Detection & recognition

Tae Hyun Kim

Roadmap

Classification

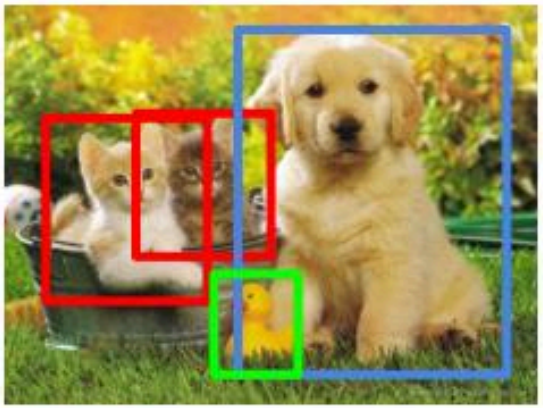
CAT

Classification + Localization

↳ 위치 찾음

CAT

Object Detection



CAT, DOG, DUCK

Instance Segmentation

CAT, DOG, DUCK

Single object

Multiple objects

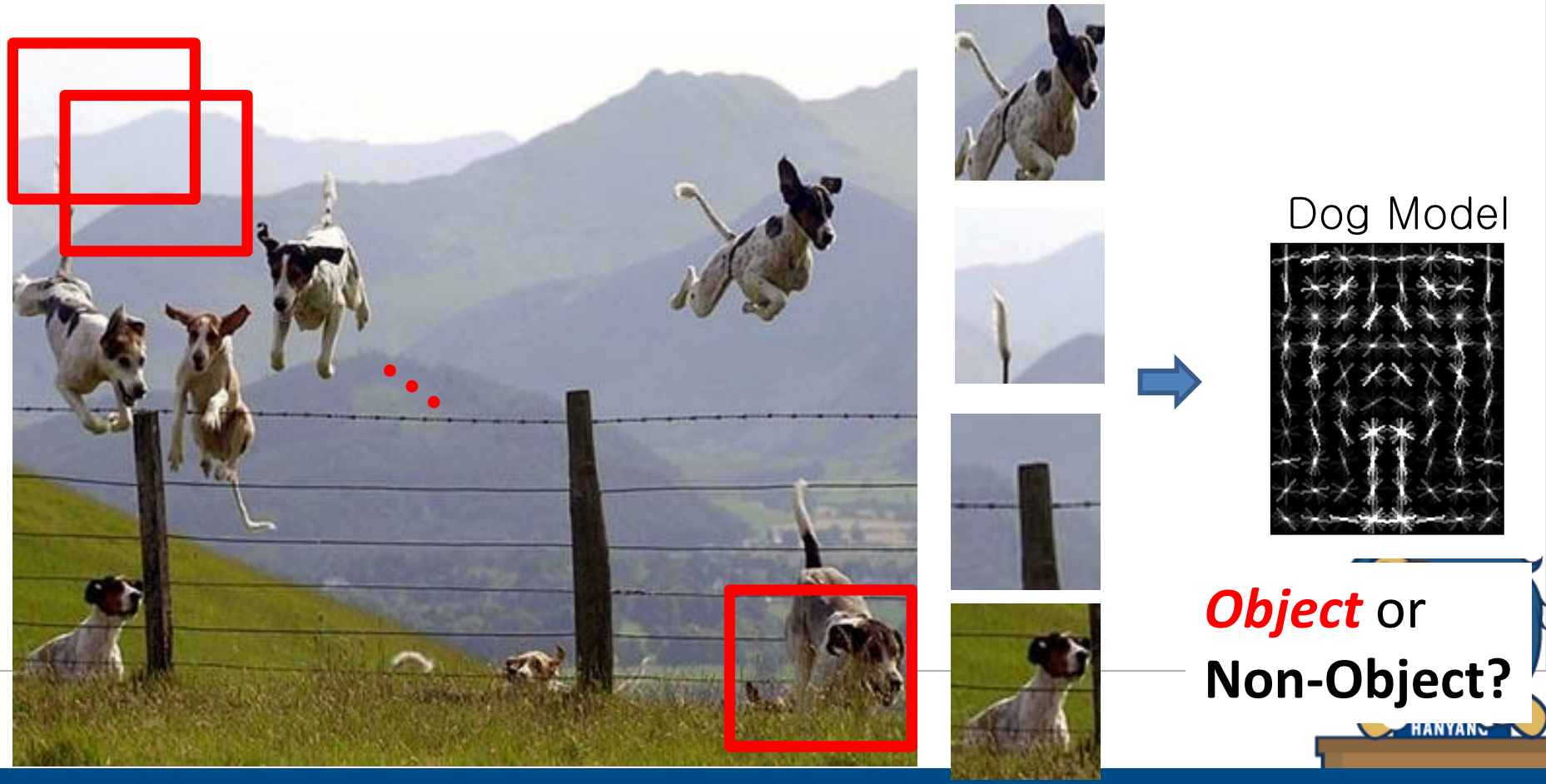
Slide from A. Karpathy

State-of-the-art Detection Method

- YOLO v2 (CVPR'18)

Object Category Detection

- Focus on object search: “Where is it?”
- Build templates that quickly differentiate object patch from background patch



Challenges in modeling the object class

Illumination

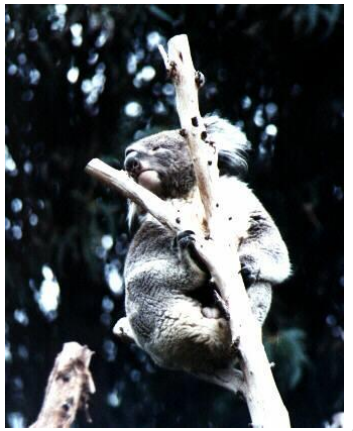
Object pose

Clutter

mv variational 공부

Occlusions

Intra-class appearance



Viewpoint

Slide from K. Grauman, B. Leibe

Challenges in modeling the non-object class

True Detections

Bad Localization

Confused with Similar Object

Misc. Background

Confused with Dissimilar Objects

General Process of Object Detection

Specify Object Model

What are the object parameters?

찾고자하는 대상들에 대한 정보 target object 찾아 봅기

Generate Hypotheses

Score Hypotheses

Resolve Detections

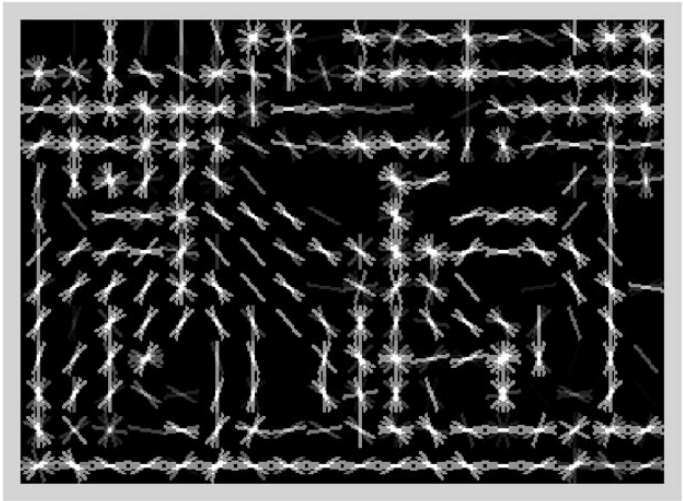
Specifying an object model

1. Statistical Template in Bounding Box
   - Object is somewhere (x,y,w,h) in image
   - Features defined w.r.t. bounding box coordinates

