max planck institut
informatik

High Level Computer Vision

Exercise 1 | SS 2019

Max Planck Institute for Informatics

April 15, 2019

Introduction

- TAs:
  - Rakshith Shetty rshetty@mpi-inf.mpg.de [Thu 15:00-16:00]
  - Yang He yang@mpi-inf.mpg.de [Wed 15:00-16:00]
- Mailing list:
  - Announcements about exams, lectures, exercises, and Q&A.
  - Send an email with your name and matriculation number to yang@mpi-inf.mpg.de with **[hlcv-subscribe]** in the subject.
- There will be no tutorial next week.

Introduction

- Grading:
  - 50% Oral exam
  - 50% Exercise sheets [1/2] + project [1/2]
- Exercise sheets:
  - Solutions **must** be submitted in groups (2-3 people).
  - It is strongly encouraged to work in groups of 3.
  - **Send the group member list along with your solutions.**

Image Filtering

- Images may need low-level adjustment such as filtering, in order to enhance image quality (e.g. denoising) or extract useful information (e.g. edges).
- Enhancement: improves contrast.
- Smoothing: removes noise.
- Template matching: detects known patterns.

Image Filtering - Q1 b)

- Gaussian filtering - image smoothing

Original image

Gaussian Blur applied

Apple Motion 4

Image Filtering - Q1 b)

- Gaussian separability – an n dimensional Gaussian convolution is equivalent to n 1-D Gaussian convolutions.

$$h(i,j) = f(i,j) * g(i,j) =$$

$$= \sum_{k=1}^{m}\sum_{l=1}^{n} g(k,l) f(i-k, j-l) =$$

$$= \sum_{k=1}^{m}\sum_{l=1}^{n} e^{-\frac{(k^2+l^2)}{2\sigma^2}} f(i-k, j-l) =$$

$$= \sum_{k=1}^{m} e^{-\frac{k^2}{2\sigma^2}} \left[\underbrace{\sum_{l=1}^{n} e^{-\frac{l^2}{2\sigma^2}} f(i-k, j-l)}_{h'} \right] =$$

$$= \sum_{k=1}^{m} e^{-\frac{k^2}{2\sigma^2}} h'(i-k, j)$$

1-D Gaussian horizontally

1-D Gaussian vertically

Image Filtering - Q1 c)

- Differentiation through convolution.

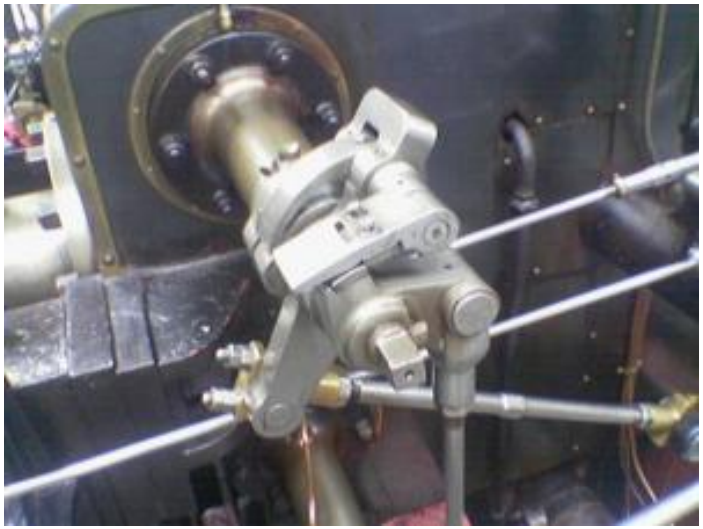
$$\frac{d}{dt}(f * g)(t) = \left(\left(\frac{d}{dt}f\right) * g\right)(t) = \left(f * \left(\frac{d}{dt}g\right)\right)(t)$$

