ECS 174: Computer Vision, Final exam study-guide

1. Linear filters

Image formation

- Light to discrete pixel arrays
- Images in Matlab

Image noise (as motivation for linear filtering)

- Types of noise

Correlation/convolution filtering with linear filters

- 1D and 2D examples
- Definition of correlation
- Implementation detail: handling boundaries
- Smoothing filters
    - Averaging/box filter
    - Gaussian filter
    - Impact of filter width in smoothing
    - Properties
- Definition of convolution
- Examples: predicting linear filter outputs
- Properties of convolution
- Separability

Median filter, an example of a non-linear smoothing filter

2. Edge detection and image gradients

Computing image gradients with convolution

- Partial derivatives in x and y via finite differences
- Other popular finite difference derivative masks
- Properties of the gradient: direction, magnitude
- Impact of noise, counteracting with smoothing
    - Smoothing and differentiating with derivative of Gaussian filters
    - Laplacian of Gaussian filter for edges
    - Impact of smoothing scale on edges found
- Compare: mask properties for smoothing vs. derivatives
- Application with image gradients: seam carving

    - Energy function definition
    - Greedy solution
    - Optimal solution with dynamic programming

3. Edges, contours, and binary image analysis

Edge detection

- Basic pipeline: smooth, enhance, localize
- Thresholding a gradient image for edges
- Canny edge detector
    - Non-maximum suppression
    - Hysteresis
- Low-level edges vs. perceived contours

Chamfer matching: comparing shapes from edges

- Definition of the Chamfer distance
- Computing Chamfer efficiently with the distance transform
    - Definition of the distance transform
    - Efficiently computing the distance transform
    - Using distance transform to get Chamfer distances against template
- Properties of Chamfer matching

Binary image analysis

- Thresholding to create a binary image
- Morphological operators
    - Dilation
    - Erosion

4. Texture

What is texture? What is it useful for?

Psychophysics of texture, "Textons" as primitive units of texture Texture representation

- Detect local patterns
- Describe their statistics
    - Example with gradients
- Compute distances in texture descriptor space
- Scale of a texture pattern

- Filter banks
    - Example application
    - Name that filter bank response
- Applications of texture representations

Non-parametric texture synthesis

- Markov chains, Markov random fields
- Text synthesis example
- Analogy for texture generation in images
- Image quilting extension
    - Minimum error boundary computation

5. Segmentation and grouping

Grouping problems in vision
Inspiration from human perception, Gestalt properties

Bottom-up segmentation via clustering

- Mode finding and mean shift
    - k-means:
        - Algorithm definition
        - Pros and cons
        - Examples of feature spaces: color, intensity,...
        - Texton histograms; clustering for feature space quantization
    - Mean-shift:
        - Algorithm sketch
        - Pros and cons
- Graph-based:
    - Images as graphs
    - Measuring affinity between nodes
    - Minimum cut
    - Normalized cuts; objective definition, pros and cons

Segmentation by cutting graphs

Graph definition

- Visualizing affinity matrices

Min cut and Normalized cut

- Objectives
- Pros and cons

6. Fitting

Hough Transform for lines

- Image space vs. parameter space and concept of voting
- Complete Hough Transform algorithm for lines
- Examples: interpreting Hough space, strengths and weaknesses
- Extensions to the basic approach: esp., using gradient direction
- Computational complexity

Hough Transform for circles

- Parameterization
- Variants depending on (un)known gradient direction, radius

Voting/Hough recap

- Strengths and weaknesses of the approach

Generalized Hough Transform

- Offline training for arbitrary shape model
- Detection of shapes in novel images
- Applicability beyond shape fitting

7. Alignment and fitting

- Feature-based alignment
  - Definition of alignment, place in fitting
  - 2D transformations
    - Parametric transformations
    - Representing with matrices, homogeneous coordinates
  - Affine fit
    - Fitting with least squares given correspondences
    - What are the correspondences?
    - Number of correspondences needed?
    - Recognition by alignment
    - Problem: outliers
  - RANSAC
    - Algorithm definition
    - RANSAC for line fitting (example)
    - Pros and cons

- Homography and image reprojection
  - Perspective projection intuition
  - Connection to image mosaics
  - Solving for homographies
  - Number of correspondences needed?
  - Applying with 2d image warping (forward, inverse)
  - Using homographies to manipulate images: rectification, stitching