찾고자 하는 object의 피처와
'좋은 피처' – 같은 카테고리 안에서 공통적인 피처

Image

gradient 정보



Template Visualization

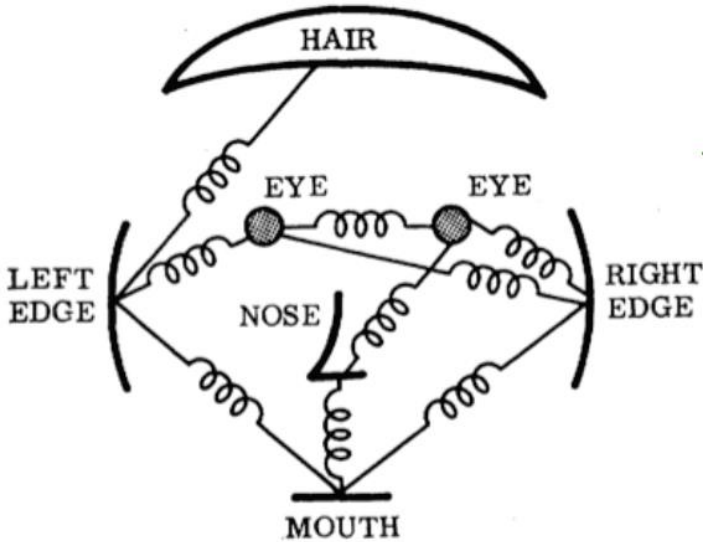
Images from Felzenszwalb

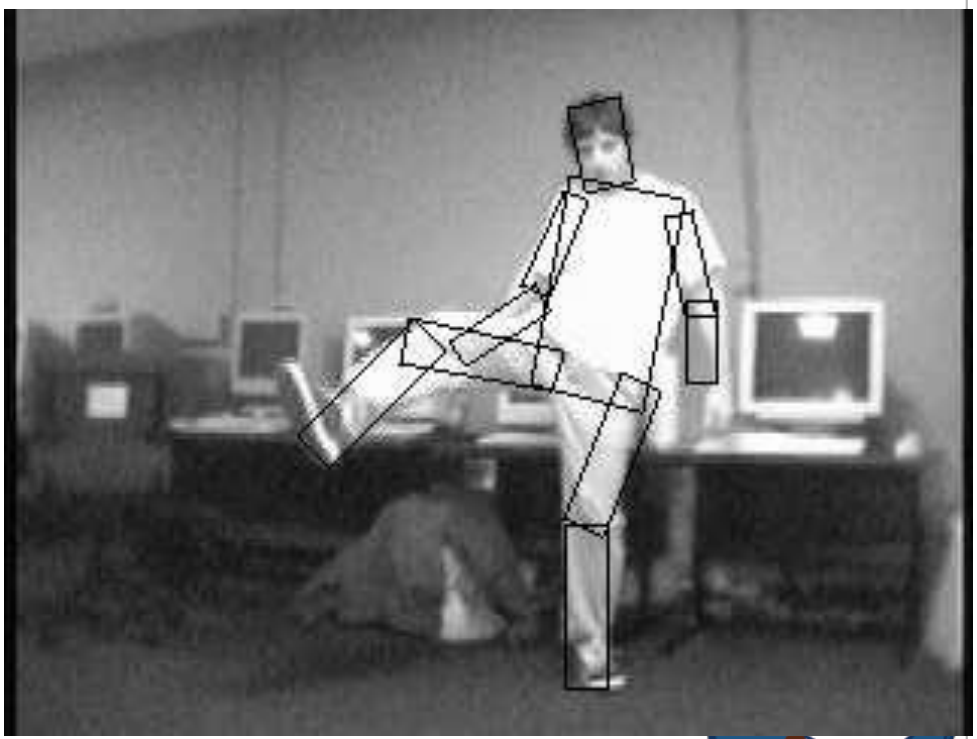
Specifying an object model

2. Articulated parts model

- Object is configuration of parts
- Each part is detectable

사전 constraint를 가지고 object define

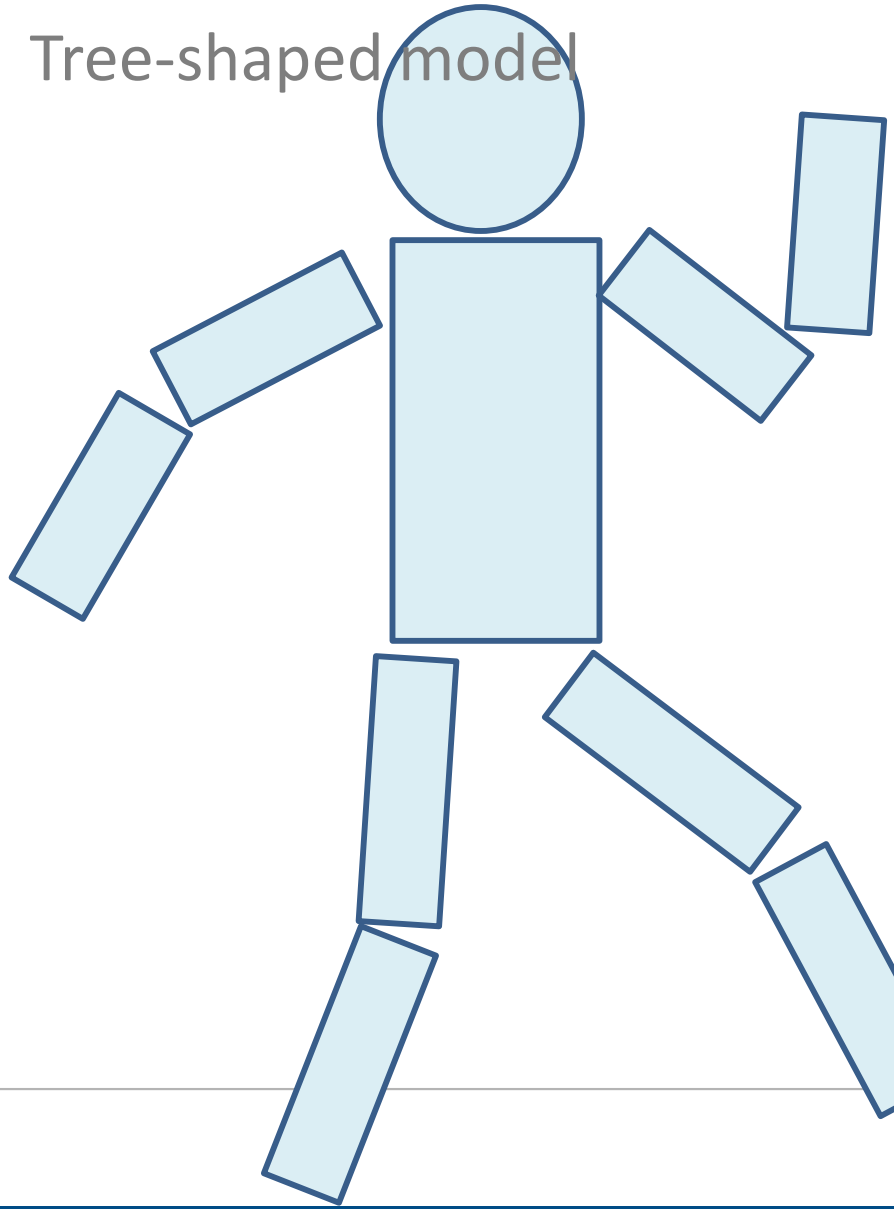


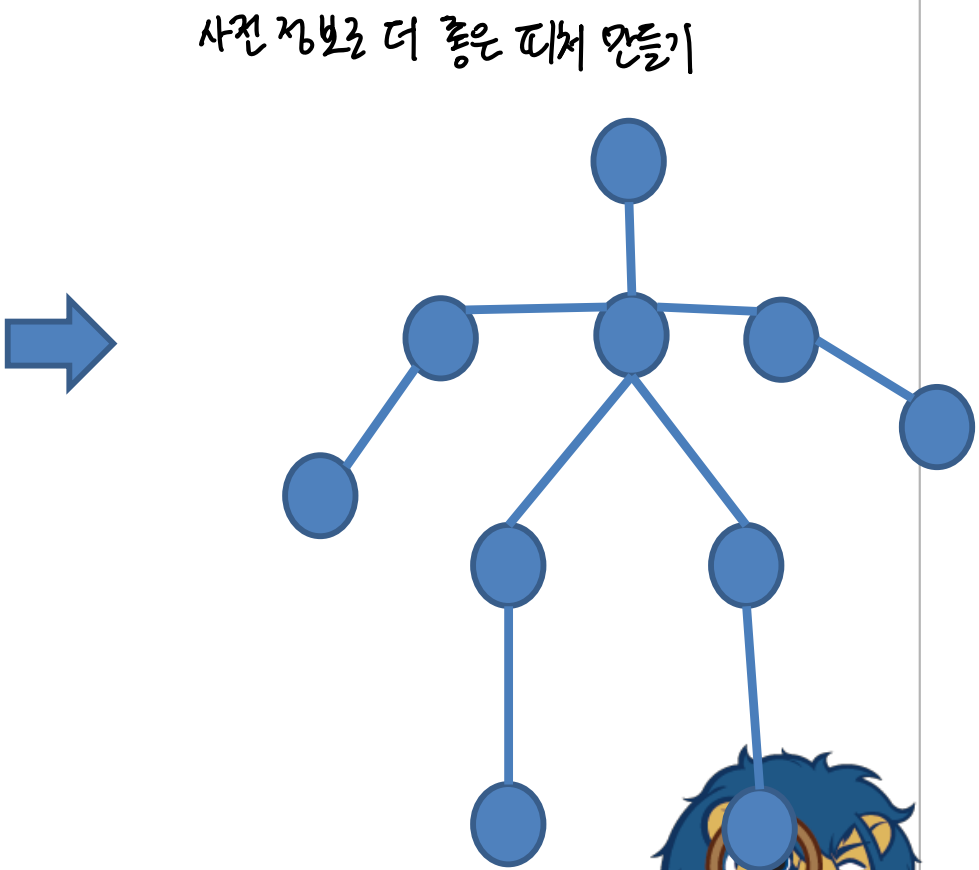


Images from Felzenszwalb

How to model spatial relations?