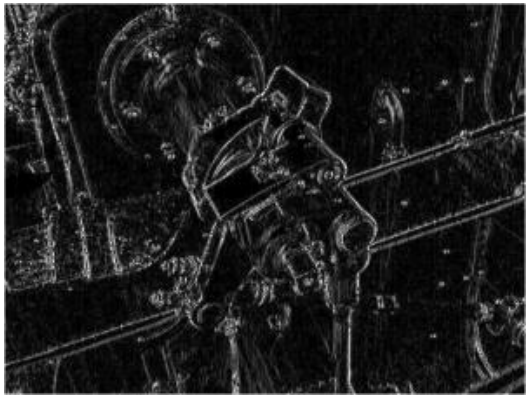
$$\frac{d}{dx}I_\sigma(x,y) = \frac{d}{dx}(I * G_\sigma)(x,y) = (I * \frac{d}{dx}G_\sigma)(x,y)$$

- Gaussian smoothing $\rightarrow$ Differentiation.
- Equivalent to smoothing with derivative of Gaussian.

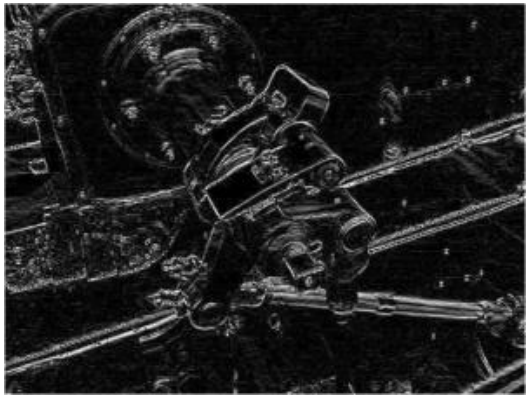
Image Filtering - Q1 c)

- Edge detection using derivative of Gaussian filter:





