Pattern Recognition
Lecture 01-2

Feature Representation & HOG

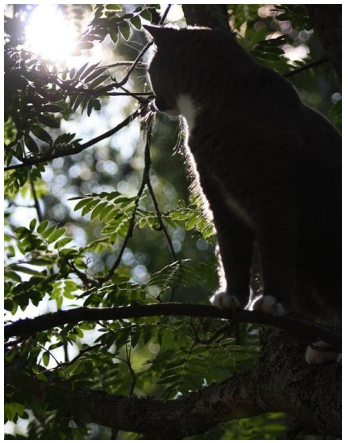
Prof. Jongwon Choi
Chung-Ang University
Fall 2022

Challenges: Illumination

This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

This image is CC0 1.0 public domain

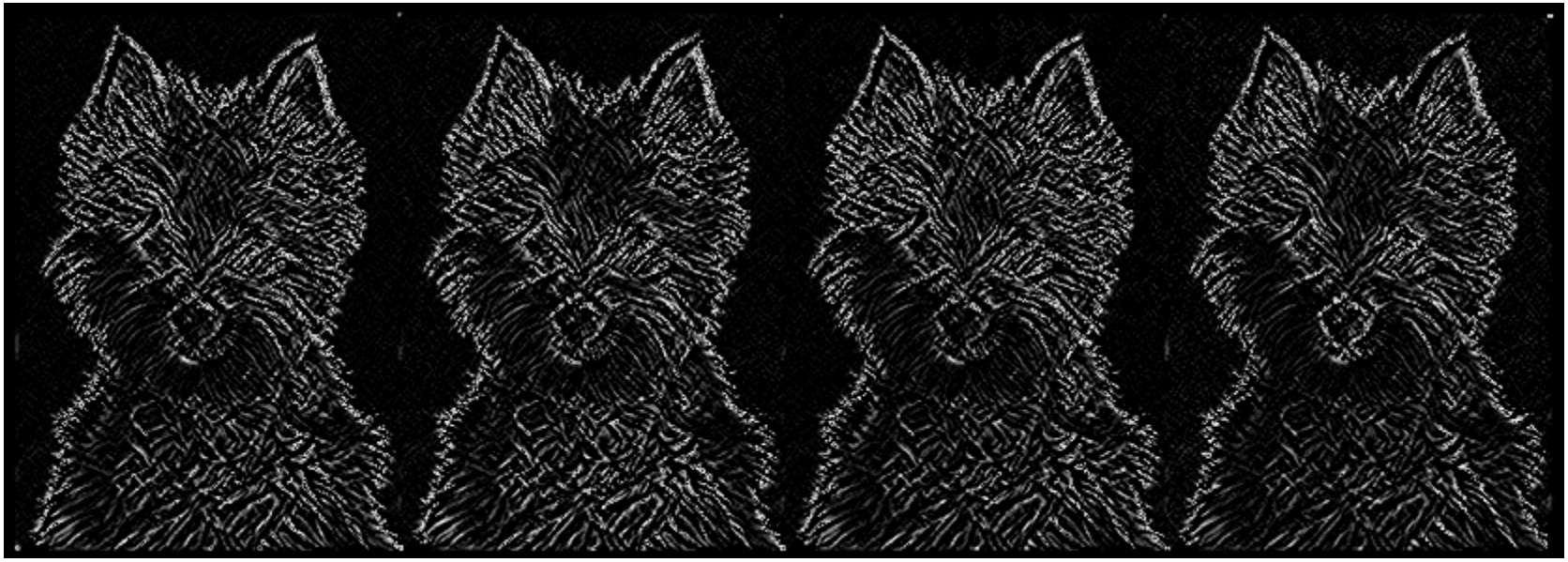


This image is CC0 1.0 public domain

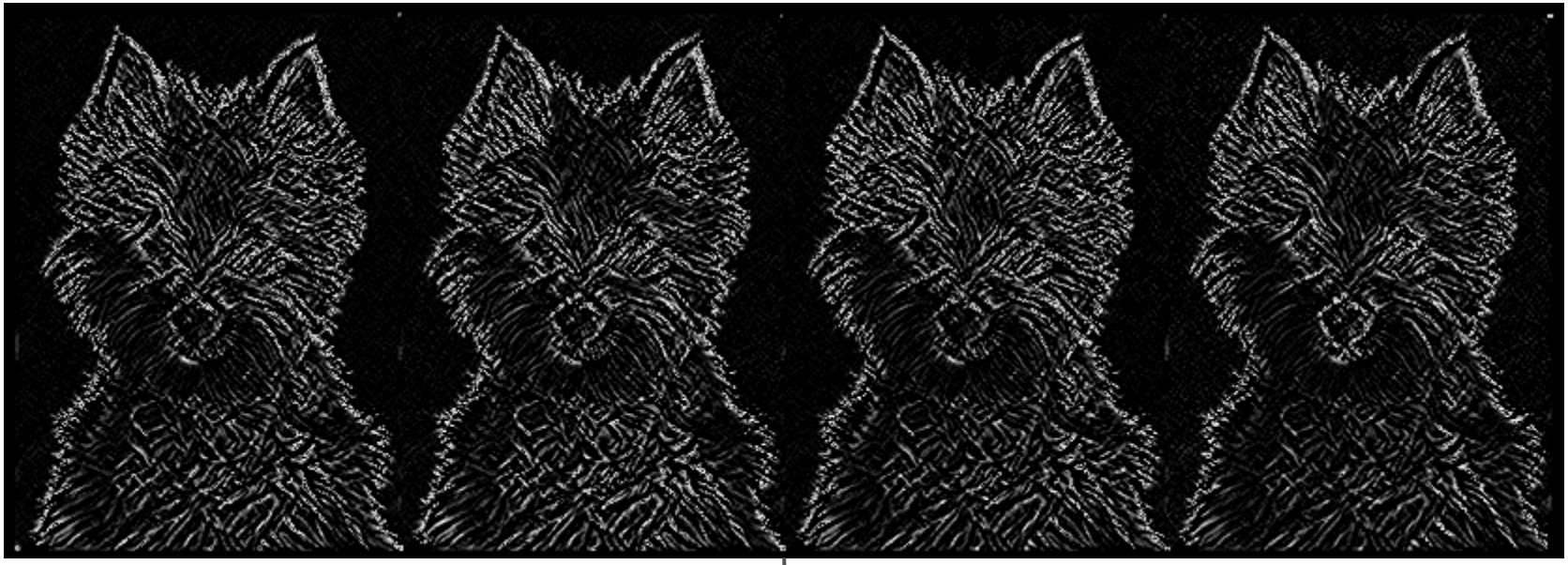
Histogram of Oriented Gradients

Color can be deceiving

But image gradients are not!



But image gradients are not!



Histogram?

Histogram of Oriented Gradients (HOG)

Histograms of Oriented Gradients for Human Detection

Navneet Dalal and Bill Triggs
INRIA Rhône-Alps, 655 avenue de l'Europe, Montbonnot 38334, France
{Navneet.Dalal,Bill.Triggs}@inrialpes.fr, http://lear.inrialpes.fr

Abstract

We study the question of feature sets for robust visual object recognition, adopting linear SVM based human detection as a test case. After reviewing existing edge and gradient based descriptors, we show experimentally that grids of Histograms of Oriented Gradient (HOG) descriptors significantly outperform existing feature sets for human detection. We study the influence of each stage of the computation on performance, concluding that fine-scale gradients, fine orientation binning, relatively coarse spatial binning, and high-quality local contrast normalization in overlapping descriptor blocks are all important for good results. The new approach gives near-perfect separation on the original MIT pedestrian database, so we introduce a more challenging dataset containing over 1800 annotated human images with a large range of pose variations and backgrounds.