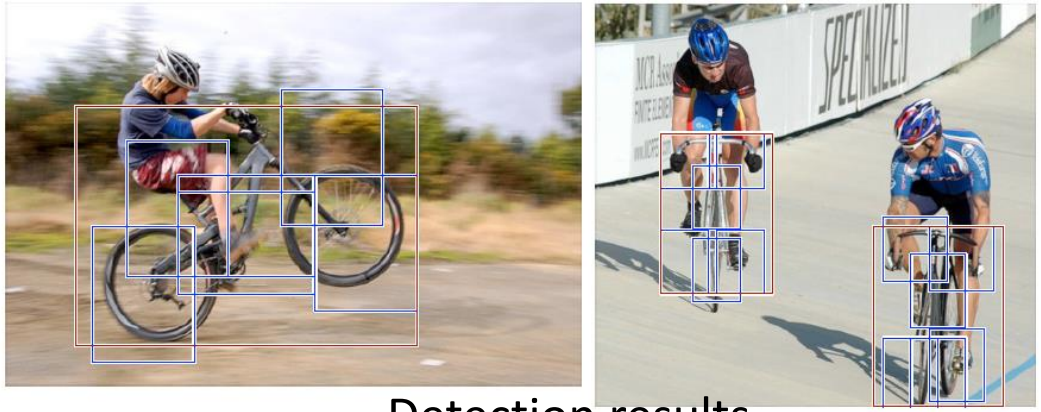
- Tree-shaped model



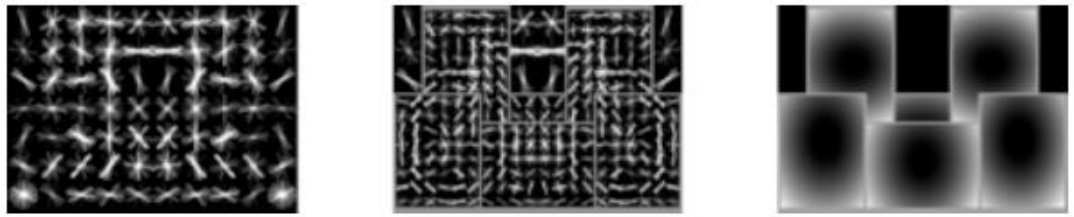


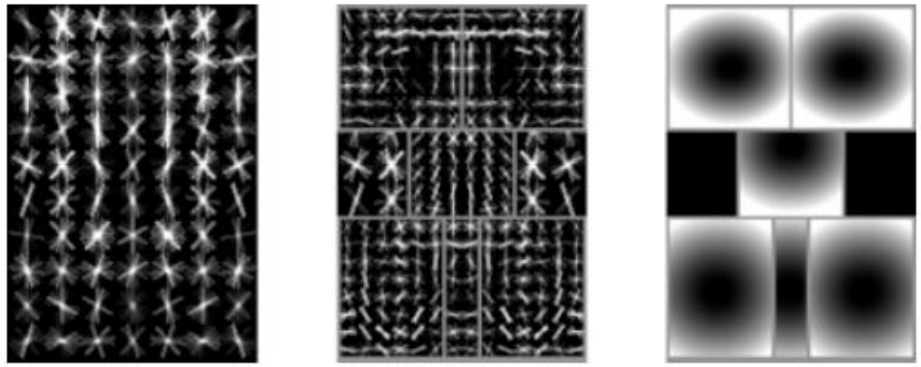
Specifying an object model

3. Hybrid template/ parts model



Detection results





Template Visualization

root filters coarse resolution

part filters finer resolution

deformation models

Felzenszwalb et al. 2008

General Process of Object Detection

Specify Object Model

피처 만듦

Generate Hypotheses

Propose object candidates in the given image

찾은 object가 주어진 이미지에서 잘 발견되는지

Score Hypotheses

Resolve Detections

Images from Felzenszwalb

Generating hypotheses

1. Sliding window
   - Test patch at each location and scale

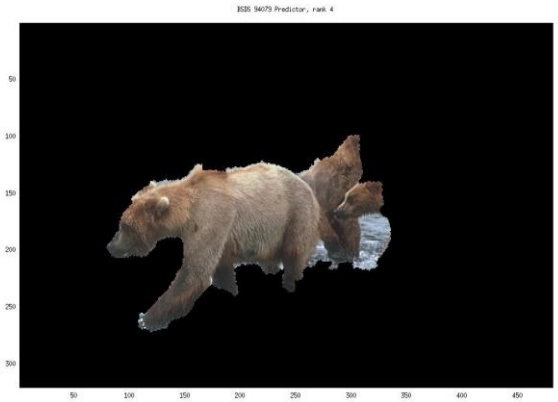
Generated hypotheses

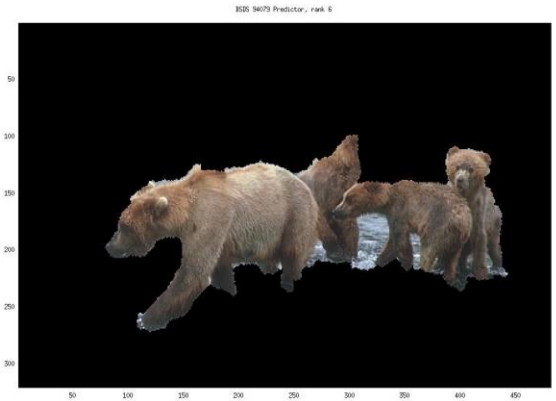
candidate → object 유무 검사

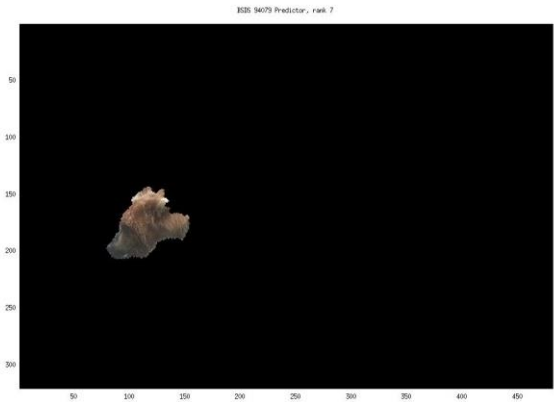
Generating hypotheses

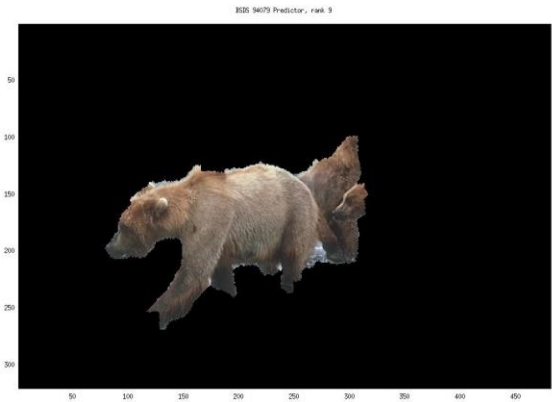
Region-based proposal

비슷한 컬러끼리 뭉쳐놈









Endres Hoiem 2010

General Process of Object Detection

Specify Object Model

↓

Generate Hypotheses

후보 생성

↓

Score Hypotheses

Compare hypotheses and the trained models (templates).
Traditional classifiers/Neural nets