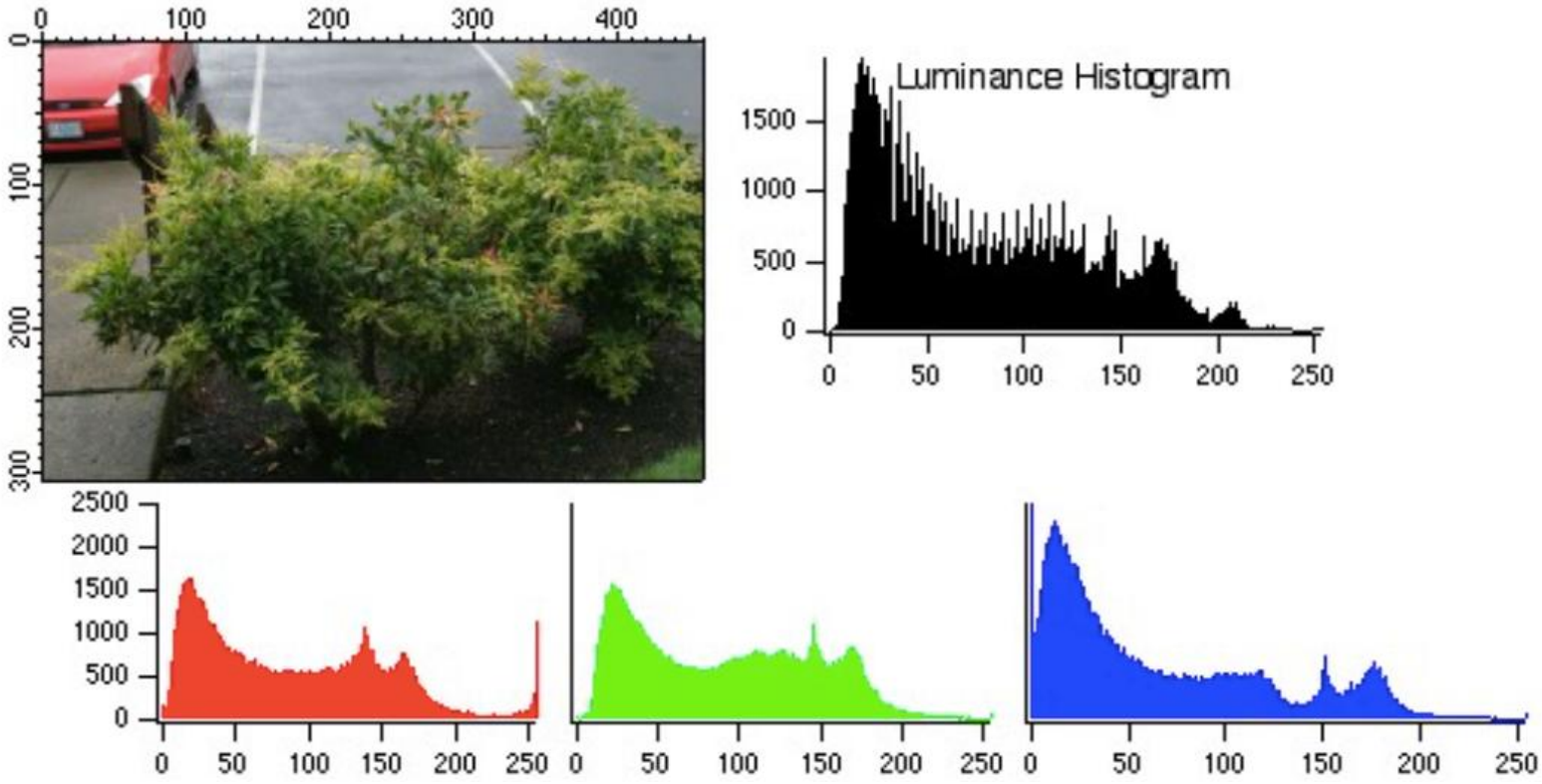
Edges along the x axis



Edges along the y axis

Image Histogram - Q2

- Gives the summary of an image.
- Embeds the image into a "more meaningful" space endowed with some notion of "closeness".



Object Identification - Q3

- Using this space (in this case the histogram space), one can perform several recognition tasks - e.g. identification.

(a) (b)

Performance Evaluation - Q4

- How can we say if method A is better than method B for the same task?

1. Compare a single number - e.g. accuracy (recognition rate), top-k accuracy.
2. Compare curves - e.g. precision-recall curve, ROC curve.

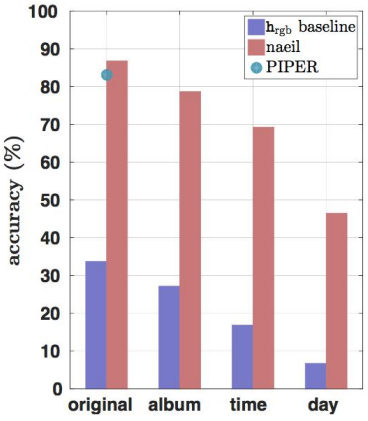
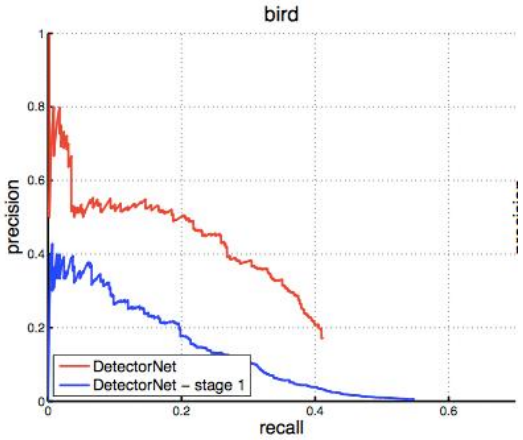
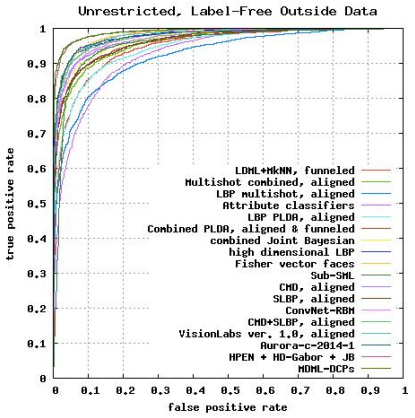


Figure 4. Recognition accuracy across different experimental setups on the test data.

