Available online at www.sciencedirect.com

ScienceDirect

Image and Vision Computing 26 (2008) 512–528

www.elsevier.com/locate/imavis

# Euclidean path modeling for video surveillance ☆

Imran N. Junejo *, Hassan Foroosh

*Computational Imaging Lab (CIL), University of Central Florida, 4000 Central Florida Blvd., Orlando FL 32816, USA*

Received 4 July 2006; received in revised form 9 January 2007; accepted 4 July 2007

**Abstract**

In this paper, we address the issue of Euclidean path modeling in a single camera for activity monitoring in a multi-camera video surveillance system. The method consists of a path building training phase and a testing phase. During the unsupervised training phase, after auto-calibrating a camera and thereafter metric rectifying the input trajectories, a weighted graph is constructed with trajectories represented by the nodes, and weights determined by a similarity measure. Normalized-cuts are recursively used to partition the graph into prototype paths. Each path, consisting of a partitioned group of trajectories, is represented by a path envelope and an average trajectory. For every prototype path, features such as spatial proximity, motion characteristics, curvature, and absolute world velocity are then recovered directly in the rectified images or by registering to aerial views. During the testing phase, using our simple yet efficient similarity measures for these features, we seek a relation between the trajectories of an incoming sequence and the prototype path models to identify anomalous and unusual behaviors. Real-world pedestrian sequences are used to evaluate the steps, and demonstrate the practicality of the proposed approach.


*Keywords:* Path modeling; Pedestrian surveillance; Metric rectification; Camera auto-calibration; Trajectory clustering; Route detection

## 1. Introduction

We consider the problem of monitoring an area of interest, e.g. a building entrance, parking lot, port facility, an embassy, or an airport lobby, using stationary cameras. Our goal is to model the behavior of objects of interest, e.g. cars or pedestrians, with the intent to cover as large areas as possible by generally deploying non-overlapping cameras. In path modeling [1–3] for surveillance, the goal is to build a system that, once given an acceptable set of trajectories of objects in a scene, is able to learn the routes or paths most commonly taken by objects in order to classify incoming trajectories as conforming to the model or as unusual and anomalous.

The definition of an unusual behavior might be different for different applications. For example, a person walking in a region not used by most people, a car following a zigzag path or a person running in a region where most people simply walk. A *path* or *route* can be defined as any established line of travel or access. This is the region that is most used by the objects. *Trajectory* can be defined as a path followed by an object moving through the space. Most objects tend to follow a common trajectory while entering or exiting a scene due to presence of pavements, benches, or designated pathways. Our approach can model the usual trajectories of the object and perform measurements to indicate atypical trajectories that might call for further investigation through any higher level event recognition. Thus, given an unusual or anomalous behavior, we are able to distinguish it from acceptable ones. Moreover, as common pathways are detected by clustering the trajectories, we can efficiently assign detected trajectory to its associated path model. Hence, the vision system needs only to store the path label and the object labels instead of the whole

☆ The support of the National Science Foundation (NSF) through the Grant No. IIS-0644280 is gratefully acknowledged.
* Corresponding author. Tel.: +1 4078233081.
*E-mail addresses:* ijunejo@cs.ucf.edu (I.N. Junejo), foroosh@cs.ucf.edu (H. Foroosh).

0262-8856/$ - see front matter 
doi:10.1016/j.imavis.2007.07.006

trajectory set, resulting in a significant compression for storing surveillance data.

It is, however, known that due to perspective projection the measurements made from the images do not represent Euclidean information. Thus the obtained object trajectories and consequently the associated probabilities represent projectively distorted data, unless we have a calibrated camera. This is evident from simple observations: an object grows larger and moves faster as it approaches the camera center, or two objects moving in parallel directions seem to converge at a point in the image plane. Or, for example, a person walking at a distance from a camera will be in the field of view for a longer period of time compared to a person walking very close to the camera. Similarly, for a person walking towards a camera, the obtained trajectory contains a fewer number of overlapping data points and it is not possible to obtain accurate object motion from such a trajectory. The projective camera thus makes it difficult to characterize object characteristics and behaviors – in terms of their sizes, motion, length ratios, and so on – unless camera is calibrated, in which case one can perform Euclidean measurements directly from images. For this purpose, as one of the steps of our framework, we present a novel robust and linear solution to *auto-calibrate* any camera that is used in the system.

