SPS Chapter Research Interim Report

Quantitative evaluation of pedestrian movement models:
A *real* many-body problem

School: Purdue University
Chapter: 5781
Requested: $1089.00
Project leader(s): Dawith Lim (2018-19), Adam Kline (2017-18)

Abstract

A number of papers have been published since 2000 which attempt to model pedestrian crowd flow dynamics using basic equations of motion. Here we propose a study which aims to critically evaluate a handful of these models in terms of their predictive accuracies, then categorize them according to their strengths and weaknesses. This will be accomplished by taking aerial footage of pedestrians motion on campus using an unmodified quadcopter drone.

1 Statement of activity

1.1 Interim assessment

1.1.1 Research goals

In this project, we set out to answer the following questions:

- Can an unmodified, off-the-shelf quadcopter drone be used effectively to collect quantitative crowd flow data that is useful for research purposes?
- Do the models identified and discussed in this project provide accurate representations of real crowd movement?

1.1.2 Description of project

To answer the questions above, we are using a commercially available drone (DJI Mavic Pro) to obtain high resolution video footage of pedestrian movement from a high, fixed, vantage point. At the outset of this project, we were uncertain that a drone within the alloted budget would be capable of producing video with the necessary stability and resolution. This uncertainty has been largely dispelled by our initial data collection runs, which will be discussed shortly.

In our proposal, we listed background subtraction [3] and support vector machines [8] as possible methods of extracting pedestrian coordinate data from footage. Since then, we have decided to instead employ an object classification method known as YOLO [9], which requires training, but produces consistent results. To identify the trajectories of individual pedestrians, we splice the footage into individual frames and analyze each individually. Because the video frame rate (20 Hz) is high enough, each pedestrian identity can be inferred between frames by continuity. When this method is completed, we can fully map the trajectories of each pedestrian captured by the footage.

By successfully extracting pedestrian movement data from video, the first question is answered affirmatively. To answer the second question, a systematic comparison between extracted data and model predictions is needed. To this end, we are developing Python simulations for each of the predictive models discussed in the proposal [4, 6, 1, 7]. Through this endeavor, younger undergraduate students acquire skills highly relevant to research in physics, including the ability to read, summarize and extract information from research papers, as well as programming simulations and advanced regression.

1.1.3 Progress on research goals

Data collection

By taking footage several times at various heights we find that 55-60 m appears to be the appropriate drone flight altitude for data collection. This considers factors such as the resolution of human features, view angle and noise/visibility. We will elaborate on the details in the initial results section.

The second part of data collection, namely, the pedestrian detection from the raw image data, is nearing completion. As of now, we have implemented YOLO (You Only Look Once; it's a highly effective open source feature detection algorithm), and trained it with frames from a video recording taken from 51 meters, which has been manually tagged frame-by-frame. The code is functional, but we wish to improve the robustness of the training set and add classifications for non-pedestrian actors such as cyclists and on-campus vehicles.

We have also experienced two drone crashes, both by improper use while not filming or around pedestrians, which caused about a month of setback in repairs. Fortunately, the drone we have purchased is highly modular, so the repair itself was cheap and simple. These two crashes also resulted in development of an internal protocol for flying the drone.

Coding models

We have not yet finished implementing python scripts representing the four predictive models discussed. Originally, the project leaders had hoped to instruct the underclassmen so they could code basic elements of these simulations themselves. For instance, the MATLAB implementation of the Helbing '95 model [2] which was built for the initial proposal was re-purposed as a teaching tool. The aim was to concretely show how a simple predictive model gave deterministic results. In practice, underclassmen had a hard time producing usable code for the other models, in part because each other model was significantly more sophisticated, and in part because most members knew neither MATLAB nor Python. Nonetheless, we feel that some progress was made in teaching the younger members how to critically read papers and code simple simulations in Python.

Member education

Our initial approach was to divide the participating members into smaller subgroups, and give them one paper to read and extract relevant equations that will be used to develop the model. The rationale behind this approach was that students will learn more by getting their hands right into the problem solving and asking the upperclassmen when they encounter issues that they are struggling to overcome.

This approach turned out to be unsuccessful, as members quickly lost motivation when they were faced with the seemingly unsurmountable task. As a consequence, we had to scale back on our goal and take a more controlling approach where an upperclassmen reads and summarize a paper and present the summary to the underclassmen. This approach was more successful in that it allowed the students to understand the contents of the paper, but it created a significant bottleneck since at this point only two upperclassmen were available to offer help.

One somewhat successful tactic was the introduction of smaller, more tractable simulation problems. For instance, an underclassman might be given the task of simulating a double pendulum given l_1, l_2, m_1, m_2, etc. This achieved moderate success in the regard that underclassmen were developing valuable skills yet did not feel defeated by their work, thus improving morale and participation.