Accuracy (Oh, ICCV'15)



Precision-Recall (Szegedy, NIPS'13)



ROC (LFW Face verification)

Performance Evaluation - Q4

- Accuracy

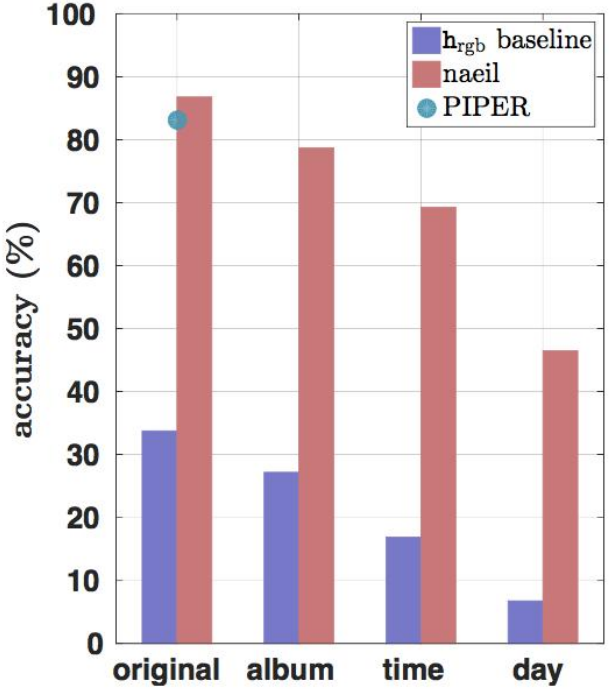


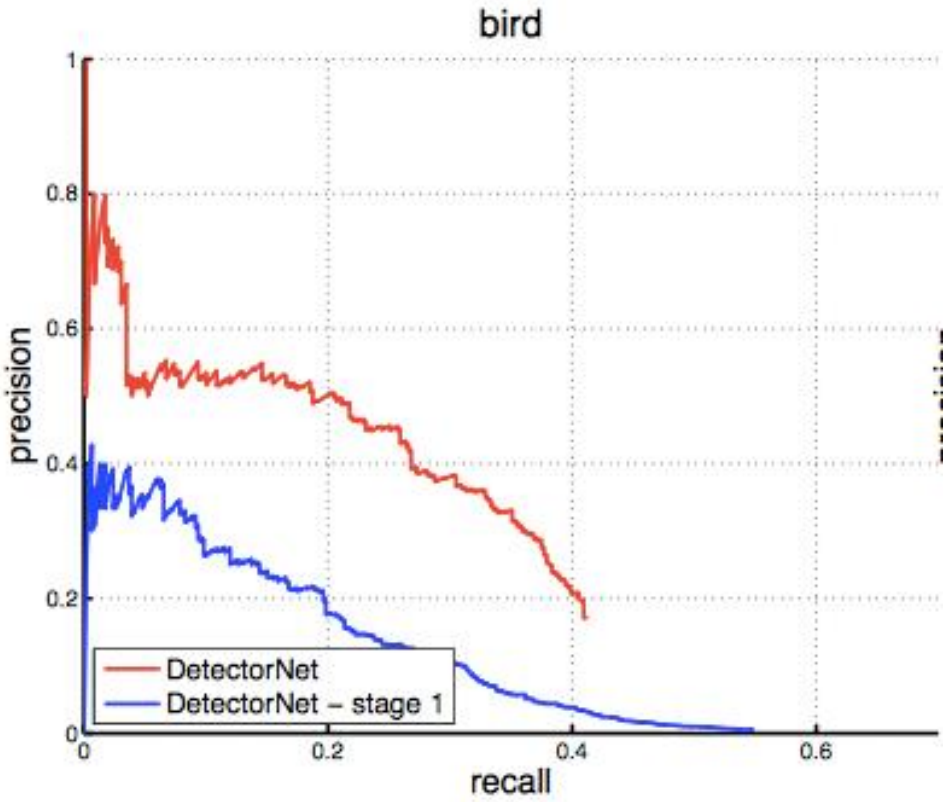
Figure 4. Recognition accuracy across different experimental setups on the test data.

