Lateral Detection

Qing Zhong (ETHZ)
Martin Schindler (IMVS, FHNW)
Christoph Stamm (IMVS, FHNW)
Introduction

- A lateral connects a branch sewer to a main sewer
- The goal is to automatically detect the lateral
Pipe Inspection

- Communal sewer networks should be examined periodically because of contamination and water waste
- Robots produce analog/digital videos of sewer pipes
- A time consuming task for human inspectors
- Automatic pipe condition assessment system is needed
Automatic Lateral Detection

- Circular and dark objects with a given radius
- LD is a twofold challenge
  - Distinguish the different objects, i.e., to delineate regions in pipe images corresponding to candidate laterals
  - Discriminate real laterals from false ones in the set of candidate laterals
- Three-step algorithm, preprocessing, segmentation and classification
Three-step algorithm

A: PREPROCESSING
1. Pipe Images
2. Outlining
3. Rearranging
4. Gray Filtering
5. Restoring
6. Crossing
7. Preprocessed Images

B: SEGMENTATION
1. Preprocessed Images
2.1 Gray Morphology
2.2 Anisotropic Diffusion
3. Dilating
4. Normalizing
5. Histogram Analysis
6. Thresholding
7. Binary Image
8. Region Search
9. Region Analysis
10. Candidates

C: CLASSIFICATION
1. Feature Extraction
2.1 Training Set
2.2 Validation Set
3.1.1 KNN
3.1.2 SVM
3.1.5 AdaBoost
4. Optimal Parameters
6. New Pipe Images
7. Feature Extraction
8. Test Set
9. Prediction AdaBoost
10. LATERALS
Preprocessing

- Raw pipe images are quite different
- To make input pipe images as homogenous as possible

![Image showing preprocessing steps: original image, cut and rearranged, gray scaled, resized, cropped]
Segmentation I

- Apply edge preserving filters beforehand
- Gray-scale morphology (closing operator, radius = 5 pixels)
- Anisotropic diffusion, (diffusion coefficient $\exp\left(-\frac{f_i^2}{2\kappa^2}\right)$, step size 0.2 and $\kappa = 15$)
Segmentation II

- Normalization and segmentation

\[ \tilde{I}_n \leftarrow (\tilde{I}_{ROI} - q_{\text{min}}) \cdot \frac{\min(z, q_{\text{max}} - q_{\text{min}})}{q_{\text{max}} - q_{\text{min}}} \]

- Region Analysis

- a) Image after morphology
- b) Image after morphology and binarization
- c) Image after anisotropic diffusion
- d) Image after anisotropic diffusion and binarization
Classification I

- Candidate laterals

- Feature Extraction
  - Geometric properties (Binary image)
  - Color properties (RGB image)
  - Monte Carlo estimation (Gray-scale)
Classification II

- AdaBoost with decision stump as weak classifier

\[ \hat{c}_B(x) = \text{sign} \left( \sum_{b=1}^{B} \alpha_b c_b(x) \right) \]

\[ c_b(x_k, p, \theta_p) = \begin{cases} 
+1 & x_{k,p} > \theta_p \\
-1 & \text{otherwise}
\end{cases} \]

- Tenfold cross validation to estimate the test error

\[ 10^{-1} \sum_{i=1}^{10} |d_i|^{-1} \sum_{k \in d_i} 1\{y_k \neq (\hat{c}_B)_{n-|d_i|}(x_k)\} \]

- The optimal model is found by calculating:

\[ \hat{c}_B^* = \arg \min_{\hat{c}_B \in C} \text{CV} \]
Experiments

- Apply three-step algorithm to about 6,000 scanned images (10,000 meters in length)
- Preprocessing and Segmentation, where preprocessing and segmentation cause about one percent error
- Feature extraction
- Training set (validation set 10-fold CV) and test set
- AdaBoost with decision stumps
- Compare to SVM with Gaussian RBF kernel and 1-NN and 3-NN, based on tenfold CV and the test errors
Results

- Parsimonious model: AdaBoost with 20 weak classifiers
- AdaBoost outperforms the others
- Test Error:  
  - AdaBoost 9.0%  
  - SVM 15.2%  
  - 1-NN 18.1%  
  - 3-NN 16.3%
Conclusion

- A three-step algorithm
- A plug-in algorithm
- Each step may be improved by other methods
  - Bilinear filter vs. gray-scale morphology or anisotropic diffusion
  - K-Means vs. simple histogram thresholding
  - SVM as weak classifier in AdaBoost
Thank for your attention!
Q & A

- Performance: ~30m/s (Pentium 4 dual core 3GHz)
- In practice, the algorithm works well
- Laterals about 520, candidate laterals 1300