Video-based Car Surveillance:

License Plate, Make, and Model Recognition

UCSD Master's Thesis 2005

Louka Dlagnekov

Presenter: Brendan Morris

Introduction

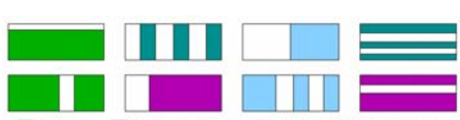
- Low Cost License Plate Recognition System
  - No Need for Expensive Hardware for High Quality Video or Other Sensors
- Extend LPR to More General Make and Model Recognition
  - Database Queries Possible with Partial License Plate and Car Visual Description

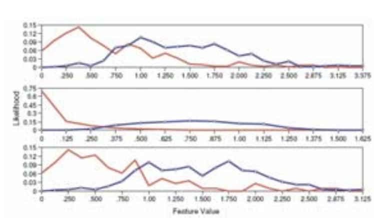
Outline

- License Plate Detection (LPD)
- License Plate Recognition (LPR)
  - Tracking
  - Optical Character Recognition (OCR)
- Make and Model Recognition (MMR)
- Conclusions and Future Work

License Plate Detection

- Window Search Over Entire Frame
  - 3 Different Sized Windows
  - Independent Classifier for Each Size
- Strong Classifier Constructed from Weak Classifiers Via AdaBoost
  - Computationally Simple





AdaBoost

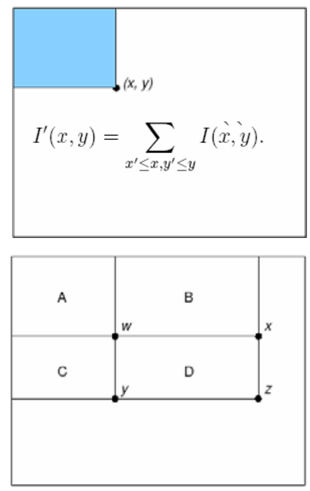
- Adaptive Algorithm Constructs a Strong Classifier as a Combination of Weaker Classifiers
- Build Initial Classifier Model
- Identify Samples not Explained by Model
  - Mis-Classified Samples
- New Model Built Using New Training Set which Includes the Difficult Mis-Classified Samples from the Previous Model

Optimizations

- Fast Detection Rates
  - 640 x 480 Image Size
  - 10 Frames/sec
- Viola and Jones (2001)
  - Integral Images
  - Cascaded Classifiers

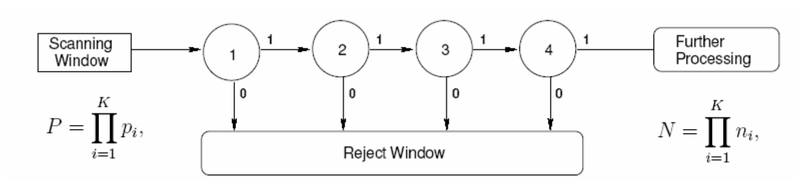
Integral Images

- MN Array Accesses for MxN Array (2400 Simple Classifier)
- Use Rectangular Structure to Reduce Accesses to 4
- Sum of Pixels Above and to Left



$$D = I'(w) + I'(z) - (I'(x) + I'(y)).$$

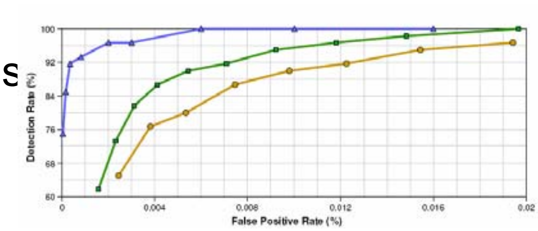
Cascaded Classifiers



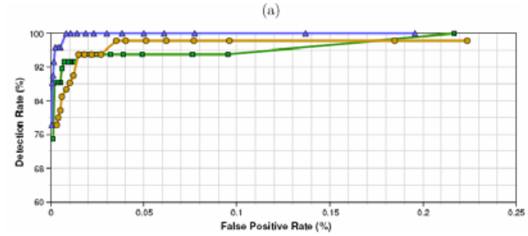
- Group Classifiers into Several Stages in Order of Increasing Complexity
  - Simple Effective Early Classifiers can Reject Most Erroneous Regions
- Train Stages on False Positives of Previous Stage

LPD Results

- Detector Trained on Several Scales
- Many False Positives Come From Other Text in Scene



(a)



License Plate Recognition

- Use Detection Result to Construct Tracks
  - Robust Plate Detection
  - Enforce Track Smoothness Constraints
  - Multiple Detections for Super-Resolution
- Optical Character Recognition (OCR)
  - NCC Template Matching