We briefly discuss previous work on human detection in §2, give an overview of our method §3, describe our data sets in §4 and give a detailed description and experimental evaluation of each stage of the process in §5–6. The main conclusions are summarized in §7.

2 Previous Work

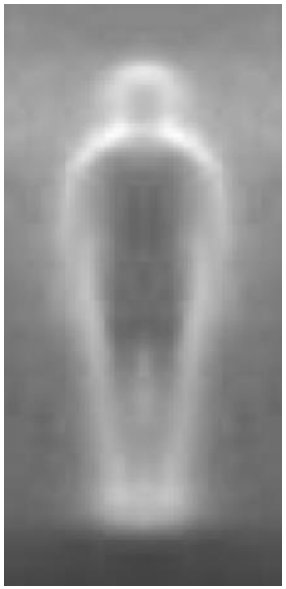
There is an extensive literature on object detection, but here we mention just a few relevant papers on human detection [18, 17, 22, 16, 20]. See [6] for a survey. Papageorgiou *et al* [18] describe a pedestrian detector based on a polynomial SVM using rectified Haar wavelets as input descriptors, with a parts (subwindow) based variant in [17]. Depoortere *et al* give an optimized version of this [2]. Gavrila & Philomen [8] take a more direct approach, extracting edge images and matching them to a set of learned exemplars using chamfer

HOG: intuition

[Dalal and Triggs, 2005]

HOG: intuition

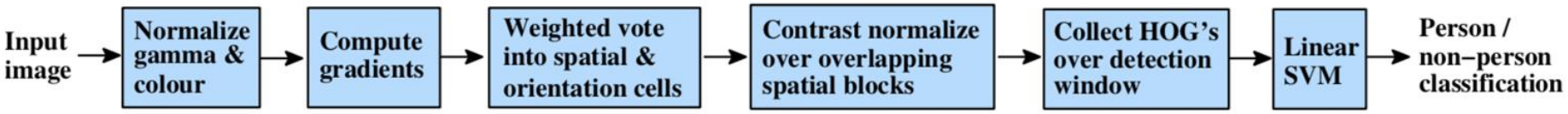
[Dalal and Triggs, 2005]



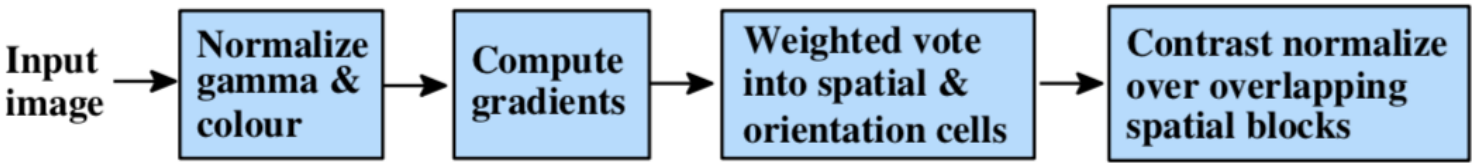
HOG: intuition

[Dalal and Triggs, 2005]

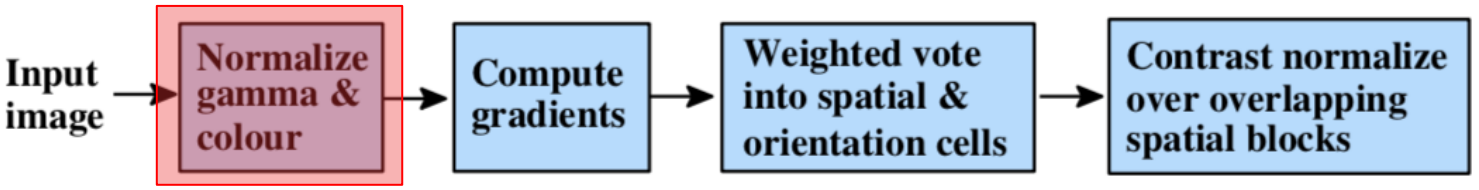
HOG: pipeline

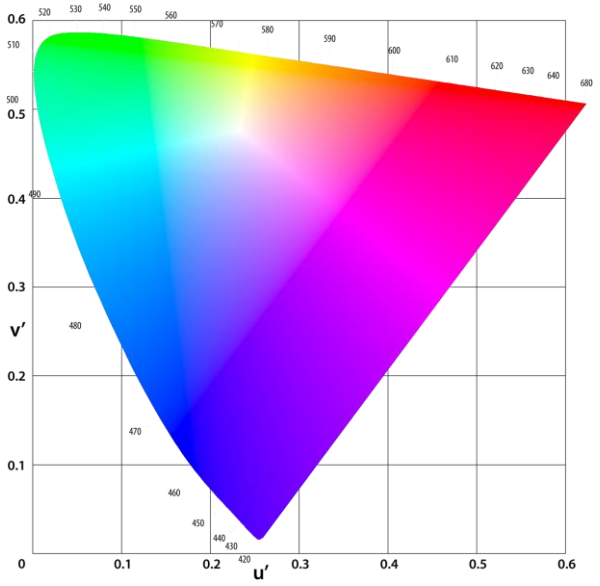


HOG: pipeline

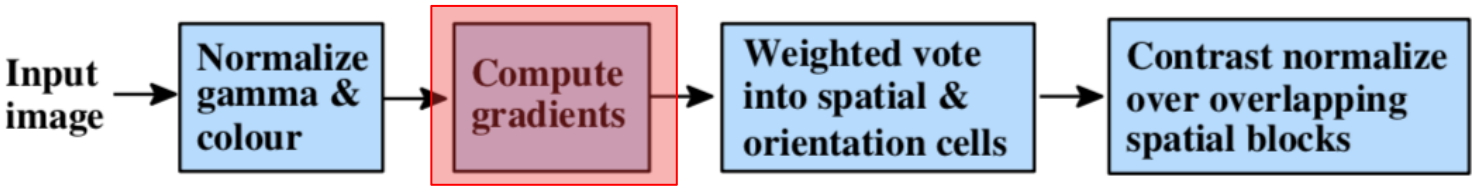


HOG: pipeline





HOG: pipeline



Images from Rafel Cosman and Tao Wang's slides

Linear filter examples

Original

$$*\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} =$$

Identical image

Source: D. Lowe

Linear filter examples

Original

$$*\begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} =$$

Shifted left
By 1 pixel

Source: D. Lowe

Linear filter examples

Original

$$* \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} =$$

Blur (with a mean filter)

Source: D. Lowe

The Sobel operator

- Common approximation of derivative of Gaussian

$\frac{1}{8}$

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

s_x

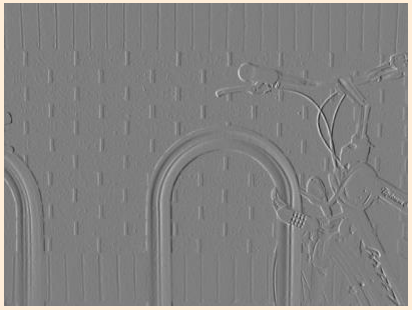
$\frac{1}{8}$

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

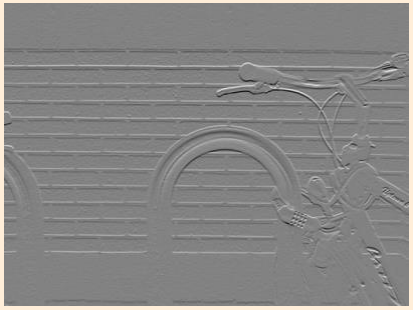
s_y

- The standard defn. of the Sobel operator omits the 1/8 term
  - doesn't make a difference for edge detection
  - the 1/8 term **is** needed to get the right gradient value

