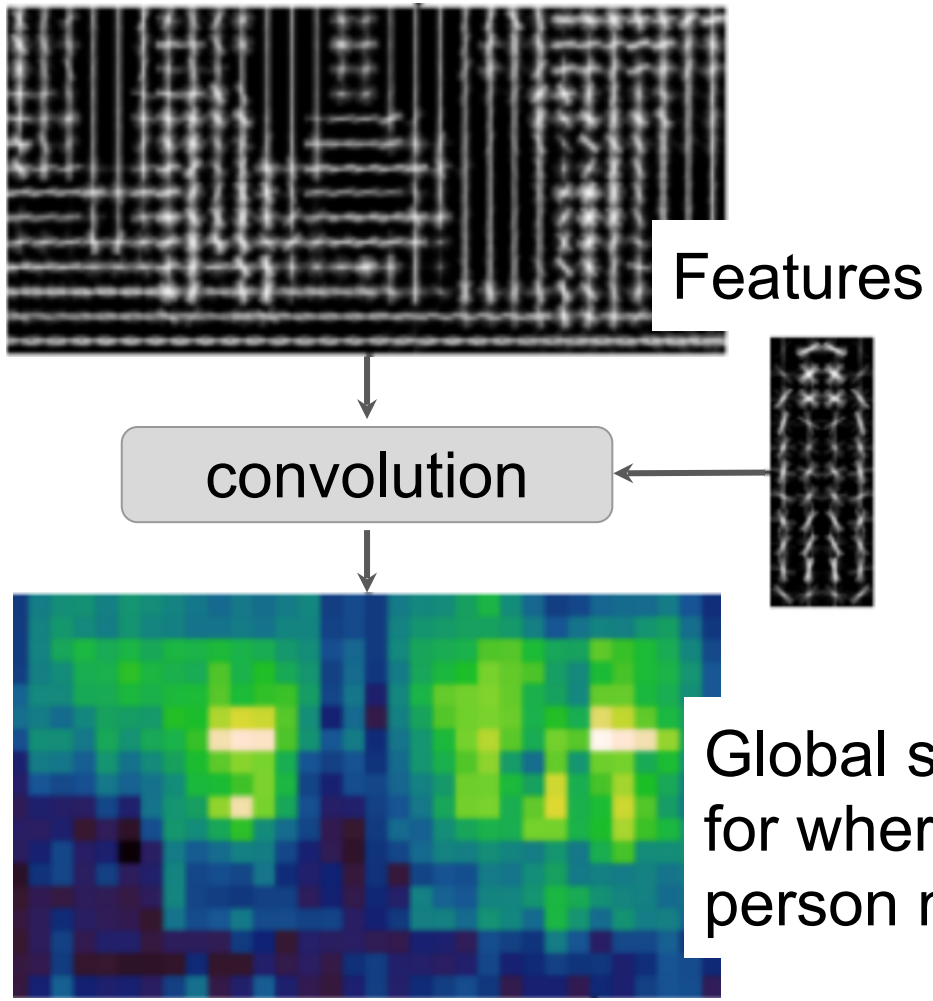
Example for detecting people

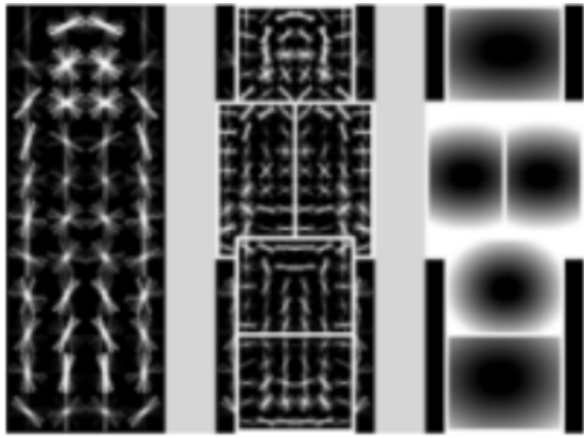
Image input

A feature template for person

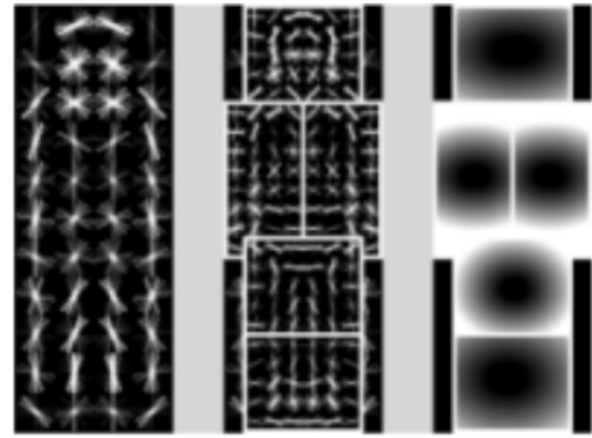


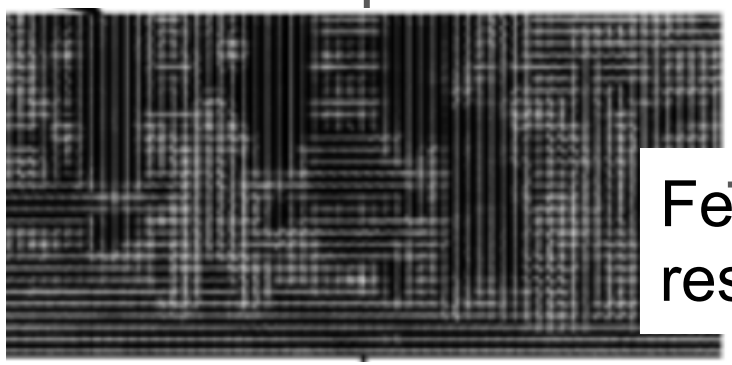
Calculate scores for global template





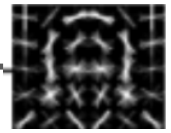