피처의 유사도 측정 ex) 유클리디안 distance
Similarity

↓

Resolve Detections

General Process of Object Detection

Specify Object Model

↓

Generate Hypotheses

↓

Score Hypotheses

↓

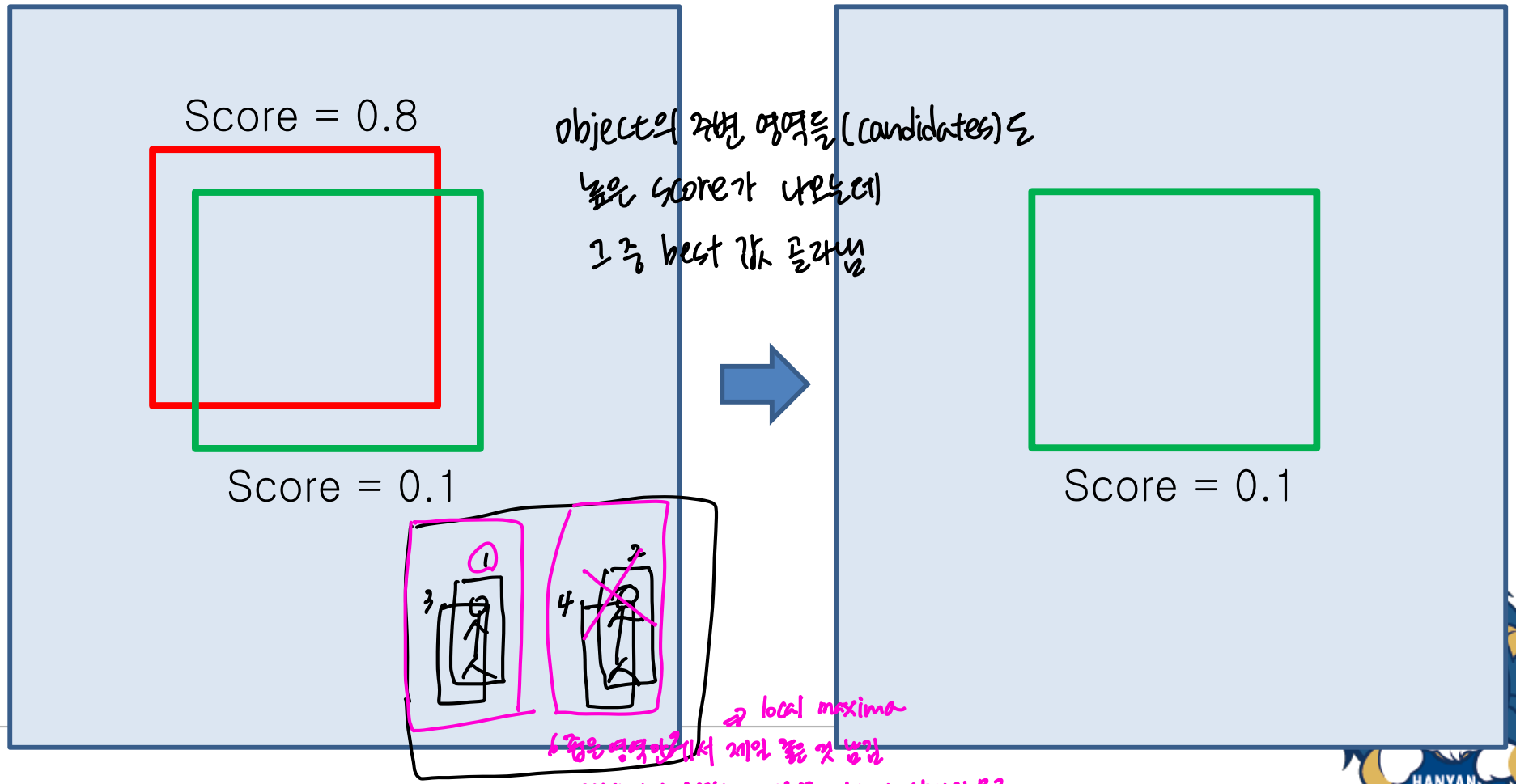
Resolve Detections

피처가 optimal 하지 않음

피처의 퀄리티 차이

Resolving detection scores

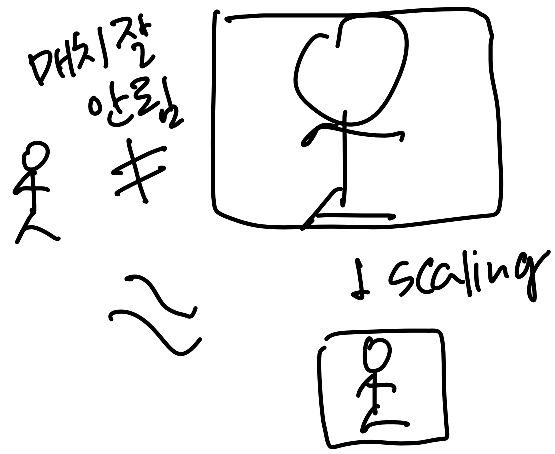
- Non-max suppression (NMS) 꼭 필요한 단계
  - Find local maxima



Object category detection in computer vision

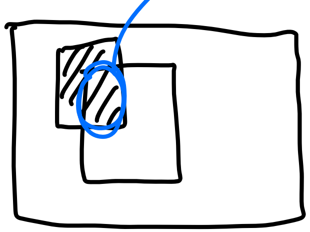
- Goal: detect all pedestrians, cars, monkeys, etc in image

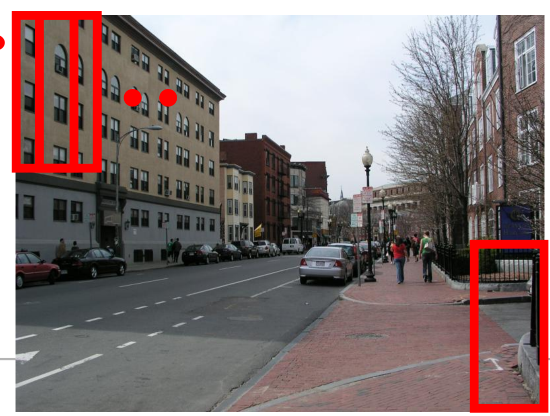
Sliding window: a simple alignment solution



모든 패치들에 대해 사람이 들어있는지 검사

중복된 피처로 바꾸는 과정 2번 거침 - 비효율적
→ 뉴럴 approach에서 많이 해결됨





Each window is separately classified

각 candidate에서 피처 뽑아서 유사도 측정

Design challenges

- How to efficiently search for likely objects
  - Even simple models require searching ***hundreds of thousands of positions and scales*** 스케일 고려
- Feature design and scoring
  - How should ***appearance*** be modeled? What features correspond to the object? 외형 무시하고 같은 카테고리라는 것을 알수있는 피처
- How to deal with different viewpoints?
  - Often train different models for a few ***different viewpoints***
- Implementation details
  - Window size
  - Aspect ratio
  - Translation/scale step size
  - Non-maxima suppression

object detector 보행자 검출

Example: Dalal-Triggs detector

1. Extract fixed-sized (64x128 pixel) window ***at each position and scale***

candidates 만듦

2. Compute ***HOG*** (histogram of gradient) features within each window

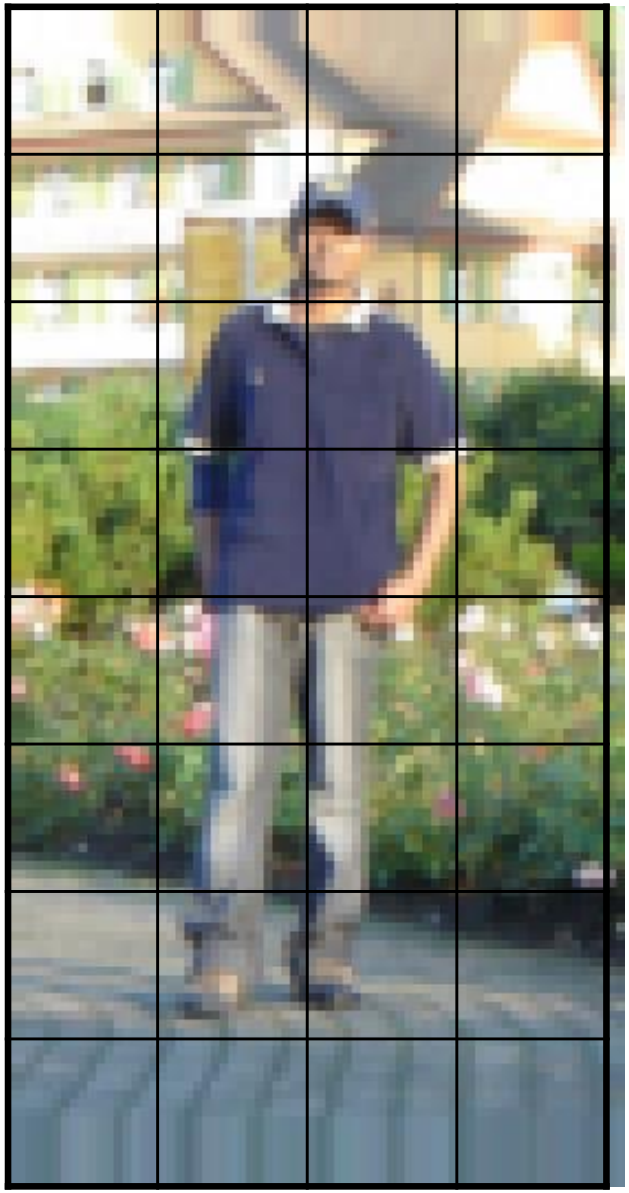
각 candidates에서 피쳐 추출

3. Score the window with a linear ***SVM classifier***

내가 가지고 있는 피쳐와 유사도 측정

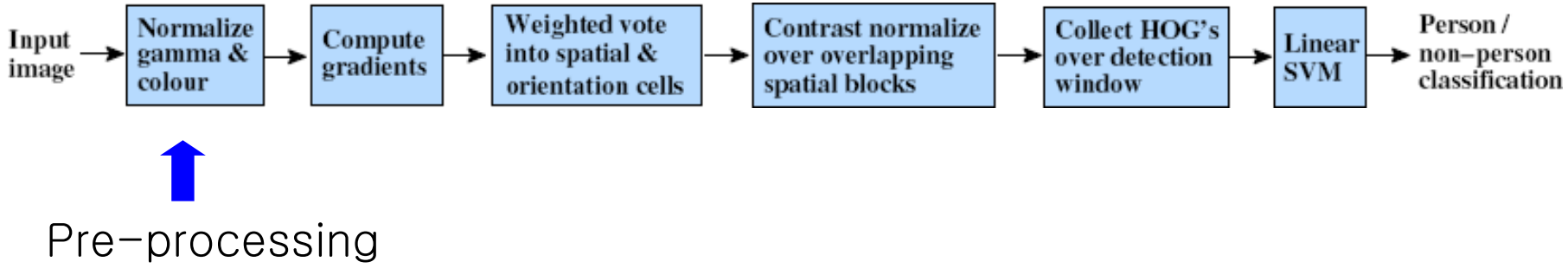
4. Perform ***non-maxima suppression*** to remove overlapping detections with lower scores

〈 feature extraction overall flow 〉



Slides by Pete Barnum

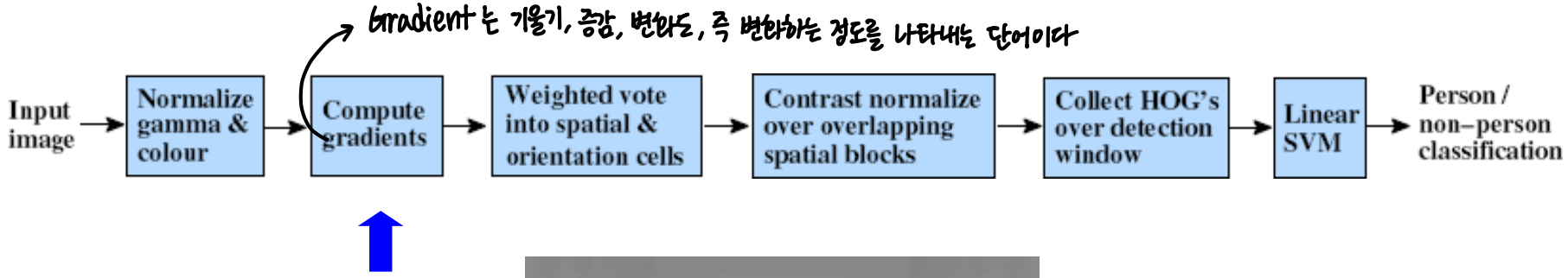
Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05



- Tested with
  - RGB
  - LAB
  - Grayscale

  } Slightly better performance vs. grayscale
- Gamma Normalization and Compression
  - Square root
  - Log