The Sobel operator



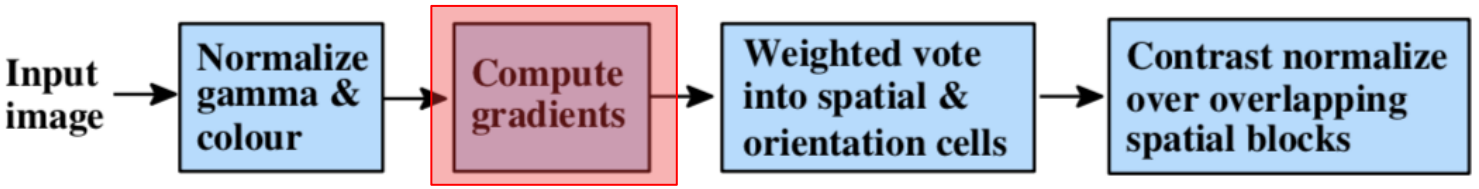
s_x

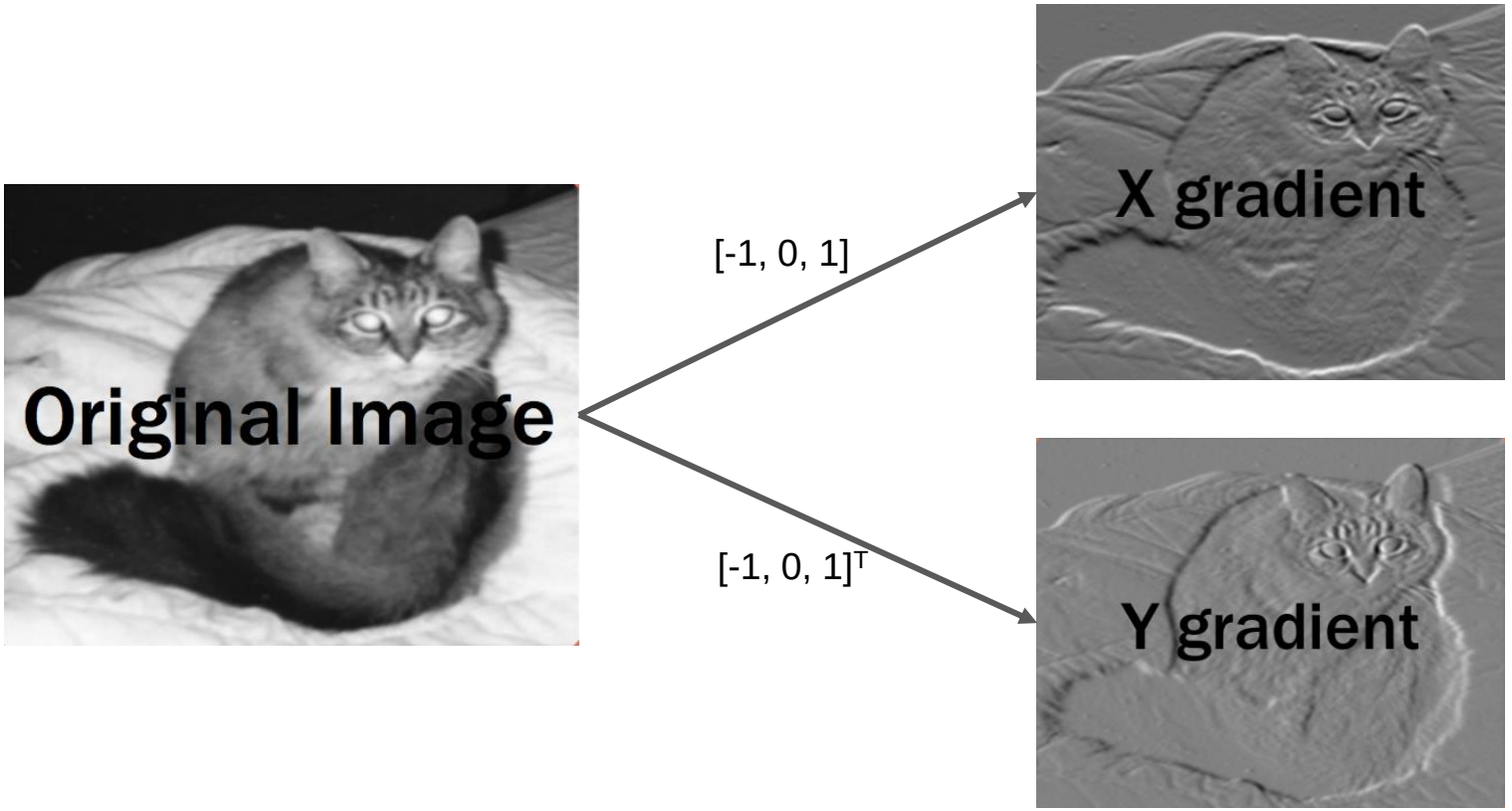


s_y

Source: Wikipedia

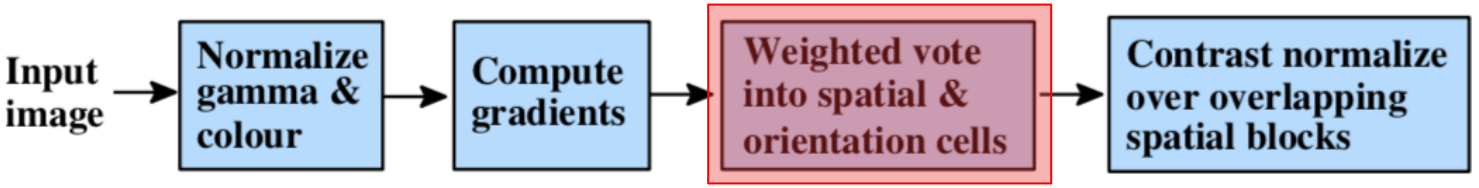
HOG: pipeline

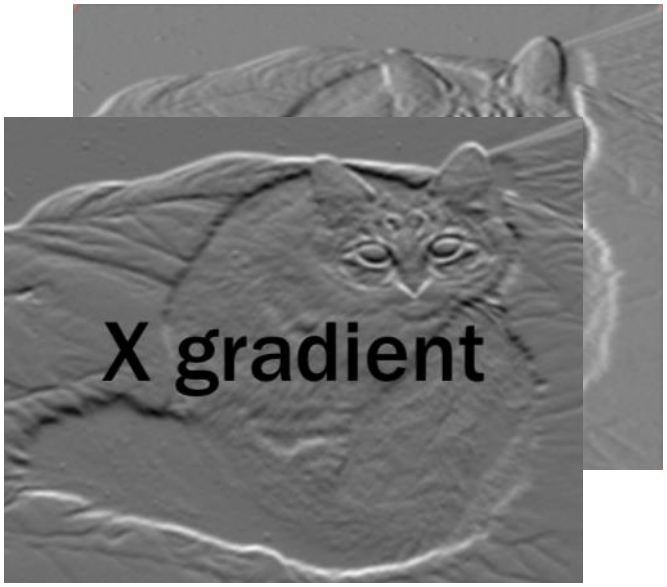




Images from [Rafel Cosman and Tao Wang's slides]

