Abstract

Video based vehicle sensing from uncalibrated cameras provides useful traffic information to modern Intelligent Transportation Systems (ITS). This requires vehicle tracking and Turning Movement Counts (TMCs). Uncalibrated cameras are required for cost effective mass deployment of video sensors in an ITS [3].

Tracking vehicles in traffic is simplified compared to the general multiple target tracking problem as vehicles are constrained to move along 1-dimensional lanes [3, 4]. It is desirable to capture any general traffic video and project it into a new video with vehicles traveling horizontally at a constant speed.

1 Introduction

There is an increasing demand for video based sensing of vehicle traffic for use by ITS to reduce congestion and decrease delays [1]. These applications require vehicle tracking to estimate detailed traffic parameters including speed, delay, and TMCs [2]. Uncalibrated cameras are required for cost effective mass deployment of video sensors in an ITS [3].

Tracking vehicles in traffic is simplified compared to the general multiple target tracking problem as vehicles are constrained to move along 1-dimensional lanes [3, 4]. It is desirable to capture any general traffic video and project it into a new video with vehicles traveling horizontally at a constant speed.

2 Method

Two curves, \( f_0 \) and \( f_1 \) lie along a lane edge. See Fig. 1. The main idea is to linearly interpolate between these two curves.

For each new frame, a new image is constructed by mapping a curvilinear coordinate space aligned along the lane in the source image to a rectilinear space in the target image. A simple transformation, \( T : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) is constructed which transforms points in parametric lane coordinates, \( u \in [0, 1] \) along the lane, and \( v \in [0, 1] \) across the lane to points from the source image. A straight lane projected image, \( R \) is constructed from the source traffic image, \( I \).

\[
\begin{align*}
T(u, v) &= v f_0(u) + (1 - v) f_1(u) \quad (1) \\
R(u, v) &= I(T(u, v)) \quad (2)
\end{align*}
\]

3 Discussion

Reparameterizations of \( u \) and \( v \) may be used to approximately correct for perspective; here \( u \) is dynamically reparameterized depending on apparent lane widths \( |f_1(u) - f_0(u)| \).

Fig. 1: Two curves along lane edge. Edge finding may be used; here, they are Catmull-Rom splines with manually placed control points.

Fig. 2: Linear interpolation between two splines.

Fig. 3: (Top) Traffic video. (Bottom) Straightened image for left turn. Some extrapolation is necessary to capture entire vehicles.

Fig. 4: Stacking projections for multiple lanes and gaps. Splines \( f_1, f_2, f_3, \) and \( f_4 \) are shared between adjacent lanes or gaps.

Fig. 5: (Left) Image with two nonparallel lanes. (Right) Projected image with both lanes normalized simultaneously.

References