Computers & Graphics 55 (2016) 68–79

Contents lists available at ScienceDirect

Computers & Graphics

journal homepage: www.elsevier.com/locate/cag

Special Section on Graphics Interface

Online parameter learning for data-driven crowd simulation and content generation

Aniket Bera, Sujeong Kim, Dinesh Manocha

Department of Computer Science, The University of North Carolina at Chapel Hill, Chapel Hill, NC, USA

ARTICLE INFO

Article history:
Received 16 August 2015
Received in revised form
30 September 2015
Accepted 19 October 2015
Available online 3 December 2015

Keywords:
Crowd simulation
Data-driven simulation
Pedestrian tracking
Crowd replication
Augmented crowds

ABSTRACT

We present an online parameter learning algorithm for data-driven crowd simulation and crowd content generation. Our formulation is based on incrementally learning pedestrian motion models and behaviors from crowd videos. We combine the learned crowd-simulation model with an online tracker to compute accurate, smooth pedestrian trajectories. We refine the motion model using an optimization technique to estimate the agents' simulation parameters. We also use an adaptive-particle filtering scheme for improved computational efficiency. We highlight the benefits of our approach for improved data-driven crowd simulation, including crowd replication, augmented crowds and merging the behavior of pedestrians from multiple videos. We highlight our algorithm's performance in various test scenarios containing tens of human-like agents and evaluate it using standard metrics.

1. Introduction

Realistic simulation of crowd behavior has many real-world applications, and has been a topic of active research in computer animation, virtual environments, robotics, vision, and pedestrian dynamics. In computer games and animations, realistic crowd simulations can enhance the user's experience and perception. In safety analysis and surveillance applications, crowd simulation methods are used to recognize normal or abnormal behaviors for live video streams. In computer-aided design, crowd simulation methods are used to predict the pedestrian flows, and in robotics they are used for autonomous navigation. A key challenge in all these applications is to model the dynamics and the variety of crowd behaviors that are frequently observed in real-world scenarios.

In general, it is challenging to simulate realistic crowd behaviors. While researchers in various fields like psychology and other social sciences have been studying and observing human behavior for decades, there are no widely accepted models that can simulate a wide variety of behaviors. This is in part because human behaviors are governed by multiple factors including the personality of each pedestrian, environment, external events, etc. Often, designers or animators take such factors into account and manually generate scene-specific behaviors or trajectories. This can be time consuming. There is extensive work on local and global crowd simulation algorithms that can be used for local collision avoidance, navigation, or trajectory computation. However, generating realistic behaviors or simulations using such methods involves considerable tweaking or variations of simulation parameters. As the number of agents or the complexity of the scenarios increase, it becomes increasingly hard to model the diversity of behaviors or the interactions between Fig. 1.

Instead of using explicit crowd-simulation models or behavior rules, data-driven crowd simulation algorithms use examples generated using crowd videos or motion capture data [1,2]. Once sufficient data on real-world crowd trajectories or behavior data has been collected, including agent trajectories, data on various styles of gestures, crowd formations, and interactions between agents, these data-driven methods can generate realistic behaviors or trajectories without excessive human intervention. One of the challenges with data-driven methods is related to collecting the trajectory or behavior data from videos or other sensor data. Advances in real-world capturing technologies have resulted in large databases of crowd videos (e.g. YouTube). However, most data-driven crowd simulation systems require manual tracking and/or annotation of environments (e.g. obstacles) and behaviors. These techniques are time consuming, limited to small group interactions, and not scalable to a large number of crowd videos. Furthermore, they are limited to simple scenarios in terms of number of agents or crowd density.