  } Very slightly better performance vs. no adjustment



Outperforms

-1	0	1

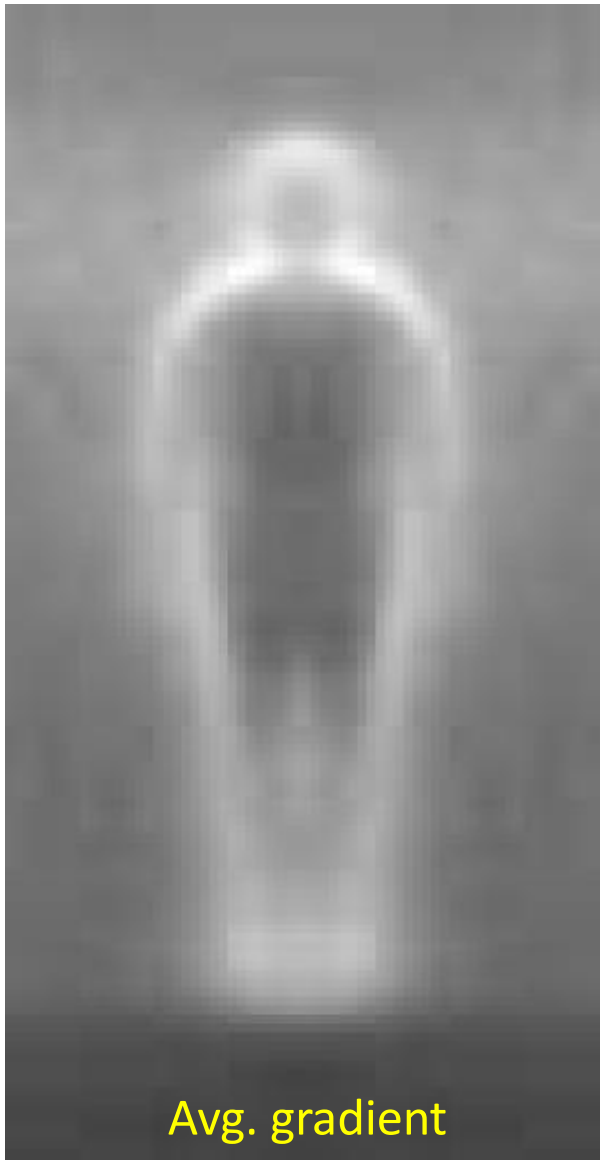
centered

-1	1

uncentered

1	-8	0	8	-1

cubic-corrected



gradient 정보 활용

0	1
-1	0

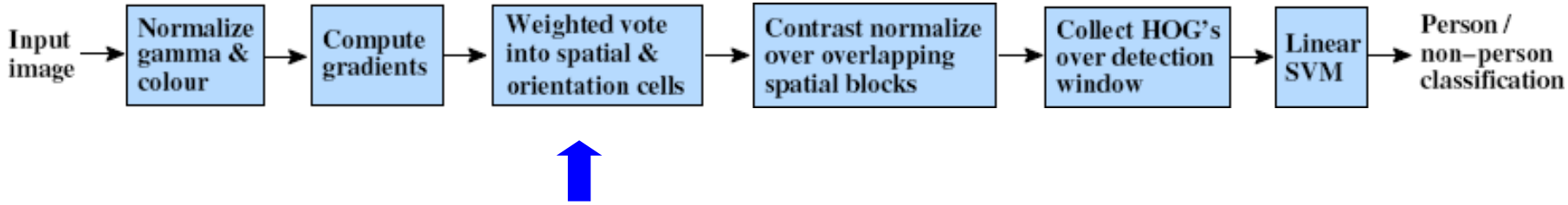
diagonal

-1	0	1
-2	0	2
-1	0	1

Sobel

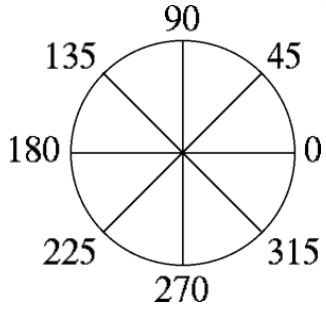
Slides by Pete Barnum

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

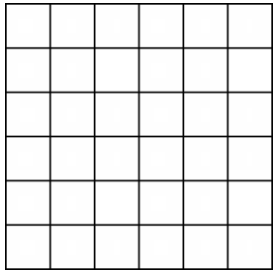


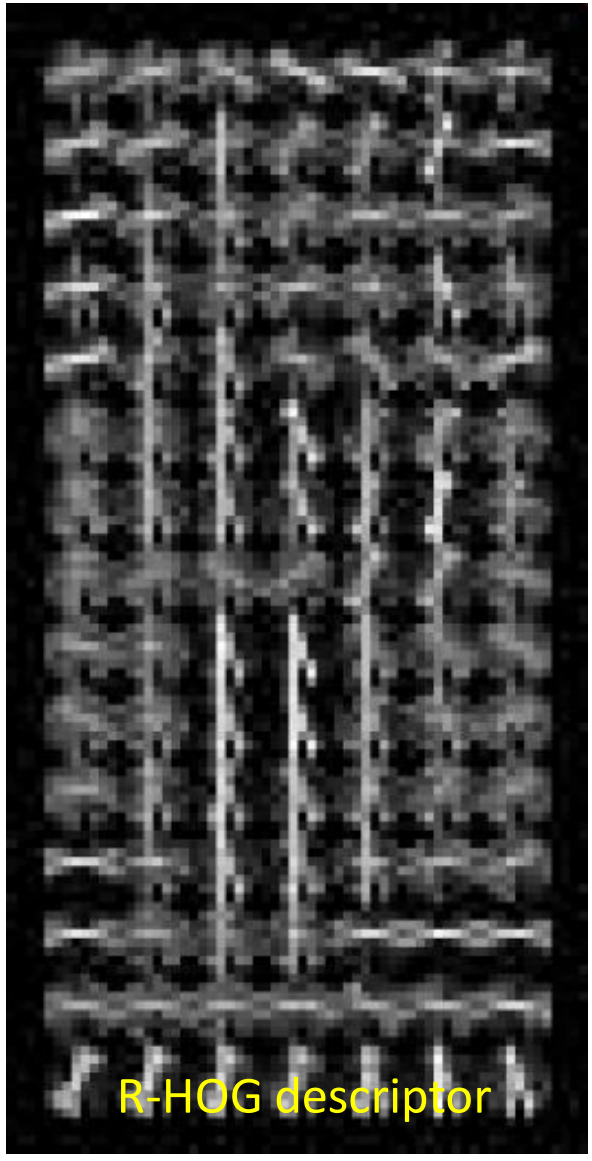
- Histogram of gradient orientations

Orientation: 9 bins (for unsigned angles)