Calculate scores for part templates



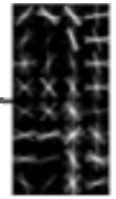


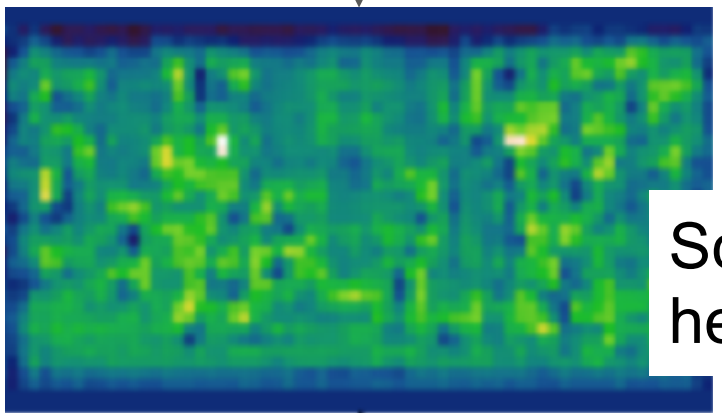
Features at 2x resolution

convolution

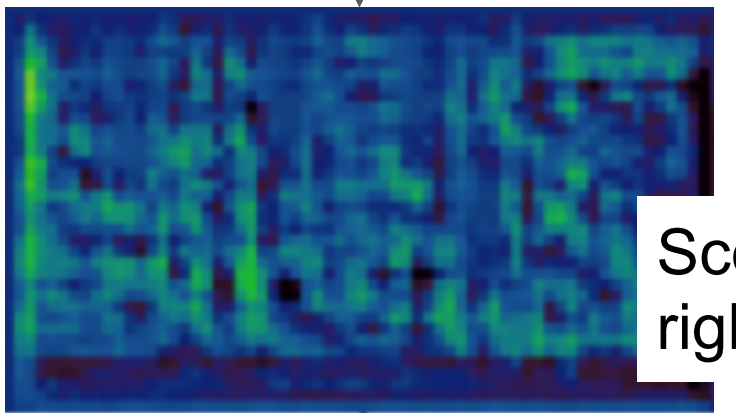


convolution





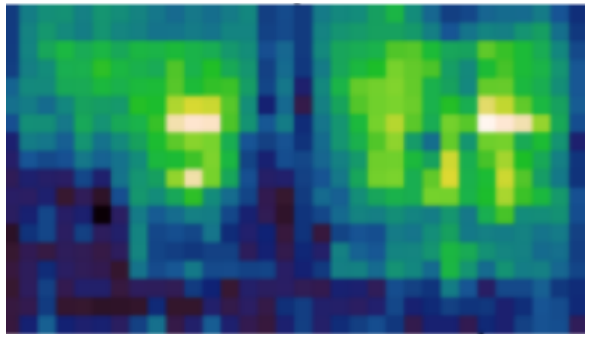
Scores for head



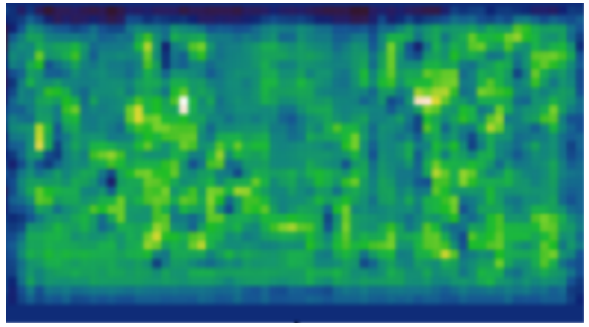
Scores for right arm

