ECE 696

Shivam Duhan

Semantic Texton Forest Training

- Learn directly from image pixels
- Use learned decision tree as a classifier

Random Decision Forests

- Ensemble of Decision Trees
- Each node has an associated learned class distribution
- Recursively branches according to learned binary function of the feature vector
- Average the class distributions over the leaf nodes reached for all trees.

Features

- Color
- Texture
- Context

Split Functions - Feature Vector

Consider small image patches P of size dxd pixels centered around current pixel.

For single pixel (x, y) in color channel b:

- Value P(x, y, b) = [0, 0, 255, 0 , 0, 255,

For pairs of pixels (x1, y1) and (x2, y2) in possibly different color channels R, G, B:

- Sum P(x1,y1,b1) + P(x2, y2, b2)
- Difference P(x1, y1, b1) - P(x2, y2, b2)
- Absolute Difference |P(x1, y1, b1) - P(x2, y2, b2)

255, 255, 0; 0, 0, 0; 255, 255, 255;

255, 255, 0; 0, 0, 0; 255, 255, 255;

255, 255, 0; 0, 0, 0; 255, 255, 255;

Get images and train. Build typical decision tree.

Color

- RGB values - per pixel
- HSV - linear transformations, pixel level

Texture

Filter responses - need regions of pixels

Gaussian, Laplacian of Gaussian filter banks

Context

- Displacement vector from each pixel to nearby pixels
- Capture local and spatial structure
-

How to use features for decision tree training

- Train decision trees to make decisions based on certain features
- Use same features to construct feature vector of each pixel

What do you need to implement this

- Feature Vector
    - How to construct
- Training loop
    - How to train decision trees
- Evaluation loop
    - Categorize streaming pixels