Super-Resolution

- Multiple Low Res Samples (L_k) Used to Construct Single High Res (H) Image

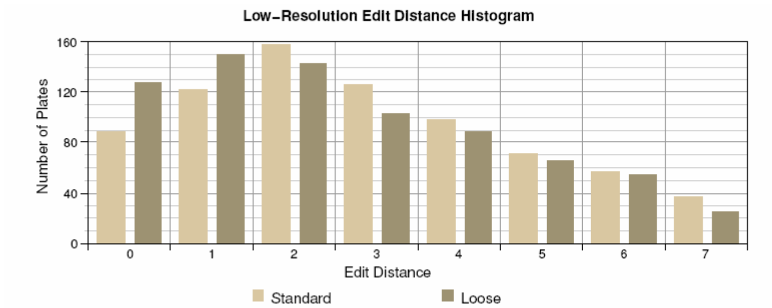
$$\widehat{L}_k(x,y) = S\downarrow(h(x,y) * H(\mathcal{T}_k(x,y))) + \eta(x,y),$$

- Estimate H Given L_k
  - Register Tracks with NCC (Tk)
  - Use Gaussian PSF (h)
  - Additive Gaussian noise (n)
  - Down Sample by 2 or 4 (S)
- Used to Separate License Characters

Super-Resolution Algorithm

- Maximize [redacted]
- MLE
  - No Priors – All Ĥ Equally Likely
- MAP
  - Solved Using Gradient Descent Methods
  - Smoothness Prior $Pr_s(\widehat{H}(x,y)) = c_s e^{-\rho(\widehat{H}(x,y)-\overline{H}(x,y))},$
  - Bi-Modal Prior $Pr_b(\widehat{H}(x,y)) = c_b e^{-(\widehat{H}(x,y)-\mu_0)^2(\widehat{H}(x,y)-\mu_1)^2},$

LPR Results



- Edit (Levenshtein) Distance for Accuracy Measure
  - Loose Measure – Avoids Penalties for Commonly Mistaken Characters {Z, 2}, {B, 8}

Make and Model Recognition

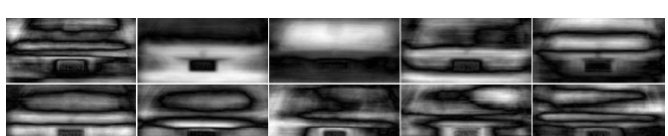
- Selected Car ROI from LPD
  - Placed in Canonical Position
- Compared Different Recognition Algorithms
  - Appearance-Based Methods
    - Eigencars
  - Feature-Based Methods
    - Shape Context Matching
    - SIFT Matching

Eigencars

- Dimensionality Reduction Using Principle Component Analysis (PCA)
  - Car Image (Pixel Intensities) as Feature Vector
- Project Each Car Image to Lower Dimensional Space
  - Classify Match as Closest (L_2 Distance) Database Car

Eigencars Results

- Recognition Rate of 23.7%
  - Recognition Rate of 2.5% for Random Guessing
- Improvements
  - Discard Largest Eigenvalues
    - 44.7 – 47.4% Recognition
  - Fisherface Method



Improved Eigencars

- Using All N Eigencars

- Using N-3 Eigencars

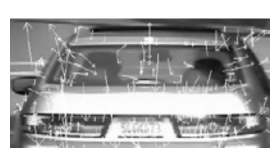
Feature Extraction

- Corner Detectors
  - Harris and Förstner

- Salient Features
  - High Entropy

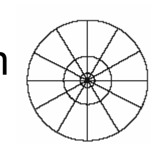
$$\mathcal{H}(s,\mathbf{x}) = \sum_{i=0}^{255} P_{s,\mathbf{x}}(i) \log P_{s,\mathbf{x}}(i).$$

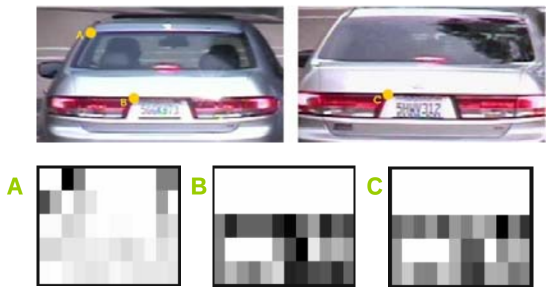
- SIFT Features
  - Scale Invariant Feature Transform