After step 1, we have scores for all parts and global template

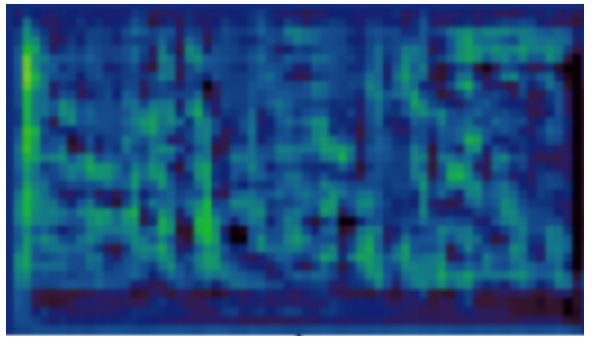
Global scores

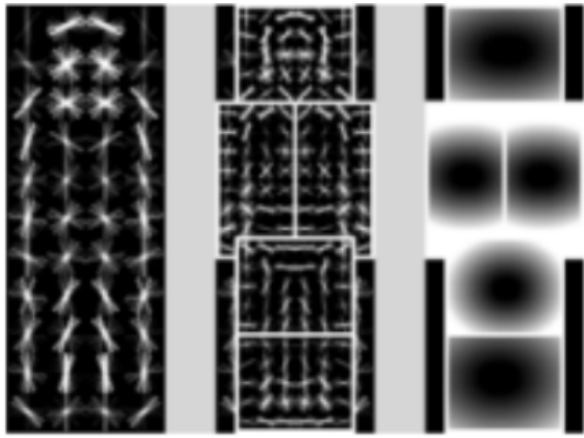


Head scores



Right arm scores



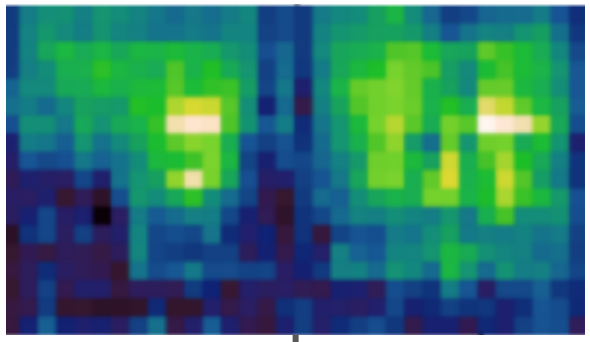


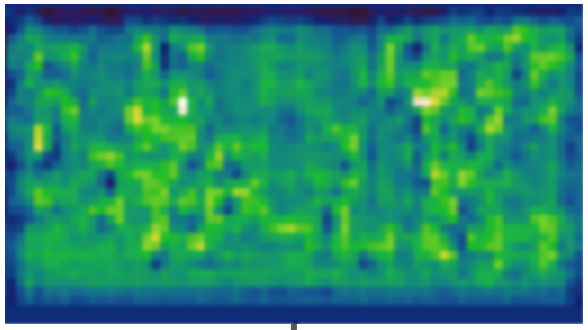
Step 3: Add up the scores for the final detections