In a nutshell, this paper addresses a comprehensive set of problems for building a path modeling system, by proposing novel methods to (i) auto-calibrate the camera and estimate the pan and tilt angles, (ii) perform metric rectification of the input sequence, (iii) register the sequence to the aerial imagery, (iv) obtain metric information about the objects from the rectified and registered images, and hence (v) build Euclidean path models to monitor and characterize behavior of the objects by observing and performing measurements on trajectories. Therefore the remainder of this paper is organized as follows: existing methods for path modeling and auto-calibration are described in Section 2, followed by a short introduction to the pinhole camera model in Section 3. The *Training phase*, consisting of auto-calibration and trajectory clustering, is described in Sections 4 and 5, respectively. Section 6 defines the *testing phase* for distinguishing a non-conforming trajectory from established paths and discusses occlusion. The experimental evaluations are presented in Section 7 followed by conclusion in Section 8.

## 2. Related work

We divide the task of path modeling for surveillance in a single camera into three steps. The first step involves detecting and tracking objects in the video frames. Through this process, one can extract image plane trajectories of moving objects, which provide projectively distorted 2D representation of the true path in the 3D scene. In the second step, projective distortions are removed from the extracted trajectories to provide a Euclidean model of the path in the 3D space. Finally, a scene path model is built, whereby anomalous behaviors are detected by matching incoming trajectories to the model path for the area under surveillance. The system is able to log the behavior of an object from the moment it enters the camera's field of view until it exits, and enables the user to determine its conformity to the path model.

The first step of tracking is essentially a correspondence problem and is not the primary focus of this paper; correspondence needs to be established between an object seen in the current frame and those seen in previous frames. Tracking is a widely studied problem in computer vision, and many suitable trackers exist for our purpose [4–8]. We used the tracker presented by Javed et al. [9] to validate our method.

The second step, i.e. removal of the projective distortion, is very essential. As argued above, in order to obtain undistorted and real-world information from any video sequence, the camera needs to be calibrated. Calibration is a necessary process in computer vision in order to obtain Euclidean information about the scene (up to a global scale), and to determine the rigid camera motion. Camera calibration methods can be classified into two broad categories:

(1) *Three-dimensional object-based calibration:* Calibration by observing an object whose geometry in the 3D space is known. The original work in this category is due to Tsai [10]. Generally, the calibration object consists of two or more planar grid patterns that may be set orthogonal to each other. This approach requires elaborate setup and can be quite time consuming.
(2) *Self-calibration:* The metric properties of the camera are determined directly from constraints on the internal and/or external parameters [11–16]. No calibration objects are required in these techniques. By simply moving a camera in a static scene, the rigidity of the scene provides constraints that are used to calibrate the camera. Therefore, if images are taken from a camera over a period of time, the point correspondences between the images are sufficient to recover both the intrinsic and extrinsic parameters.

Another *intermediate* technique for camera calibration, which is closely related to our approach, is based on the *geometric constraints* of the scene. The knowledge of the scene geometry, e.g. *vanishing points* or *vanishing lines*, can be used to impose constraints on the camera parameters [17–20]. Original work on camera calibration using vanishing points is due to Caprile and Torre [20]. Liebowitz et al. [18] developed a method to compute the camera intrinsics by using the Cholesky decomposition [21]. Similarly, Cipolla et al. [19] use three orthogonal vanishing points and one reference point to determine both intrinsic and extrinsic parameters. Lv et al. [22] were the first to propose calibration by recovering the horizon line and the vanishing points from observed walking humans. However,