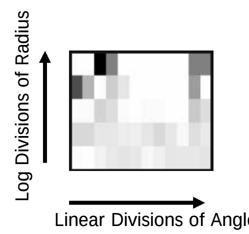


Shape Contexts

- Radial Edge Pixel Count Histogram





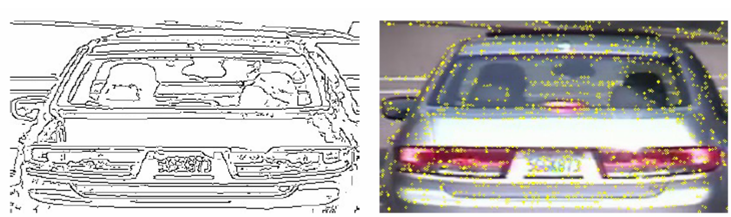


- Usually Compared Using Chi-Squared or L_2 Distance

$$d(\mathbf{h}_i, \mathbf{h}_j) = \sum_{\text{bins } k} \frac{\|\mathbf{h}_i(k) - \mathbf{h}_j(k)\|^2}{\|\mathbf{h}_i(k) + \mathbf{h}_j(k)\|},$$

Shape Context Matching

- For Database Entries, d, and Query Images, q, Take N Random Sample Points of Corresponding Edge Images and Compute Shape Context Around Each Point



(a) (b)

Shape Context Matching

- For Each d
  - For Each Sampled Edge Point, p_q, in q Find Best Matching Point p_d Within a Radius Threshold Using Chi-Squared Distance
  - Create Match Cost as Sum of Distances For Every Correspondence
- Choose d with Lowest Cost as Match

Shape Contest Results

- Descriptor Radius – 35 Pixels
- Sampling Size – N = 400 Points
- 65.8% Recognition
  - 5 x 12 Shape Context
- 63.2% Recognition
  - 9x4 Shape Context

SIFT Features

- 4 Step Procedure
  - Scale-Space Extrema Detection
  - Keypoint Localization
  - Orientation Assignment
  - Descriptor Assignment
- Scale Space

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}.$$

σ – quantized scale factor

SIFT Features

- Keypoint Localization
  - Find Extrema in $D(x, y, \sigma) = L(x, y, k\sigma) - L(x, y, \sigma),$
- Orientation Assignment

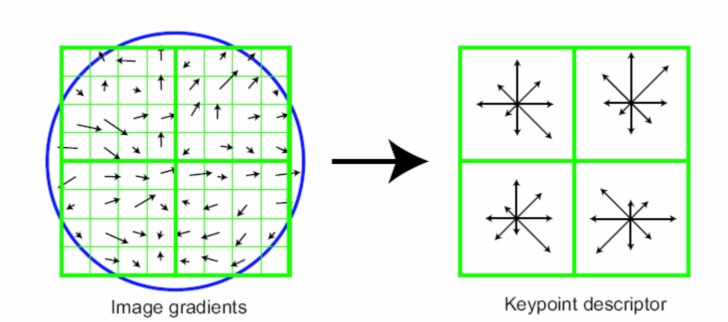
$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2}$$

$$\theta(x,y) = \tan^{-1} \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}$$

- Descriptor Assignment (16 x 8 = 128 Dim)
  - Divide Region Around Keypoint into 16 Symmetric Sub-Regions and Create 8 Orientation Bins

SIFT Descriptor

- Scale and Rotation Invariant
  - σ – Scale Factor
  - Keypoint Orientation



SIFT Matching

- Extract SIFT Features for Each Image d, Database Entry, and q, a Query Image
- For Each d
  - For Each Keypoint, k_q, in q find k_d with Smallest L_2 Distance and is at Least a Factor of α Smaller than Second Nearest Neighbor
  - Count Number of Matched Descriptors
- d with Largest Count as Best Match

SIFT Results

- (After Applying Keypoint Pruning)
- 89.5% Recognition Rate

MMR Summary

Method	Recognition rate
Eigencars using all eigenvectors	23.7%
Eigencars without 3 highest	44.7%
Shape context matching with 9×4 bins	63.2%
Shape context matching with 5×12 bins	65.8%
SIFT matching	89.5%

- Achieved High Recognition Rates
  - Mis-Classifications had Few (<5) Database Examples
- High Recognition at Cost of Computation
  - 30 sec for Shape Context and SIFT vs 0.5 for Eigencars

Conclusion

- Developed Car Recognition Framework Combining LPR and MMR
- Can Be Used in a Query Based Car Surveillance System
- High Recognition Rates
  - Only LPR Currently Real-Time

Future Work

- MMR
  - Speed Up Recognition (Real-Time Application)
    - Group Database into Vehicle Type {SUV, Truck, …}
    - Formulate as Text Retrieval (Sivic and Zisserman)
- Add Color Inference
- Database Query Algorithm Development
- Make and Model 3D Structure