Histograms in 8x8 pixel cells

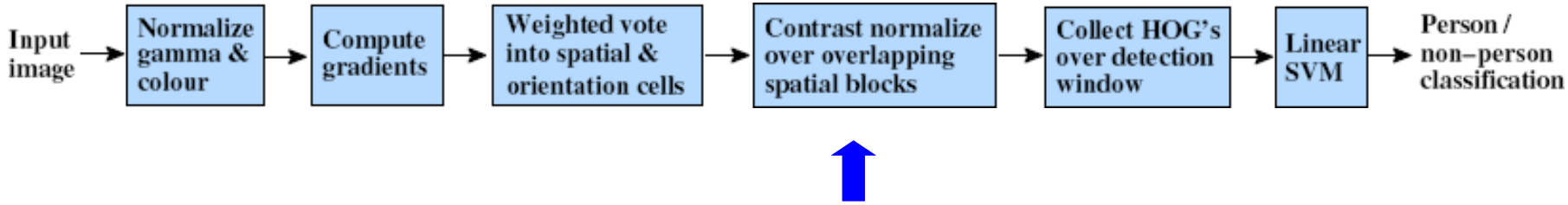




– Votes weighted by magnitude

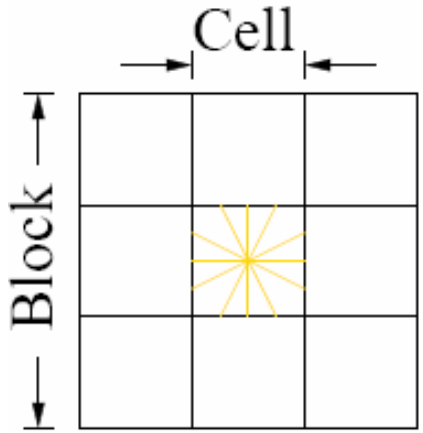
– Bilinear interpolation between cells

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



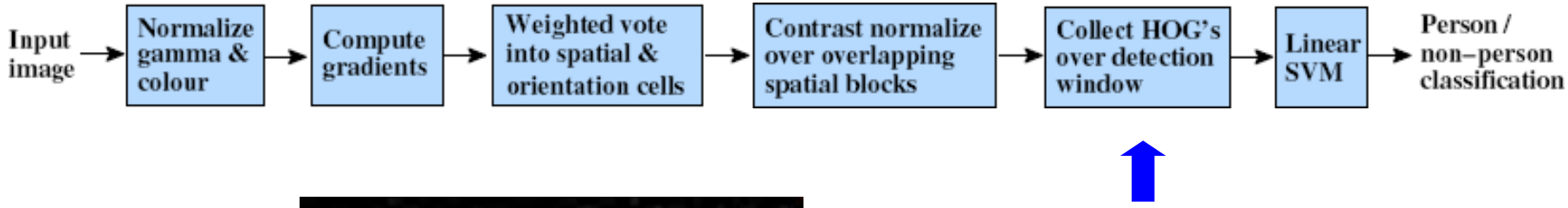
R-HOG

Normalize with respect to surrounding cells

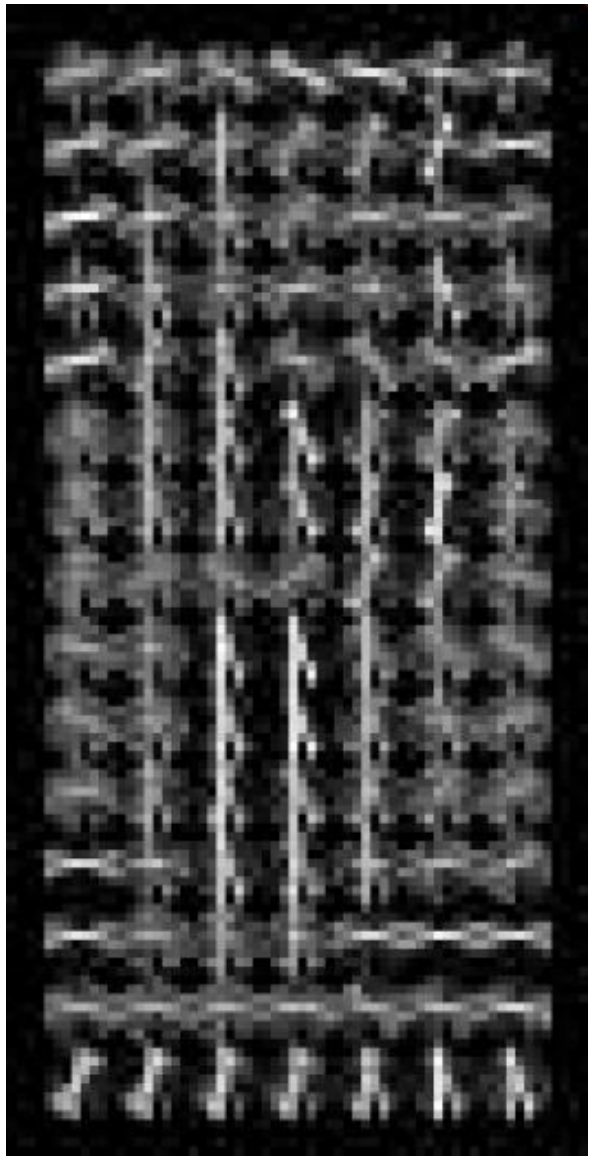


$$L2-norm: v \longrightarrow v/\sqrt{||v||_2^2+\epsilon^2}$$

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



X=



features = 15 x 7 x 9 x 4 = 3780

orientations

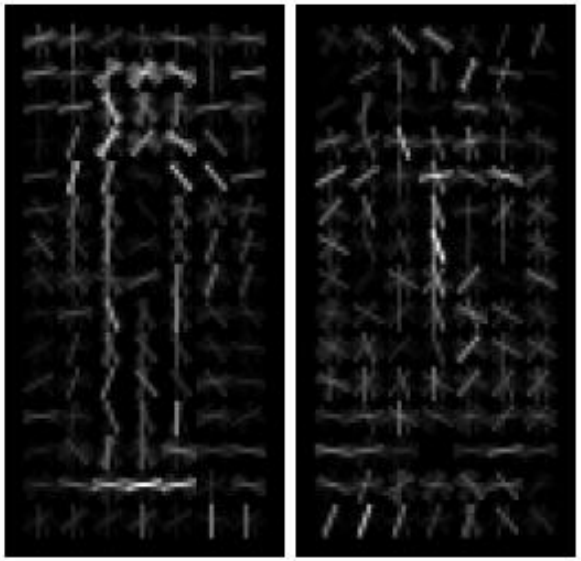
cells

normalizations by neighboring cells

Slides by Pete Barnum

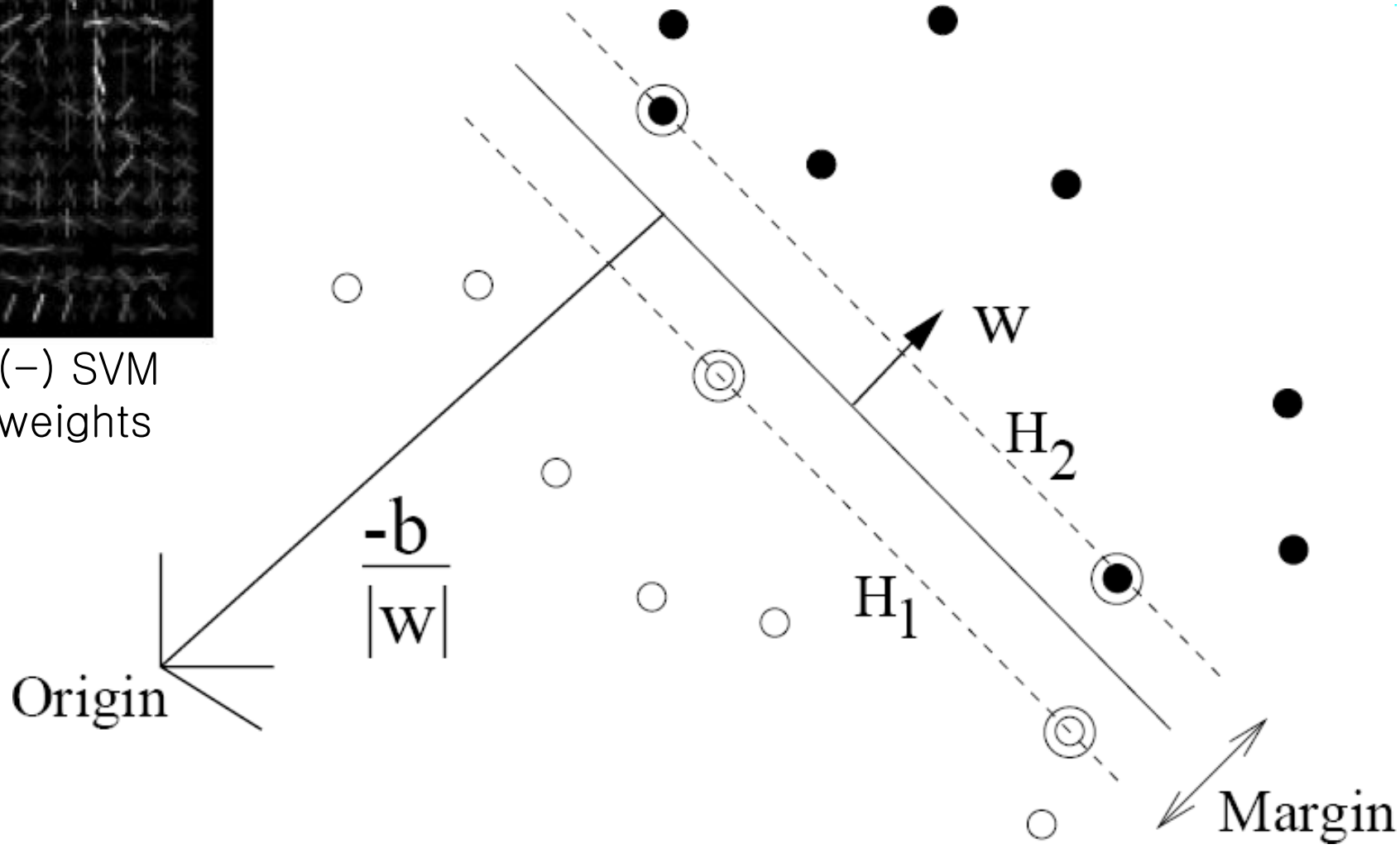
Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR05