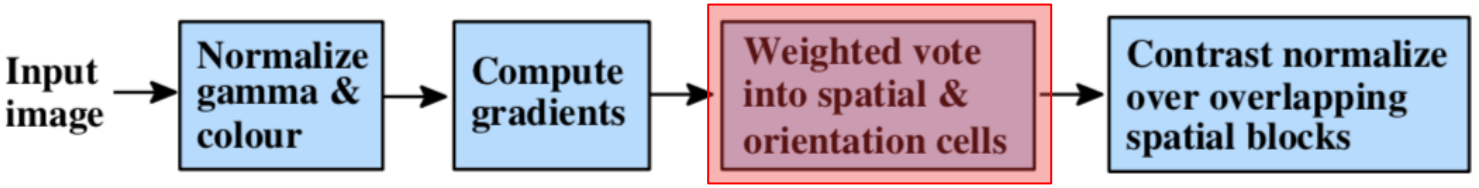
HOG: pipeline

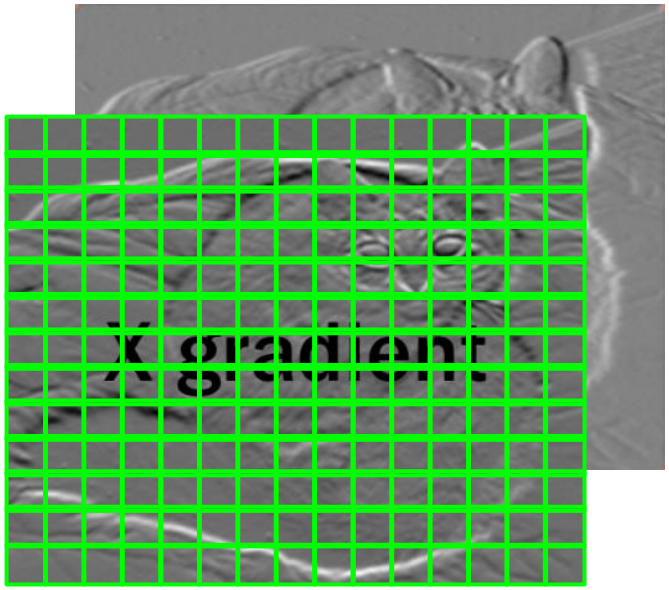




Images from Rafel Cosman and Tao Wang's slides

HOG: pipeline

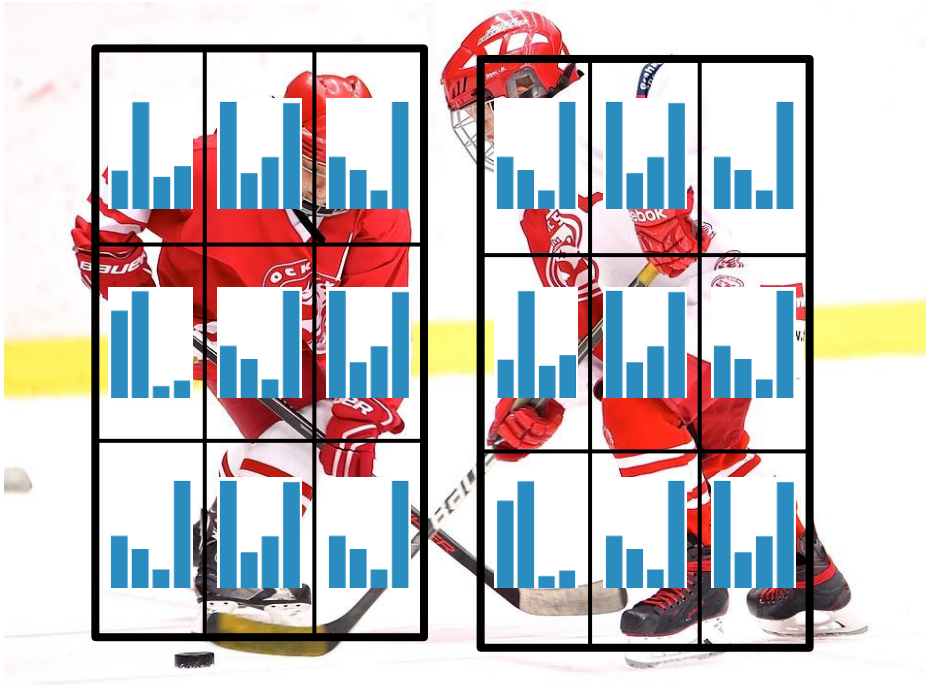




Divide image into 8x8 pixel "cells"

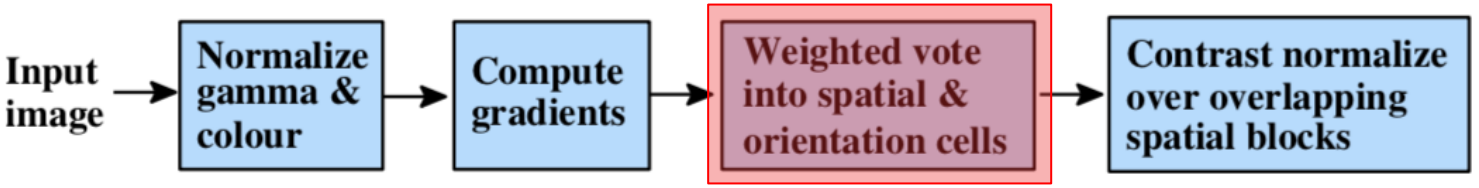
Images from Rafel Cosman and Tao Wang's slides

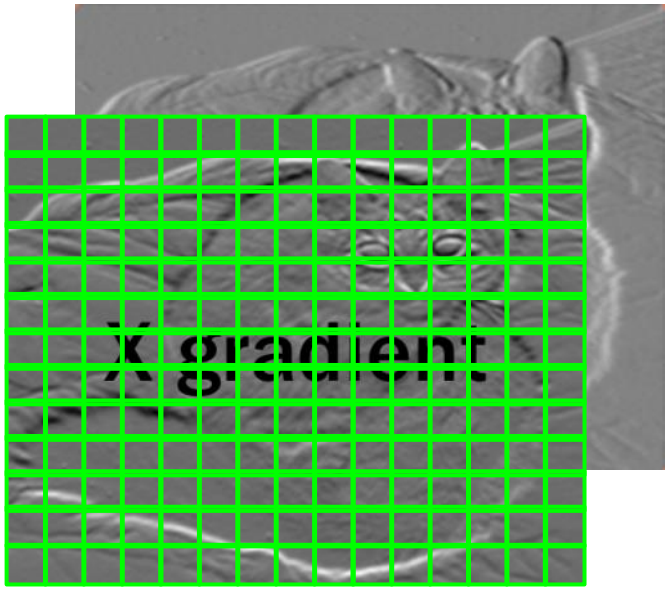
A solution



Divide image into
multiple regions?