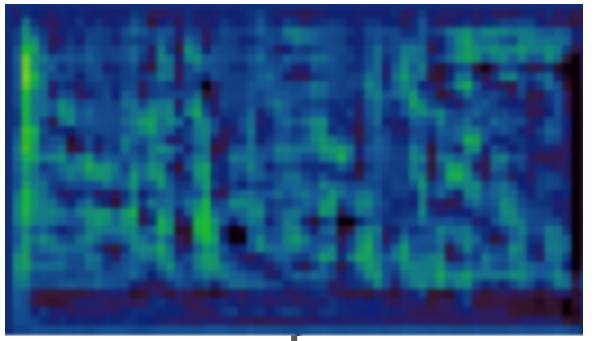
Head scores

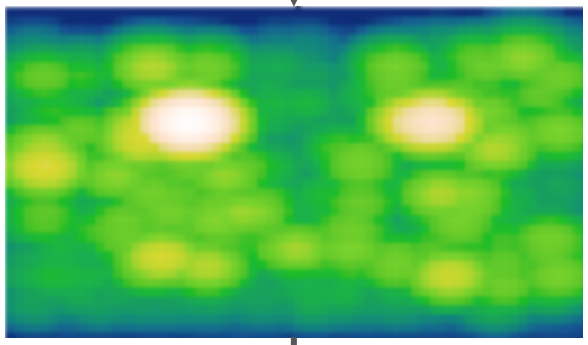
Right arm scores

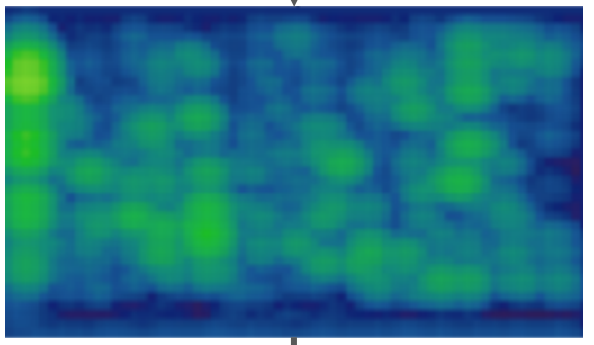
Global scores

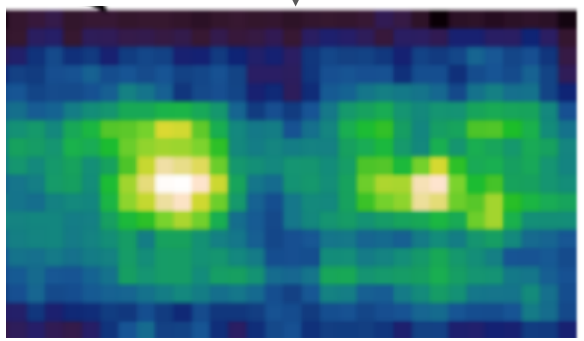




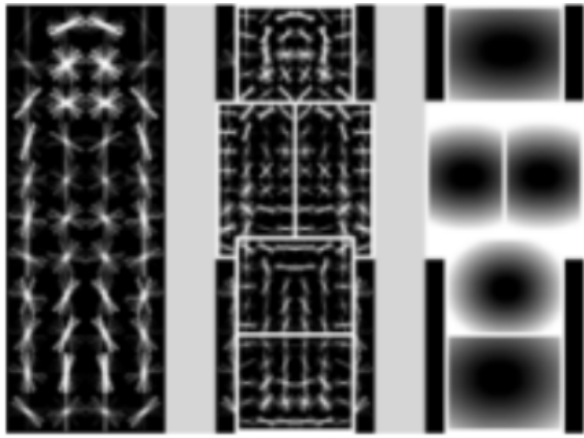








Add up final scores



Deformation: score propagation (Your HW4)