classifier



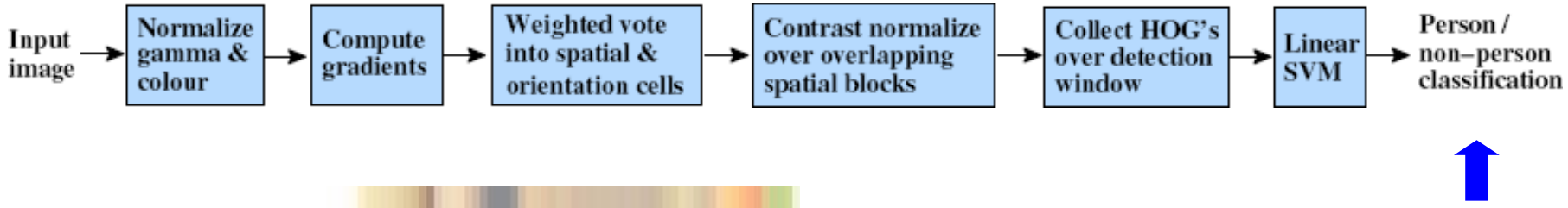
(+) SVM weights

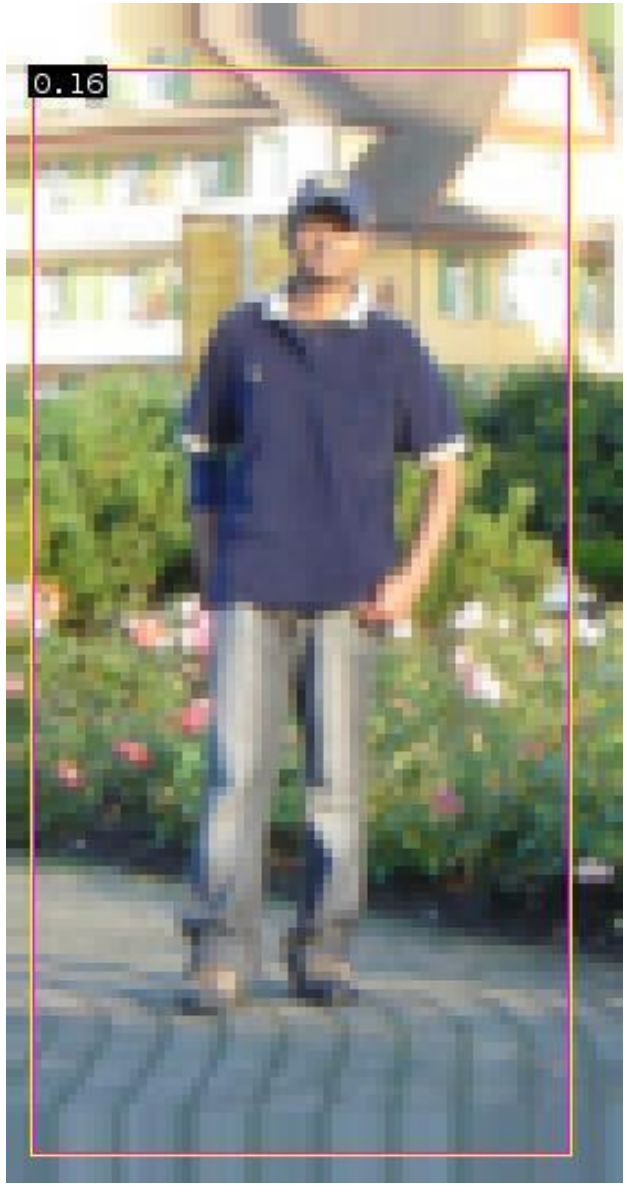
(−) SVM weights



Slides by Pete Barnum

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





$$0.16 = w^T x - b$$

$$sign(0.16) = 1$$

$$=>$$ pedestrian

Slides by Pete Barnum

Navneet Dalal and Bill Triggs, Histograms of Oriented Gradients for Human Detection, CVPR0

Detection examples

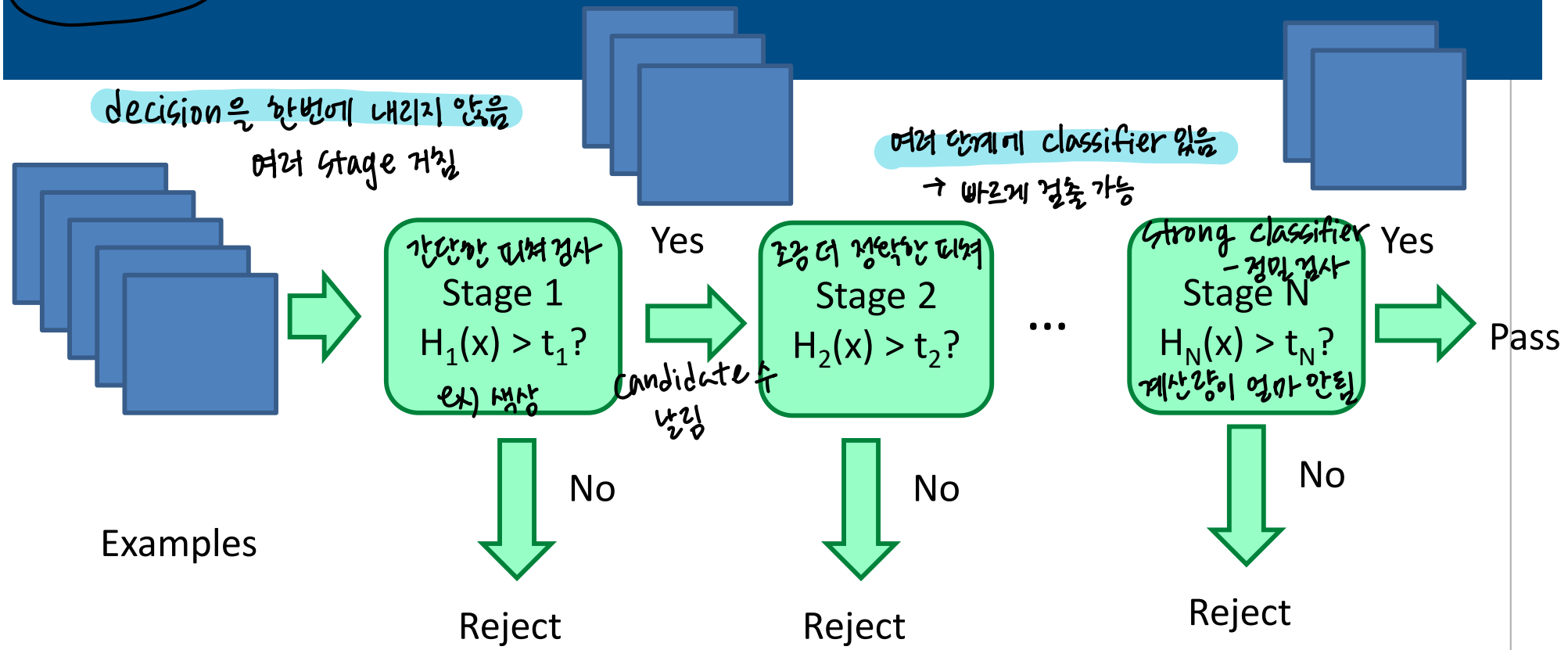
Viola-Jones sliding window detector

Fast detection through two mechanisms

- Quickly ***eliminate unlikely windows***
- Use ***features that are fast*** to compute

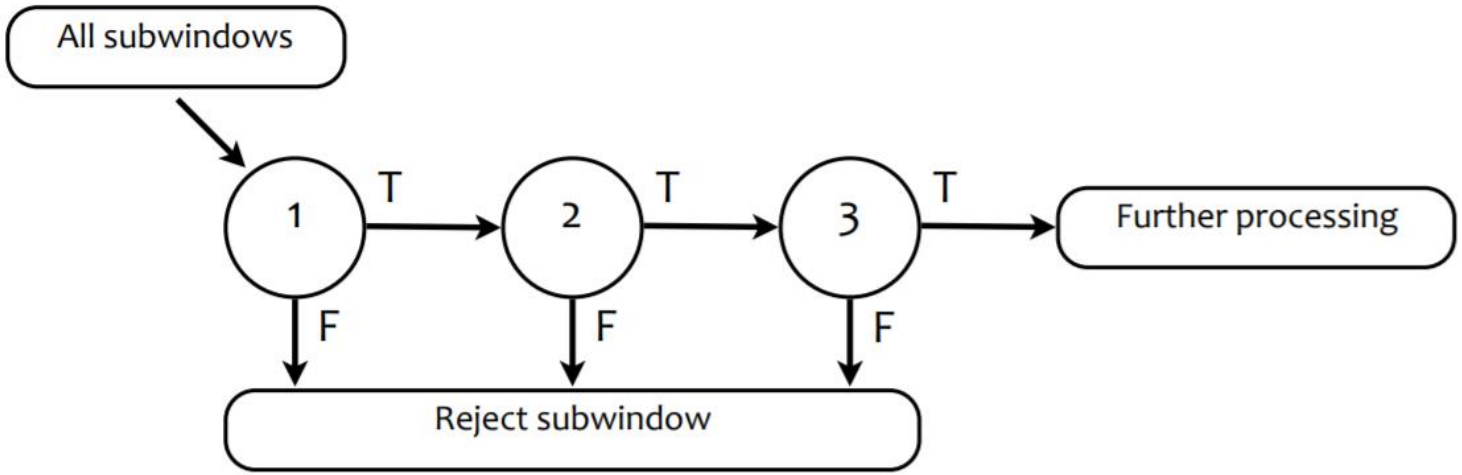
Cascade approach

Cascade for Fast Detection



- Choose threshold for low false negative rate
  - Avoid rejecting facial images
- Fast classifiers early stage
- Slow classifiers later, but most examples don't get there

Cascaded Classifier



The cascaded classifier is almost *10 times faster* since only positive examples are considered in the subsequent stages