HOG: pipeline

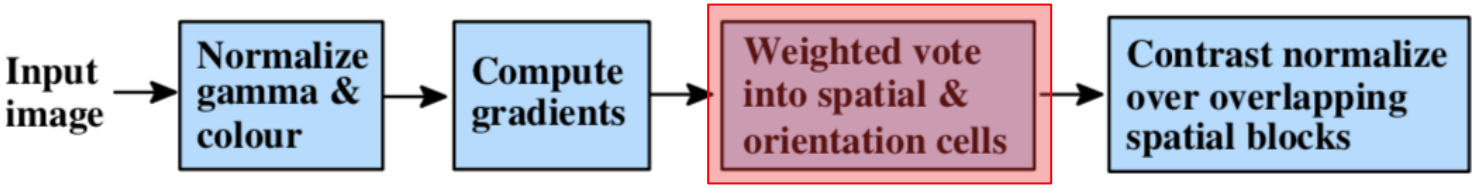


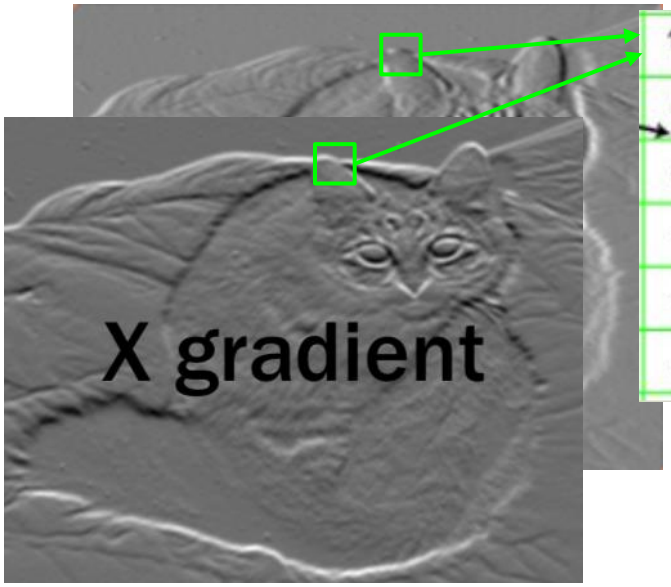


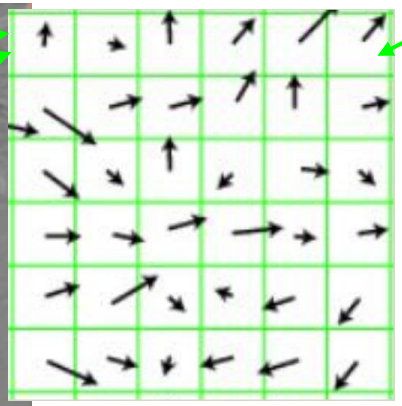
Divide image into 8x8 pixel "cells"

Images from [Rafel Cosman and Tao Wang's slides]

HOG: pipeline





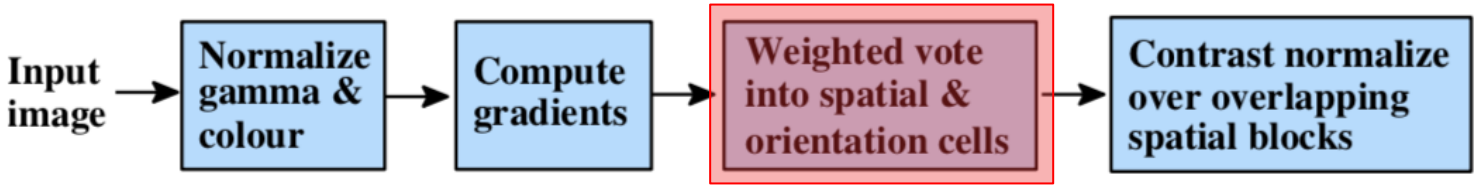


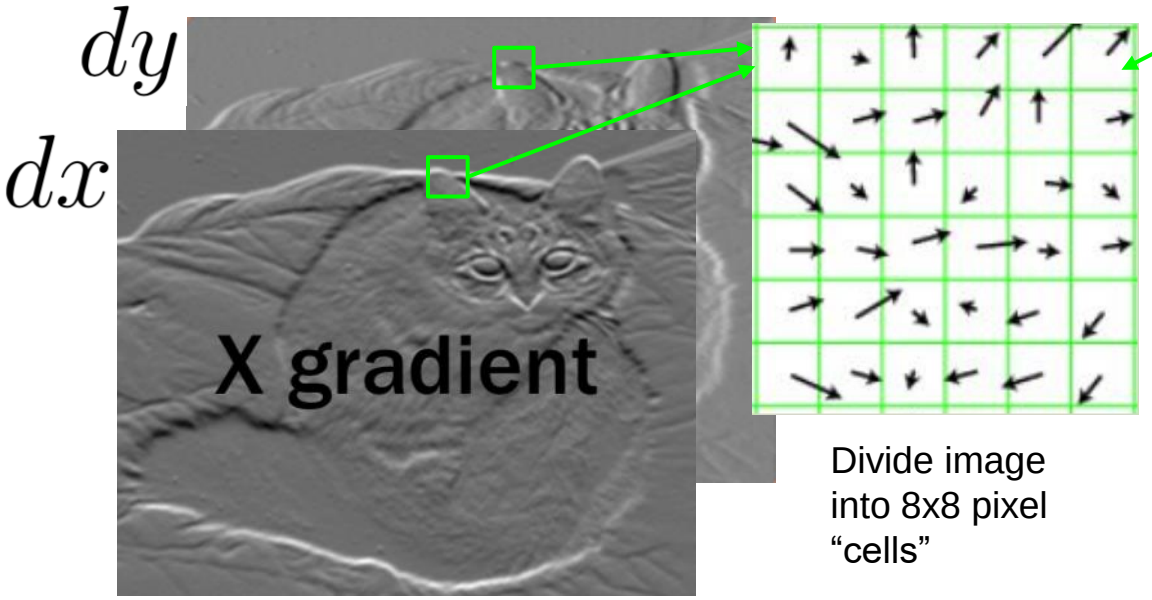
Single pixel

Divide image into 8x8 pixel "cells"

Images from [Rafel Cosman and Tao Wang's slides]