8. Local invariant feature detection and description

Overview of main components:

1. Detection: find the interest points
2. Description: extract a descriptor for each one
3. Matching: determine correspondence

Desirable properties: repeatability during detection, distinctiveness during description

Detection

Rotation invariant interest point detection

- Harris corner detection: review of measure, main steps, properties

Scale invariant interest point detection

- Intuition behind automatic scale selection
- Laplacian of Gaussian filter: 1d, 2d
- Characteristic scale selection
- Interest point detection across an image
- Laplacian approximated as DoG in practice

Description

Desired invariance properties: geometric and photometric transformations

Simplest solution: SSD on patch intensity

SIFT descriptor

- Definition
- Rotation invariance

Matching

- Generating candidate matches
- Eliminating ambiguous matches
- Robust alignment pipeline using detection, description, and matching.
- Applications

9. Indexing local features and applications to visual search

- Visual words: Quantizing local descriptor space
  - Connection to textons
  - Issues in visual vocabulary formation
- Bag of words representation and matching
  - Comparing bags of words
  - Tf-idf weighting
  - Scoring retrieval results with precision recall
  - Pros and cons

10. Instance recognition

- Motivation: visual search

- Visual words
  - Feature space quantization (recap)
  - Indexing with inverted file
  - Issues in visual vocabulary formation
  - Bag of words representation and comparisons
  - Vocabulary trees for large vocabularies
  - Pros and cons of visual words

- Spatial verification strategies, using affine transformation
  - RANSAC
  - Example applications, evaluating retrieval results with precision recall

- More text retrieval influence
  - Tf-idf weighting
  - Query expansion

11. Generic category recognition

Introduction to the problem: applications, challenges, state-of-the-art
Basic pipeline of category recognition techniques
Supervised classification

- Definition
- Example with skin color detection

12. Window-based models for generic object detection

Basic pipeline for generic category recognition with a window-based representation

- Representation choice: pixels, gradients, colors
- Discriminative classifier choice
- Sliding windows to generate candidates
- Classifier to score candidates

Boosting classification algorithm

- Intuition
- Weights on training samples
- Training process
- Pros and cons

Viola-Jones face detector: a prime example of window-based object detection

- Overview
- Feature definition: rectangular features, integral images
- Selecting discriminative features among all candidates with Adaboost
- Attentional cascade of classifiers
- Summary of detector pipeline
- Example results

Strengths and weaknesses of window-based detection paradigm

13. Discriminative classifiers for image recognition

- Nearest neighbors and k-nearest neighbors
    - Definition of algorithm
    - Example for scene/location recognition, global "Gist" descriptor
    - Pros and cons

- Support vector machines (SVM)
    - Large margin motivation
    - Definition of algorithm for linear case, solution
    - Non-linear SVMs and the kernel trick
    - Multi-class classification via binary SVMs
    - Example for gender classification
    - Pros and cons

14. Deep learning

- Linear perceptron
- Simple backpropagation example
- Multi-layer neural network
    - Neural network properties
    - Detection (testing) process
- Convolutional neural network
    - Convolution, Non-linearity, spatial pooling, feature maps

15. Deep neural networks II

- Why (convolutional) neural networks?
- Neural network definition
    - *Nonlinear* classifier
    - Can approximate any continuous function to arbitrary accuracy given sufficiently many hidden units
    - Activations

    - Nonlinear activation function *h* (e.g. sigmoid, ReLU)
- Multilayer networks
    - Cascade neurons together
    - Output from one layer is the input to the next
    - Each layer has its own sets of weights
    - Depth = power (usually)
- How do we train them?
    - The goal is to iteratively find a set of weights that allow the activations/outputs to match the desired output
    - minimize a loss function
    - loss function quantifies the agreement between the predicted scores and GT labels
    - Classification goal; Hinge loss, Softmax (cross-entropy)
- How to minimize the loss function?
    - Loss gradients: how the loss changes as a function of the weights
    - We want to change the weights in such a way that makes the loss decrease as fast as possible
- Gradient descent
    - Update weights iteratively
    - Mini-batch gradient descent
    - Cycle through all training examples multiple times
    - Epoch
- Learning rate selection
- Gradient descent in multi-layer nets
    - How to update the weights at all layers?
    - Backpropagation
    - *computational graphs*