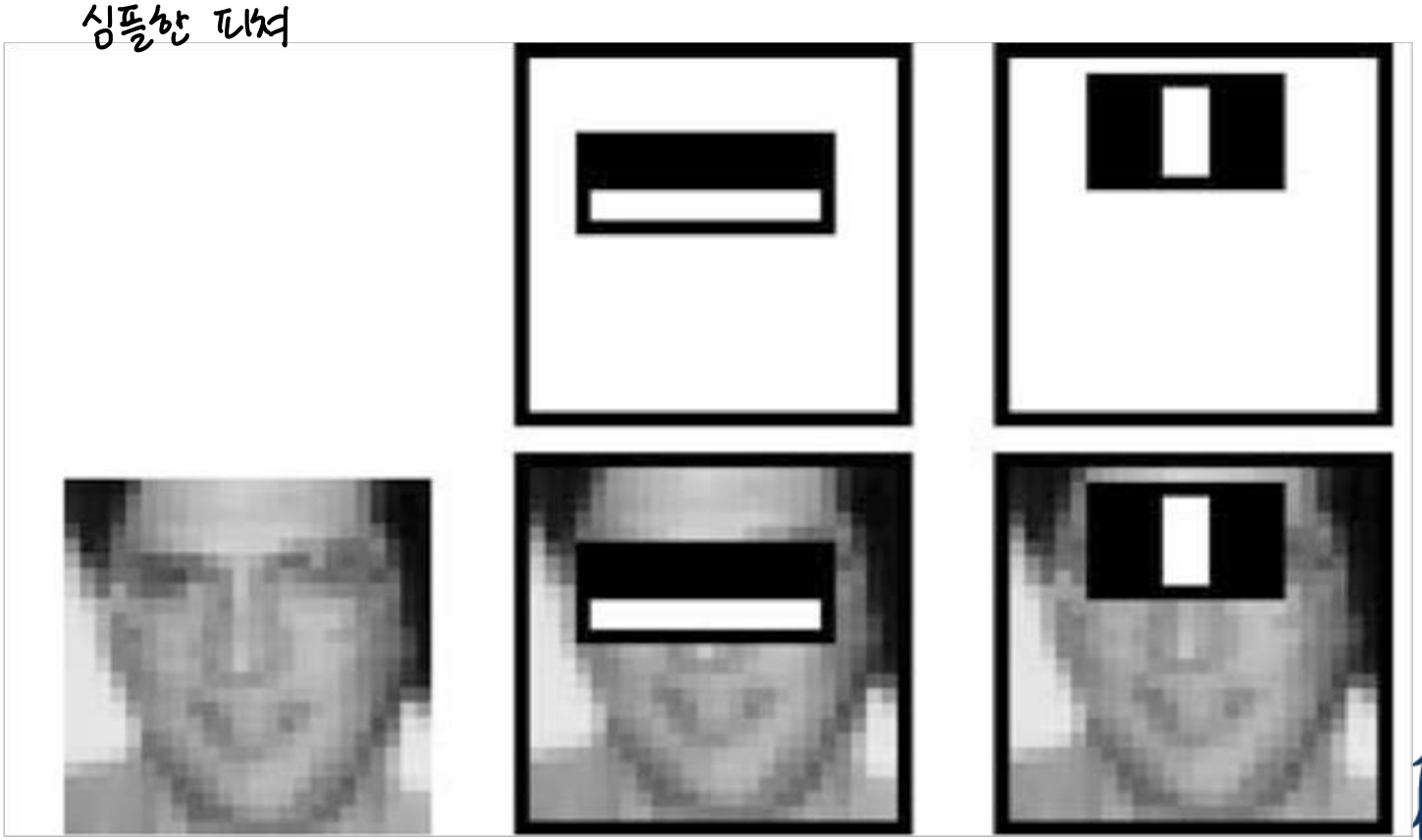
- We need to determine
  - The number of stages
  - Weak classifiers (features) in each stage.
  - The threshold of each stage
- Finding the optimal parameter is extremely difficult.

Characteristics of AdaBoost

- Training AdaBoost is the procedure to find
  - A ***set of weak classifiers*** with associated features
  - The ***weight*** of each weak classifier
- A collection of weak classifiers
  - Each ***rectangle feature*** is used to construct a weak classifier
  - The strong classifier is simply a set of weak classifiers
- An iterative training procedure
  - AdaBoost performs a series of trials.
  - A new weak classifier with associated weight is selected in a greedy manner in each iteration
  - Jointly find weak classifiers and extract features

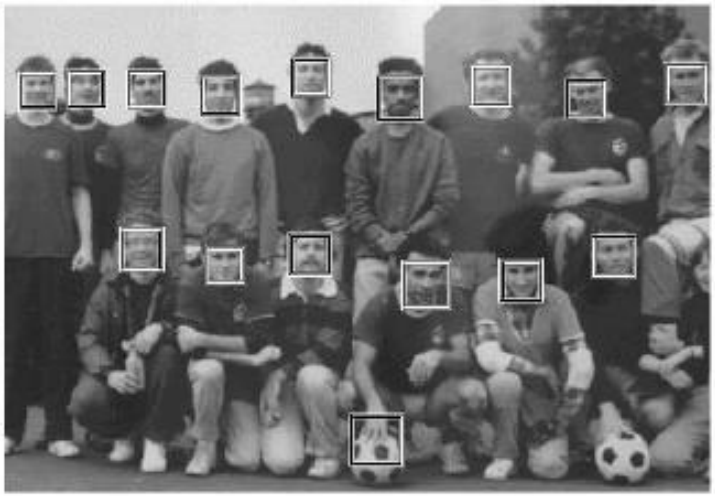
Top 2 selected features

- These two-feature classifier can be adjusted to detect 100% of the faces with a false positive rate of 50%



Viola Jones Results

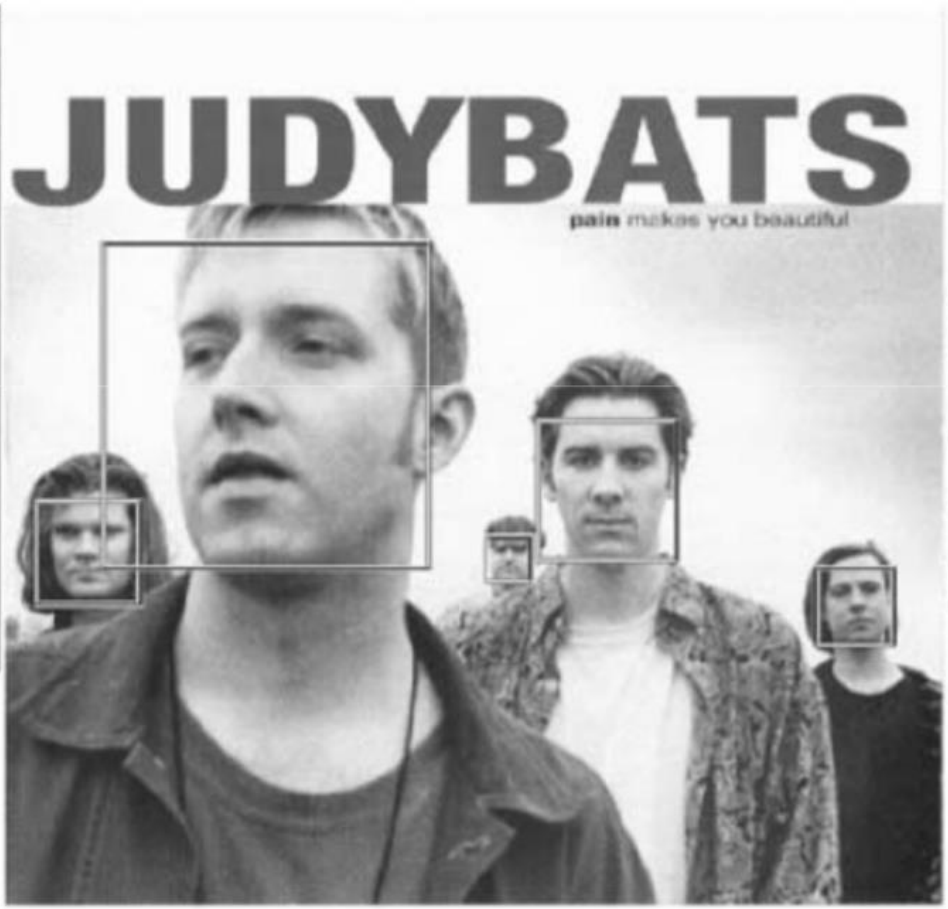
Speed = 15 FPS (in 2001)

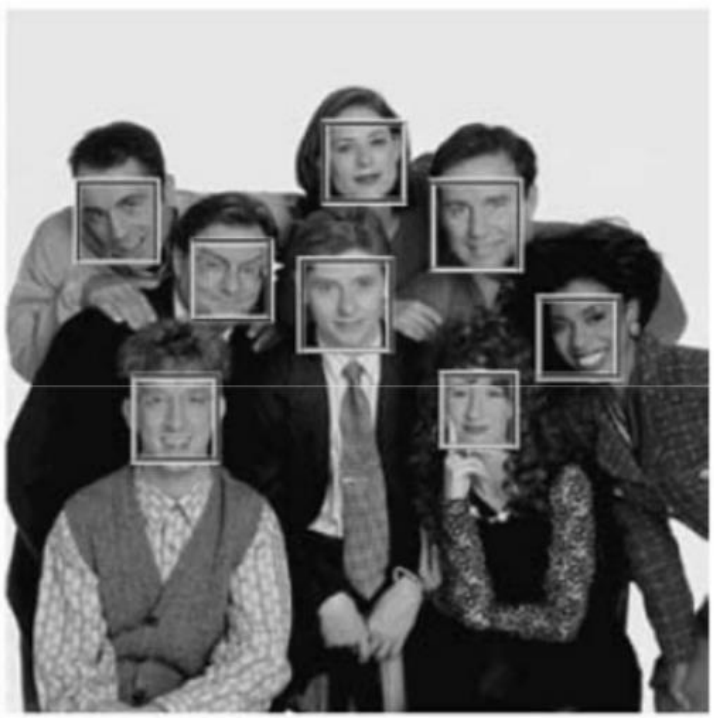


Detector \ False detections	10	31	50	65	78	95	167
Viola-Jones	76.1%	88.4%	91.4%	92.0%	92.1%	92.9%	93.9%
Viola-Jones (voting)	81.1%	89.7%	92.1%	93.1%	93.1%	93.2 %	93.7%
Rowley-Baluja-Kanade	83.2%	86.0%	-	-	-	89.2%	90.1%
Schneiderman-Kanade	-	-	-	94.4%	-	-	-
Roth-Yang-Ahuja	-	-	-	-	(94.8%)	-	-

MIT + CMU face dataset

Viola Jones Results





Thank you!