Structure from Motion

Pinhole Camera: Test
Homogenous Coordinates: (x, y) $\leftrightarrow$ (λx, λy, λ)
Intrinsic Parameters: K-Matrix
Extrinsic Parameters: [R|t]
Projection Matrix: Π = K[R|t]
Essential Matrix: TR, 5° of freedom
Epipolar Geometry: Test

Camera Calibration

Rig: Test

Stereo Matching

Binocular Stereo: Test
Window Search: Test
Markov Random Field: Test
Graph Cut: Test

Image Classification

Cross-Validation: Split training set into n-folds, [0, n-1] for training, [n] for testing

KNN

Hyperparameters: K and Norm(L1 better, reduces background noise)
Pros: Simple
Cons: Expensive(Use PCA), Norm choice
Curse of Dimensionality: Overfitting

Linear Classifier

Foundation in neural networks
Score Function: Map data to class scores
Loss Function Quantifies prediction/ground truth disparity

SVM

SVM: Hinge loss — Softmax: Cross-entropy loss
Multi-Class Loss Data loss + Regularization loss

Softmax Classifier

Provides probabilities for each class
Cross-Entropy Loss: Maximize cross-entropy

AdaBoost

Ensemble method, combine weak(base) learners to form strong learner, robust to overfitting
Weak Learner: Error<50%

Viola-Jones Face Detector

Uses AdaBoost
Haar-like Features: +/- Rectangles
Integral Image: $\Pi_{ij} = \sum i_{x \leq i; y \leq j}$
Classifier Cascade: Each successive strong classifier uses more features, lower threshold to reduce false negative(increase false positive), Every classifier must be positive to be classified as positive

Neural Network

Layers of neurons, each neuron is a linear classifier, each layer is a non-linear transformation
Score Function: Employs activation function
Activation Function: Non-Linear

Pooling Layer

Downsamples spatial dimensions, reduces computation
Max Pooling: Take max of each window to downsample

Fully-Connected Layer

: *Neurons* Connected to every neuron of next layer

Convolutional Layer

Colloquially uses cross-correlation, uses CONV/FC/POOL layers
Output Size(Image$_{N \times N}$, Filter$_{F \times F}$): (N - F)/(stride + 1)
Stride: Jump of filter over image
Padding: To make output same size
Hyperparameters: # filters, filter size, stride, padding

Gradient Descent

Numerical: Slow, Approximate, Easy to Write
Analytic: Fast, Exact, Error Prone

Backpropagation

Forward/Backward API: Forward- Compute operations, save immediates for gradient computation, Backward-

Apple chain rule to compute gradient of loss function

Object Detection

Window-Based: Viola-Jones, Stengths- Simple detection protocol, good feature choices critical, past successes for certain classes, Flaws- High computational complexity, need low false positive rates, not all objects box shaped, assumes fixed viewpoint

Object Proposals

Proposals: Object-like regions
Person Detection: HoG and linear SVM
Dalal-Triggs Method: Sliding Window, HoG, + linear SVM
Deformable Part Model: Star Model- Coarse Root Filter + Higher resolution part filters
Object Hypothesis: Level+Position of the i-th filter

Active Contour

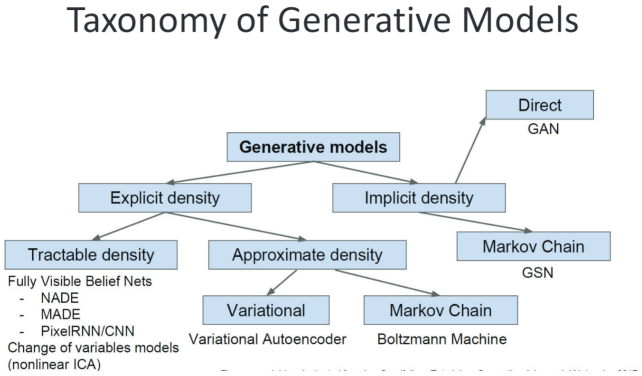
Test: Test

R-CNN

Pre-trained, fine-tuned on PASCAL VOC

Decision Tree

Testing attribute should split training samples into subsets that are as pure as possible
Leaves: Decisions
Information: Reduction in uncertainty
Entropy: Expected amount of information
Information Gain: Information before split - after
Bagging: Reduce variance, committee of trees, samples with replacement
Random Forest Classifier: ex. Microsoft Kinect, efficient, distributed, variable importance, easy to update algorithm, cons: interpretability



Generative Models

Generative Adversarial Networks: Test
Variational Autoencoder: Autoencoder- Reconstruct data

Semantic Segmentation

Markov Random Field: Smooth
Sliding Window: Approach 1
Fully Convolutional: Approach 2

Instance Segmentation

FCN Methods: Divide results from semantic segmentation into individual instances
RCNN Methods: Use segmentation level proposals then train classifier to classify proposals