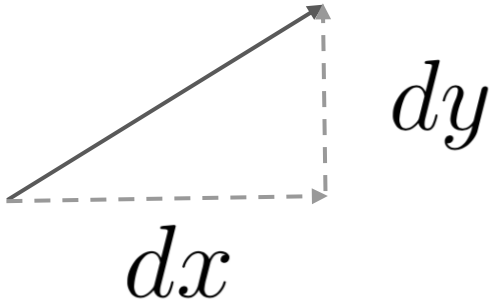
HOG: pipeline





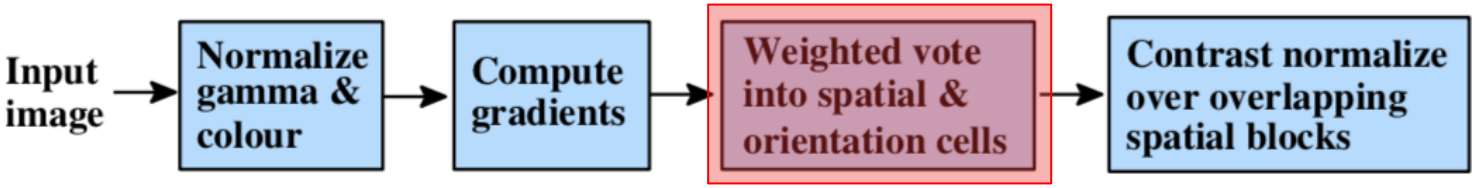
Single pixel

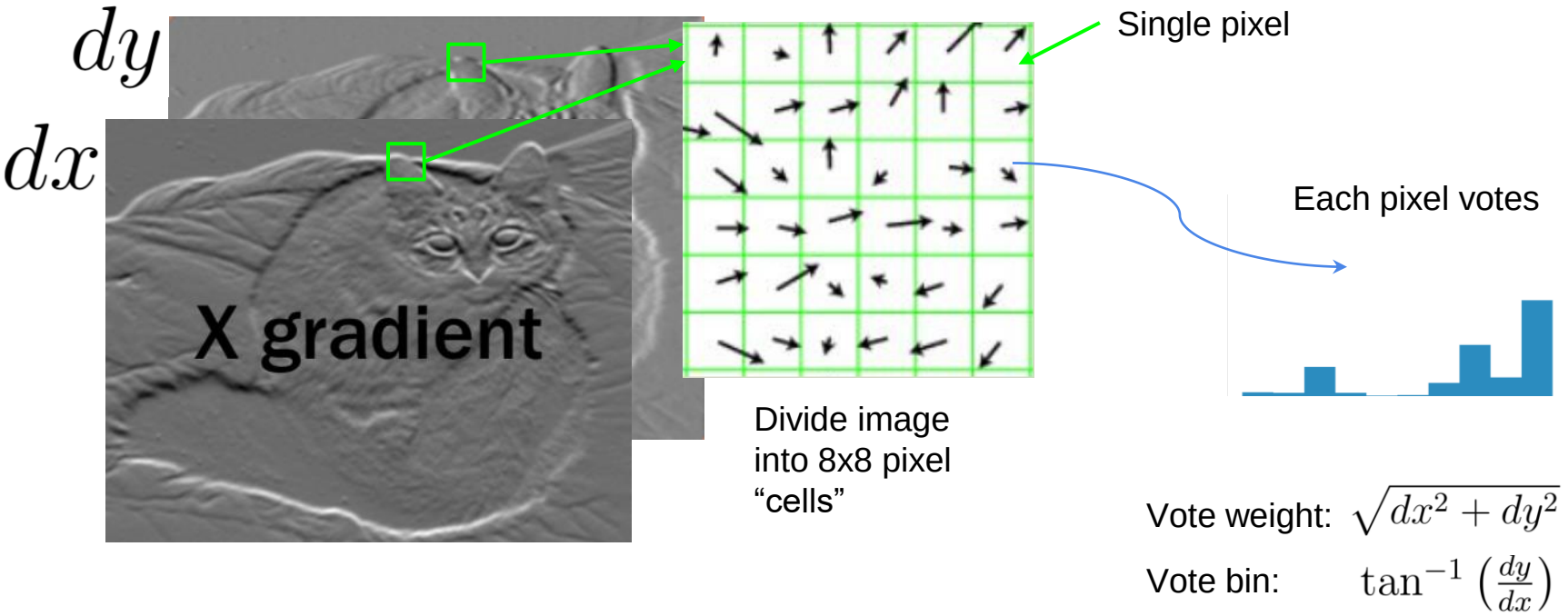
Divide image into 8x8 pixel "cells"



Images from [Rafel Cosman and Tao Wang's slides]

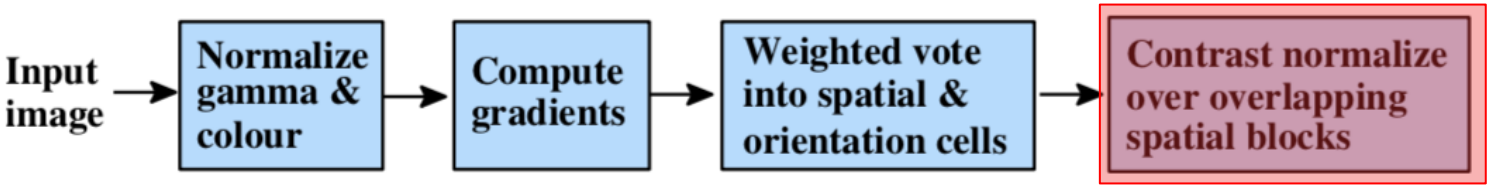
HOG: pipeline

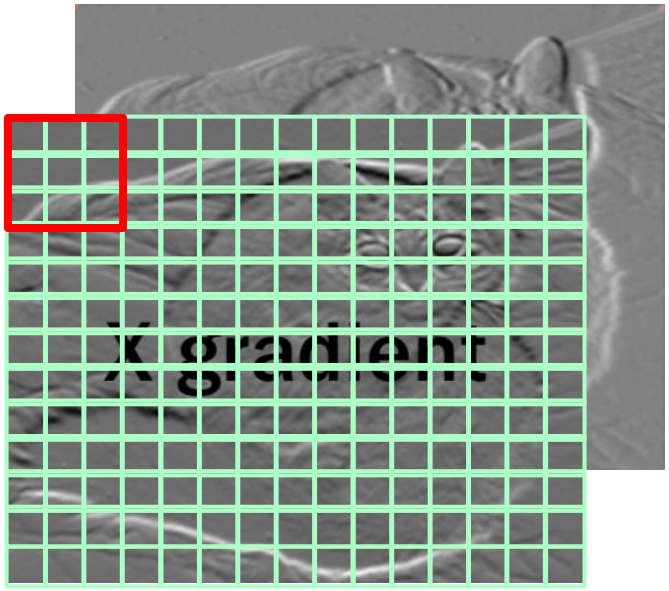




Images from [Rafel Cosman and Tao Wang's slides]

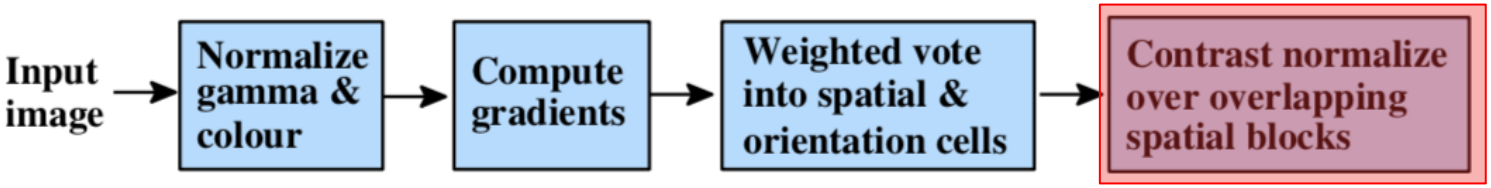
HOG: pipeline

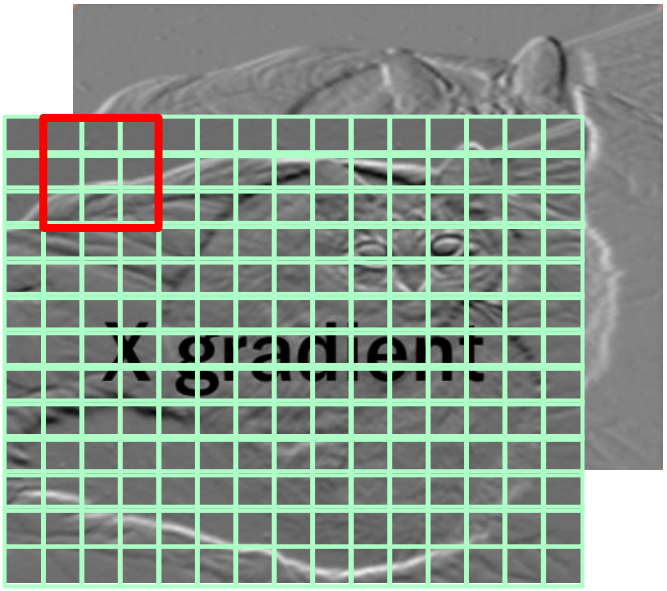




Images from [Rafel Cosman and Tao Wang's slides]

