LIDAR Feature Extraction Continued

Noel Welsh

02 December 2010

Announcements

- None

Recap

- We are looking at the feature extraction method of [Bosse and Zlot(2009)].
- Last lecture we went over the three methods they used to find feature location (x and y coordinates) and orientation.
- Today we're going to go over the feature representation and how to construct a classifier given features.

Feature Finding

- Three methods
  - Segment clusters
  - Curvature clusters
  - Mean shift clusters

Feature Orientation

- Weighted histograms of normals

Feature Representation

- Now we have found a number of features
- We need a compact way to represent them
- Consider six different methods
  - *Normal orientation histogram grid*
  - Orientation and projection histograms
  - Hough transfrom peaks
  - Gestalt encoding
  - Shape context
  - *Moment grid*
- Normal orientation histogram and moment grid perform best in their experiments.
- Let's discuss these two representations then turn to how we can build and evaluate a place detector using them.

Normal Orientation Histogram Grid

- Based on SIFT features [Lowe(2004)]
- Algorithm sketch:
  - Place grid on scan points
  - Compute histogram of normal vectors within each grid cell
    - Points are weighted by distance from center of cell
    - Points also contribute weight to neighbouring cells
  - Normalise histogram by sum of all weights
- Resulting descriptor is then the normalised histogram heights for all the cells in the grid
- Gives a crude measure of the shape of the feature

Moment Grid

- Like normal orientation histogram grid stores a measure of shape of points lying in a grid cell
- Implementation uses two different sized grids (2m $\times$ 2m and 3m $\times$ 3m) with points weighted by distance from the cell centre
- The features this time are *moments* of the points in the cells

Moments

- What are moments?
- Derived from the concepts in physics, give a measure of shape
- First moment is the mean (or expectation, or average) μ
- Higher moments usually measured about the mean

$$\mu_k = \mathbb{E}[(X - \mu)^k] \tag{1}$$

  for some random variable X
- Also called *central moments*, and usually normalised by σ^k to obtain a dimensionless quantity
- Second central moment is the variance, measures "width" of the distribution
- Third central moment, *skewness*, measures assymmetry
- Fourth central moment, *kurtosis*, measures how "peaked" a distribution is compared to a normal distribution with the same variance

Moment Grid Details

- The eight specific features used are given in the paper. In summary uses:
  - The square-root of the weight
  - The means and variances in the x and y directions
  - Weighted normalised sin and cos of point orientations
- Basically this gives:
  - The Centre of points in the cell
  - An ellipse around the points (variance). Thus can tell rough shape
  - The average orientation of the points in the cell

Building a Classifier

- Ok, we have features.
- How do we use them to answer “have I been here before?”
- How do we tell if our features are any good?

From Features to Dictionaries

- The method is the paper for building a classifier is quite complex, and specific to their features
- We're going to look at a widely used method of constructing a classifier given features – a *dictionary* based method
- Then we'll briefly review the method in the paper

Dictionaries

- What is a dictionary?
  - A book full of words that form a vocabulary
- We want a vocabulary of features to describe our observations. Each feature is a word
- Features are noisy and numerous. We would like to represent a group of similar features by an exemplar. This reduces the number of features we have to consider and make us more robust to noise.

Dictionaries Cont...

- The standard dictionary or vocabulary or bag-of-words method is:
  - Extract features from a training set
  - Cluster features to obtain a smaller number of “words”. This makes up the vocabulary in our dictionary
  - When we see a feature “in the wild”, we replace it with the cluster exemplar that best represents it
  - We then use some method (to be described shortly) to go from words to answering “have I been here before?”

FAB-MAP

- FAB-MAP [Cummins and Newman(2009), Cummins and Newman(2008)] is a good example of a dictionary-based method
- Represents images with SURF [Bay et al.(2008)Bay, Ess, Tuytelaars, and Gool] features (SURF features are similar to SIFT)
- Use approximate *k*-means clustering to select words
- Vocabulary as large as 100'000 words
- Results on image sequence about 1000km long ($\approx$$100'000 images)
- FAB-MAP is worth reading about, but we don't have time to go into details

From Words to Probabilities

- Every time we observe a scene we end up with a number of words/features
- How do we go from words to probabilities?
- *k*-nearest neighbours (KNN)
- Bag-of-words probability models
- Fancier stuff (e.g. FAB-MAP)

k-nearest neighbours

- *k*-nearest neighbours is really simple. Given the words we extract from the current scene:
  - Find the *k* examples we've seen previously that are closest to the current data
  - Vote amongst the *k* examples – the place with the highest number of examples wins
  - If there is no clear winner, or no close points perhaps we're in a new place
- Need to define "close". A large part of the Bosse and Zlot paper deals with mapping features into Euclidean space so we can use Euclidean distance
- Can use other methods. E.g. learn a metric, use Hamming distance for binary variables, and so on.

Simple Bag-of-words models

- The bag-of-words model is just the vocabulary model we have been talking about.
- The bag-of-words is the words we happen to observe in the current place.
- Popular term and representation in text classification.
- If we make an independence assumption between words we can use our old friend Naive Bayes

Fancier Stuff

- The Naive Bayes assumption does recognise any dependencies between words. This is bad.
- Can try to represent dependencies.
- Some example methods:
  - FAB-MAP uses a tree structured Bayesian network
  - Topic models (e.g. LDA) represent a place as a mixture of "topics". E.g. a house consists of windows, doors, and wall. Each topic has a probability distribution over words
  - Various extension of topic models to visual and geometric data

Feature Representation Evaluation

- How should we evaluate the feature representations?
- Metric used in [Bosse and Zlot(2009)] is *separation*
- Separation measures the tradeoff between two types of errors
  - Missed detections – not seeing a feature when it is present
  - False alarms – detecting a feature that isn't really there
- Assume we can tune our detector in some way. For example in KNN we can change k or the cutoff distance within which we search.
- Can then find the parameter setting that maximises separation
- Normal orientation histogram grid and moment grid maximise separation in their experiments
- Can use other metrics, such as precision and recall.

Bibliography I

Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool.
SURF: Speeded up robust features.
Computer Vision and Image Understanding (CVIU), 110(3): 346–359, 2008.

Michael Bosse and Robert Zlot.
Keypoint Design and Evaluation for Place Recognition in 2D Lidar Maps.
In *Robotics and Autonomous Systems*, 2009.
URL `http://portal.acm.org/citation.cfm?id=1651935.1652208`.

Bibliography II

Mark Cummins and Paul Newman.
FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance.
The International Journal of Robotics Research, 27(6):647–665, 2008.

Mark Cummins and Paul Newman.
Highly scalable appearance-only SLAM - FAB-MAP 2.0.
In *Proceedings of Robotics: Science and Systems*, Seattle, USA, June 2009.

Bibliography III

D.G. Lowe.
Distinctive image features from scale-invariant keypoints.
International journal of computer vision, 60(2):91–110, 2004.
ISSN 0920-5691.
URL http://www.springerlink.com/index/H4L02691327PX768.pdf.