$$\frac{\#\text{Correct Predictions}}{\#\text{Total Examples}}$$

Oh, ICCV'15

Performance Evaluation - Q4

- Precision-recall curve



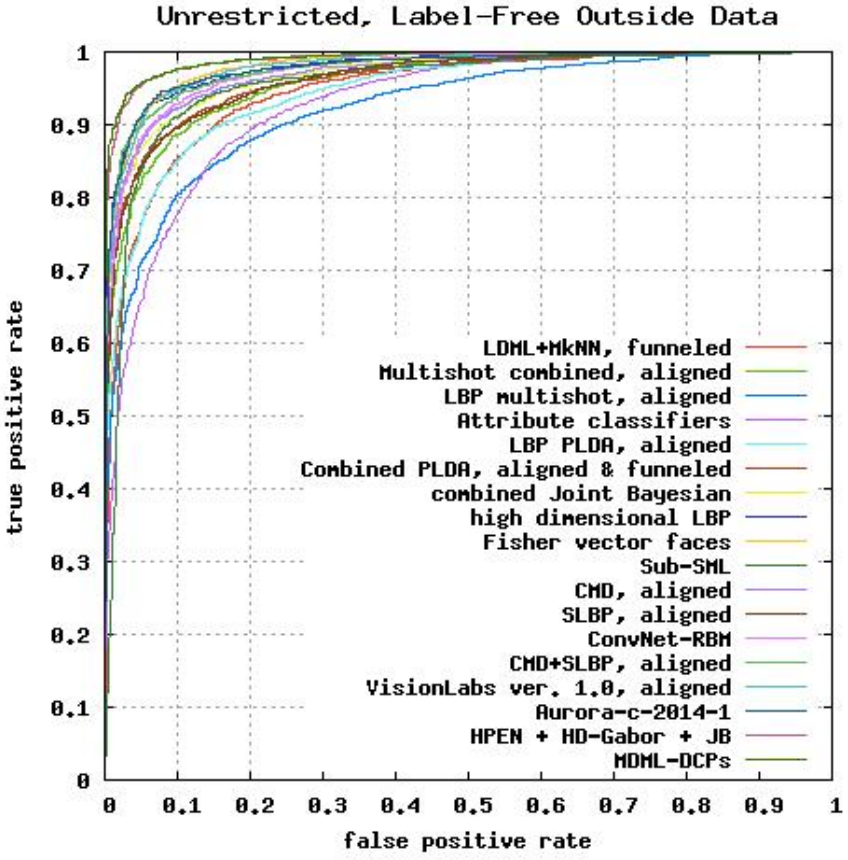
Szegedy, NIPS'13

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Performance Evaluation - Q4

- ROC curve



LFW Face verification

$$TPR = \frac{TP}{P} = \frac{TP}{TP + FN}$$

$$FPR = \frac{FP}{N} = \frac{FP}{TN + FP}$$

Submission

- Friday, April 26th, 23:59.
- Send an email to: yang@mpi-inf.mpg.de
- Please send a single .tar.gz or .zip file containing all the solutions and the report.
- **In the email, include your group members (names and matriculation numbers).**