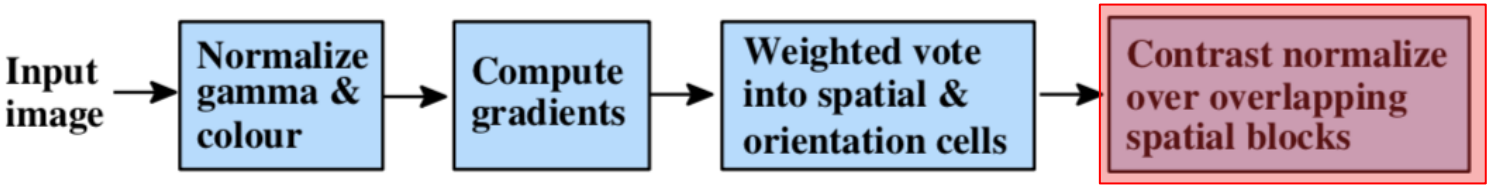
HOG: pipeline

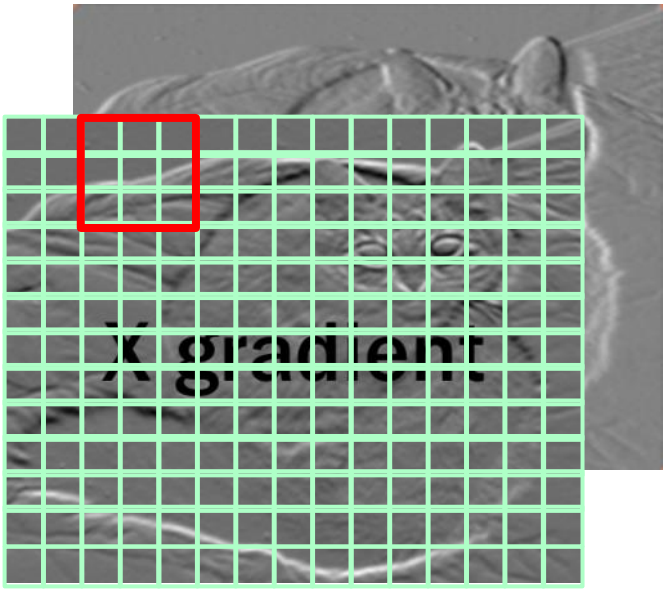




Images from [Rafel Cosman and Tao Wang's slides]

HOG: pipeline

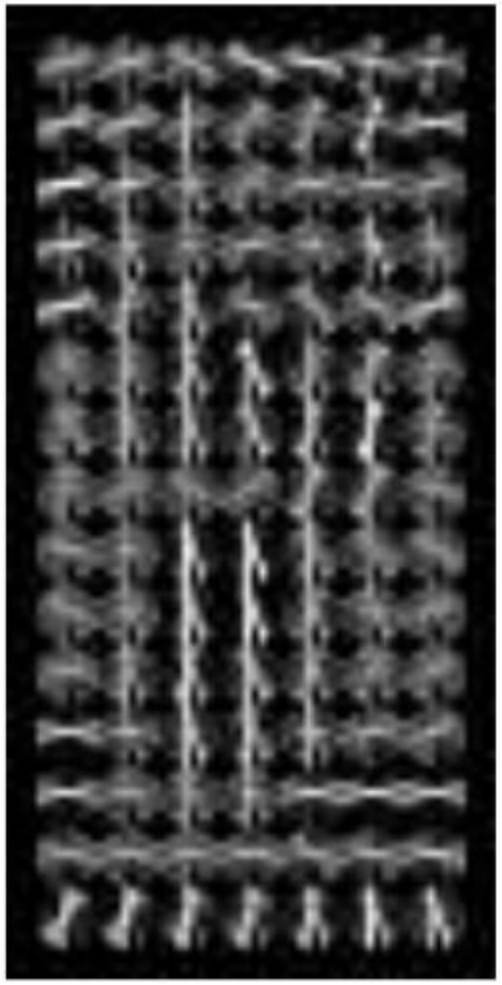




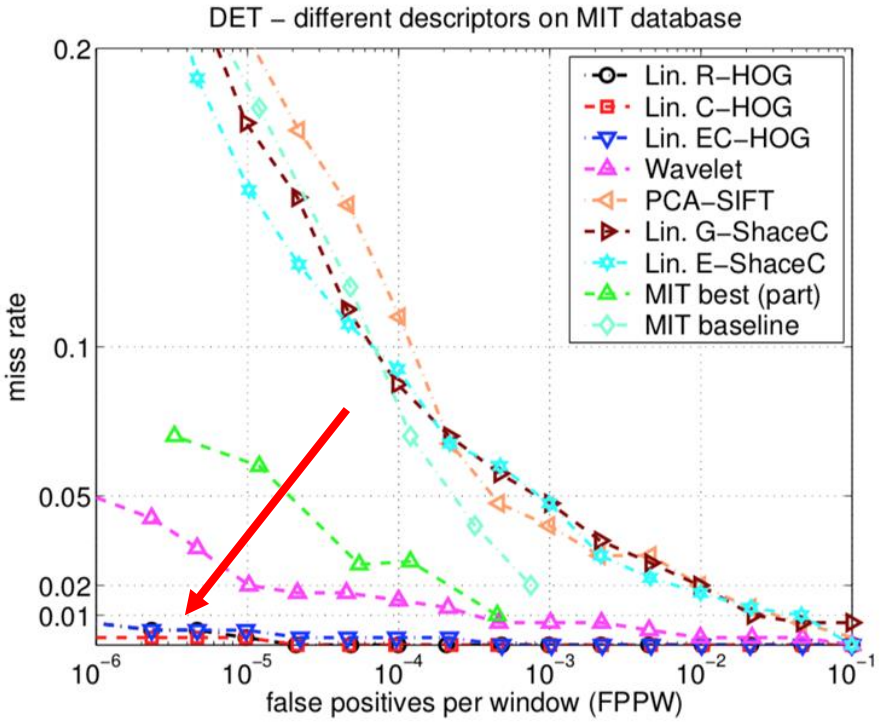
Images from [Rafel Cosman and Tao Wang's slides]

