MORYNE: city traffic flow data collection through public transport vehicles

Christophe Parisot, Jérôme Meessen, Agnès Lapeyronnie
Multitel ASBL, Mons, Belgium
MORYNE

- Enhancement of public transport efficiency through the use of mobile sensor networks
- FP6-STREP EU co-funded project
- Partners:
  - EADS Secure Networks, France
  - EADS Secure Networks, Germany
  - Multitel ASBL, Belgium
  - Martec, France
  - Temex ceramics, France
  - GMV systemas, Spain
  - Euskaltel, Spain
  - UASO, Germany
  - BVG, Germany
  - AED, Belgium
  - KTI, Hungary
- 27 months. Project total budget: 4,000,000 euros

3rd Multitel workshop on video analysis – 30th May 2008
MORYNE objectives

3rd Multitel workshop on video analysis – 30th May 2008
Image processing

- Objectives
- Inputs and outputs
- Algorithm
  - Overview
  - Extraction of low-level information
  - High-level decisions
- Setup of the system
- Results
- Conclusion
Objectives

• With video sensors looking outside of the bus
  – Advertise the PTCC about the surrounding situation of the road traffic when busses are operating on dedicated lanes.
  – Gather traffic data that are not available with state-of-the-art fixed sensors (e.g. Potsdamer straße).
  – Send alarms when congestions are presumed.
  – Send alarms when bus lane violations are presumed.

• Constraints
  – No driver intervention required
  – Real time processing
  – Embedded platform (low CPU and memory resources)
  – Outdoor video conditions
Inputs and outputs

– Inputs

• Live video of the bus surroundings:

• Odometer and GPS data (speed and context of the bus):

– Outputs

• Traffic flow classification
• Congestion alarms
• Bus lane violation alarms

3rd Multitel workshop on video analysis – 30th May 2008
Algorithm Overview

- Video data
  - Extraction of speeds relative to the bus
  - Odometer data
    - Extraction of speeds relative to the road
      - Classification of instantaneous speeds
        - Traffic flow classification and alarms
Algorithm
Extraction of apparent speeds

• Define tracklines as segments aligned with the road within the image processing Region Of Interest

Region Of Interest
Algorithm
Extraction of apparent speeds

- Along each trackline, extract features such as
  - minimum and maximum of luminance
  - minimum and maximum of the gradient
Algorithm
Extraction of apparent speeds

- Perform feature matching among time for each trackline
- Merge information per lane
Algorithm
Correction of speeds

Extraction of speeds relative to the road:

\[ \text{absolute\_speed} = \text{apparent\_speed} + \text{bus\_speed} \]

Correction of speeds:

\[ \text{true\_speed} = \text{absolute\_speed} \times \frac{(\text{camera\_height} - \text{features\_height})}{\text{camera\_height}} \]
Algorithm
Detection of coloured vehicle features

• In order to differentiate between vehicles and ground marks when speeds are very low.
Algorithm
High level decisions

- Inputs
  - From the odometer/GPS:
    - History of odometer information every second
    - History of bus context information (e.g. bus lane position)
  - From previous image processing steps:
    - History of extracted speed every second
    - History of coloured vehicle features observations

- Outputs
  - Traffic flow classification
  - Congestion alarms
  - Bus lane violations alarms
Algorithm: Instantaneous speeds classification

- < 10 km/h without colour information: these are ground markings. The traffic is fluid.
- < 30 km/h: the traffic is congested.
- < 40 km/h: the traffic is busy.
- > 40 km/h: the traffic is fluid.
Algorithm
Traffic flow classification

– Congestion
  • $speed < congestion_{speed\_threshold}$ AND
  • $queue_{length} > queue_{length\_threshold}$

– Slow down
  • $speed < congestion_{speed\_threshold}$ AND
  • $queue_{length} > \alpha \times queue_{length\_threshold}$

– Busy
  • $speed < busy_{speed\_threshold}$ AND
  • Vehicles observed for more than half the observed period AND
  • Vehicles observed for more than half the distance operated

– Free-flow
  • Few vehicles, high speeds, etc.
Algorithm
Raising of congestion alarms

- Take past information into account
- Compute the observed queue length
- Raise alarm

\[ \text{queue\_length} = (\text{vehicles\_speed} - \text{bus\_speed}) \times \text{duration} \]

Congestion alarm when:

\[ \text{Abs(queue\_length)} - \text{distance\_other} > 50 \text{ m} \]
Algorithm
Raising of bus lane violations alarms

- Take past information into account

- If vehicle presence on bus lane within the image processing region of interest ➔ raise an alarm.
Setup of the system

– Physical installation

– Software setup
Setup of the system
Camera installation

Image processing region of interest

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system
Camera calibration

Initialisation of the calibration plugin

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system
Camera calibration

Select straight lines

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system
Camera calibration

Automatic computation of the distortion parameters

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system
Camera calibration

Indicate reference measures

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system
Camera calibration

Result of the calibration

3rd Multitel workshop on video analysis – 30th May 2008
Setup of the system Configuration

3rd Multitel workshop on video analysis – 30th May 2008
Results
Traffic data outputs

• Video duration: 8 min
• Weather conditions: rain and white splashes on the road
• Traffic flow classification updates: 30 s
• Congestion queue length threshold: 50 m

Real time image processing demonstration
Results of the image processing algorithm
Results
Traffic data outputs

3rd Multitel workshop on video analysis – 30th May 2008
Results
Accuracy of speeds

Average speed over periods of 12 minutes over a 3-day period

Time

Reference  Proposed algorithm

3rd Multitel workshop on video analysis – 30th May 2008
Objective evaluation

• 128 minutes of video including 38 bus lane violations and 5 congestions of more than 50 meters length

• Bus lane violations:
  – 55% true detections and 2.6% false detections

• Congestion alarms:
  – 100% true detections and 0% false detections

• Traffic flow classification
  – 4 classes: fluid, busy, slow down, congestion
    • 89% good classifications, 9% with one-class error, 2% with two-classes error
  – 3 classes: normal, slow down, congestion
    • 96% good classifications, 4% with one-class error
Conclusion

- Feature tracking
  - Fit vehicle movements
  - Compact representation
    - Algorithm complexity reduced
    - Easier to process in real time environment
    - Processing power and cost of the embedded platform reduced
  - Robust to environmental conditions (e.g. when compared to pattern recognition techniques, etc.)
Conclusion

• Solution already runs on the MORYNE embedded platform
• Good subjective traffic flow classification with 4 classes
• Good flow speed accuracy when compared with state of the art traffic sensors
• Able to detect most bus lane violations
Q&A

Thanks for your attention!

http://www.fp6-moryne.org