Main Results: We present a trajectory extraction and behavior learning algorithm for data-driven crowd simulation. Our approach is used to generate smooth trajectories for one or more pedestrians in the simulated environment. Furthermore, we can combine the trajectories extracted from two or more different videos to generate *mixed* data-driven crowd simulation. Our tracking algorithm uses the first few frames to automatically learn the best parameters for each pedestrian and dynamically compute

http://dx.doi.org/10.1016/j.cag.2015.10.009
0097-8493/

Fig. 1. (a) Pedestrian tracking using a simple particle filter and motion model. Yellow circles indicate some of the problems with prior tracking results, e.g. missing or incorrect tracks (b) Tracking using our algorithms with improved accuracy and smooth trajectories. (c) Rendering of the pedestrian trajectories from our data-driven crowd simulation system.

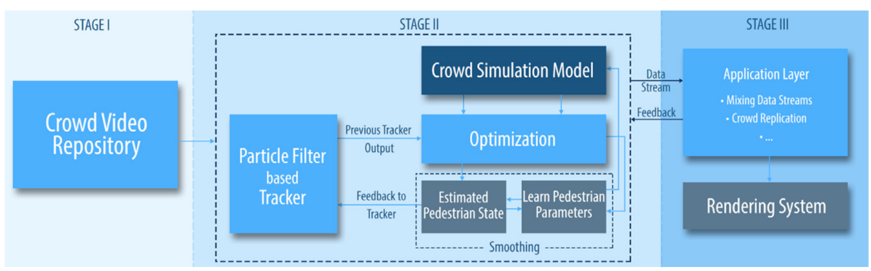

Fig. 2. An overview of our Data-Driven Crowd Simulation System. We have a three-stage pipeline. We begin with a selection of crowd videos from the pool of crowd videos available. Next we feed this into our tracking pipeline. We iteratively learn the motion model parameters and use them to improve the tracking. The feedback is bidirectional; the simulation model is re-trained after a fixed number of frames. We iteratively compare prior pedestrian state history (pedestrian position, velocity) and prior tracker inputs to refine parameters and generate smooth trajectories. Lastly, our learned behavior and the model's resulting trajectories are used as input to our application layer, where we use them to replicate the crowd behaviors seen in individual source videos, or to mix agent trajectories from multiple crowd videos.

a motion model to extract smooth trajectories for each pedestrian in an online manner.

There are many benefits and possible applications of our approach. Our tracking algorithm results in improved accuracy; it reduces the number of *ID switches* and *lost tracks* and generates accurate trajectories that can be used for data-driven simulation. Moreover, the trajectories our approach generates are smooth and can be directly used for the pedestrians in simulated environments. Our approach can automatically replicate real-world crowd trajectories and behaviors, and generate crowd trajectories similar to that observed in real-world videos. Additionally, our method allows for easy mixing of multiple tracking results, which allows for modeling of more complex scenarios. We demonstrate the performance of our approach using a database of multiple outdoor crowd videos with tens of agents.

This paper is an extension to our previous work [3]. We improve over prior work by (a) introducing an adaptive particle filtering scheme which optimizes computation cost, (b) adding more applications like augmented crowds, (c) more detailed analysis and evaluation with many more datasets.

The rest of the paper is organized as follows. Section 2 reviews related work in data-driven crowd simulation and tracking. Section 3 introduces our notation and terminology and gives an overview of our approach. We give a general overview of our tracking and learning algorithm in Section 4 and highlight some of its applications using real-world videos in Section 5.

2. Related work

In this section, we give a brief overview of the prior work on data-driven crowd simulation and pedestrian tracking.

2.1. Data-driven crowd simulation

Many techniques have been proposed to fit motion-model parameters to a given real-world video, and evaluate how closely they can replicate the original data. Lerner et al. [5] use density-based measures, Guy et al. [6] propose an entropy-based measure for similarity, Berseth et al. [7] use performance criteria such as minimizing time and effort, Wolinski et al. [8] present optimization techniques to find good parameters for different motion models. Musse et al. [9] suggest a method to learn the intentions from tracked data by building a desired velocity field extrapolated from low density tracking data. Patil et al. [10] use flow field extracted from an input video or from a sketch to direct the virtual