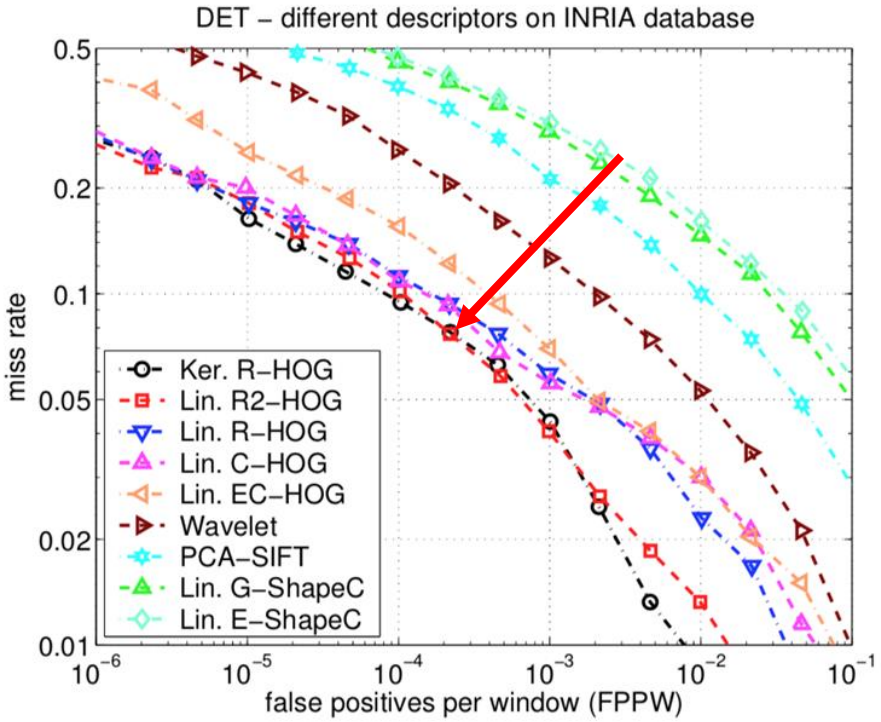
HOG: How it looks

[Dalal and Triggs, 2005]

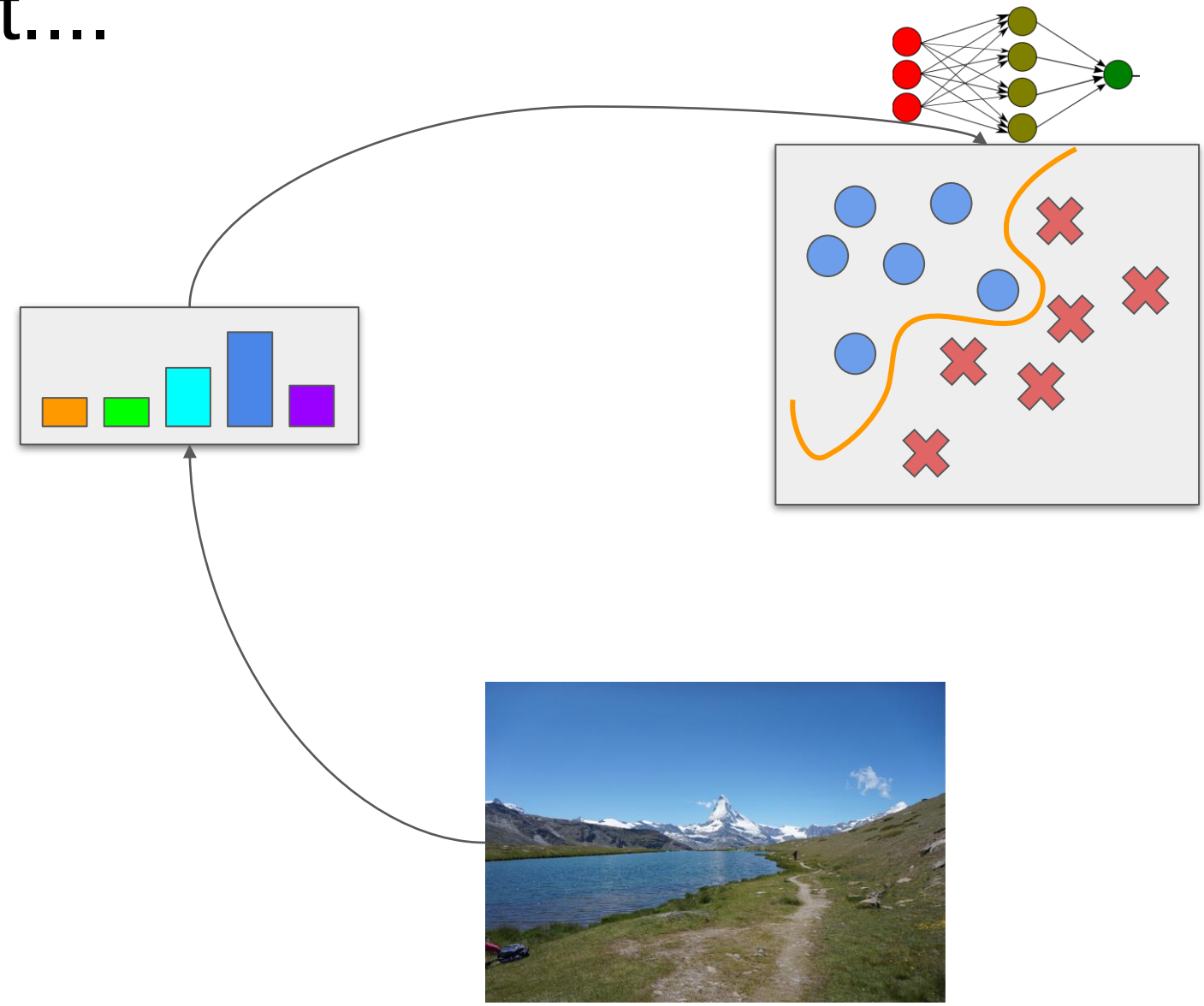


HOG: Performance

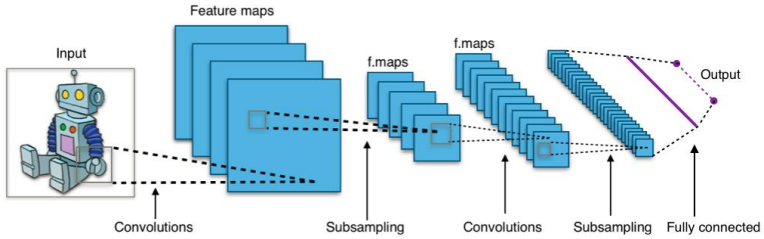


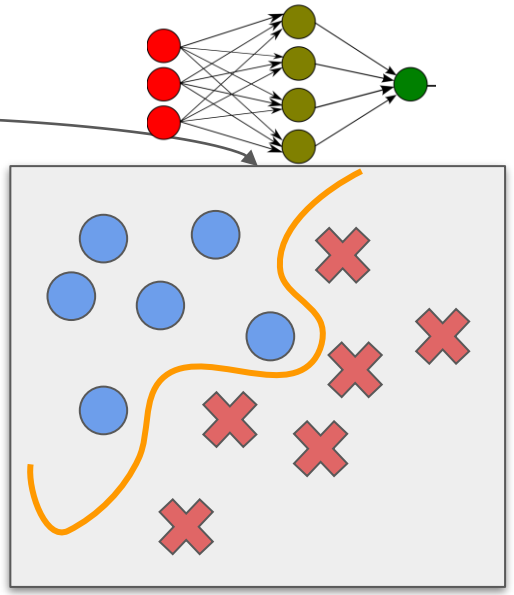


But....



But....


Input
Feature maps
f.maps
f.maps
Output
Convolutions
Subsampling
Convolutions
Subsampling
Fully connected



So far...

- Color spaces
  - The very first stage is also important
- Color/edge histograms
  - Means to abstract images into robust data
- Connection to Deep Learning
  - Although replaced with learned counterparts, the goal is the same