Moving forward, the project leaders will be handling most of the predictive model coding, mostly to ensure that the concrete milestone of a data-ready computational framework is in place by the summer's end. At this milestone, we do not plan to stop encouraging underclassmen participation in the theory part of this project. As is shown in our timeline, active members will practice the art of distilling models down to straightforward, predictive systems of equations. They can then contribute to the final report, and hopefully we can send a young member to present our results in a poster session or conference.

1.1.4 Modification in the scope of project

Initially, we supposed that group egress models [4, 10, 11] would be interesting to investigate. These models typically reveal some scaling law involving group size, room geometry, door size, panic levels, and egress rate. A few problems arise: In the case of *egress* we can only observe pedestrians exiting an opening; little can be said about geometry inside given the vantage point of our drone. If we instead consider a heavy crowd entering a building, we run into a technological issue: our image detection algorithm is noted by its creators to "struggle with small objects that appear in groups, such as flocks of birds". We may forgo this endeavor for the sake of thoroughly completing the more promising component of the project.

1.1.5 Personnel

10 members (all of whom are/were members of Purdue chapter of SPS) are currently participating in the project: Adam Kline, who was initially the project leader, has graduated and now provides help with the theoretical aspects of the project. He will finish most of the model coding and provide consultation when we are faced with conceptual difficulties, especially during the final analyses. Dawith Lim, who is the newly appointed project leader, delegates tasks to members and manages the overall trajectory of the project, and is also working on reproducing the models based on research papers. Charles Li handles the machine learning portion of the project with help from Albert Xu, and provides Python expertise in all other coding endeavors. Albert Xu, Akshat Jha, Andrew Gustafson, Braden Buck, Ethan Zweig, Henry Dawson, Vic Dong, and Wyatt Montgomery are underclassmen who are in the process of learning python programming and/or learning literature review skills. Among these members, Albert, Braden and Henry have also been trained to fly the drone to obtain the data.

1.1.6 SPS connection

We have made a tangible progress in helping underclassmen to read and understand research papers with guidance from upperclassmen, and also made progress in teaching other students python programming, although there is still much room for improvement. At a national level, we are setting a precedent for robust, low-budget replication studies that affect recent insights in untraditional fields of physics. We hope our work inspires other SPS chapters to consider the power and general applicability of physics methods in a broader scope when we present our work.

1.2 Updated background

In the proposal, we noted that one goal of this research is testing both quantitative and qualitative accuracies of several predictive models. One metric which we did not mention for evaluating qualitative accuracy was the distribution of movement errors. If we see fluctuations away from models that are Gaussian, that should indicate good qualitative agreement. As the error distribution variance decreases, quantitative agreement is increased. These relatively simple metrics are more straightforward than the previously proposed Lyapunov exponent definitions, and will be adopted for the final analysis in addition to the aforementioned methods.

A new piece of literature by Mejia et al. [5] experimentally evaluates a social force model by using infrared cameras in a mall hallway of known dimensions. The paper is not comprehensive in scope with regards to past findings in social-force models, but it is interesting in two regards. First, pedestrian trajectory is captured with an apparatus distinct from ours, which is less flexible and more costly. Secondly, the force model is replaced by an effective potential, derived by extremizing the action functional of trajectory.

1.3 Description of research - methods, design, and procedures

Because the YOLO object detection method requires significant training, and because there are seemingly no vast online repositories of tagged overhead pedestrian data, tagging was done manually; frame-by-frame and person-by-person. To achieve this, Charles set up a server which was accessible via internet to anybody in our SPS chapter. Upon loading this page, a frame from the video was displayed (see Fig. 1) and the user was instructed to click on all of the humans they could identify. By adding a leaderboard functionality, friendly competitiveness between chapter members pushed the massive tagging operation to completion. Presumably, if we decide to expand the feature set to include moving obstacles (cyclists, vehicles, etc.) and other weather conditions, a similar procedure could be carried out.

Figure 1: Still-frame from tagging run with bounding boxes capturing individuals. Boxes are 35×35 pixels.

Nearly every model which we are investigating (the exception being [1]) assumes some physical parameters concerning the mobility characteristics of individual pedestrians, e.g. preferred walking speed, reaction time, or destination, to name a few. Simply adopting default values for these parameters as they are given in the literature would be easy, but may introduce avoidable errors. After all, what purpose do free parameters serve beyond optimizing model agreement? To this end, we will attempt to build our simulations in Theano, a tensor manipulation library for Python that is intended for deep learning applications, but can be used for regression more generally [12].

1.4 Initial results

Before deciding to work with YOLO, Albert attempted a background-subtraction based approach to data tagging. This produced early success in picking out isolated people. However, for close groups proved tricky to separate into individuals, and spurious signals sometimes arose in the surrounding environment.

1.4.1 Data collection

For this particular model, the hum of the drone is audible from ground when it is flying below 35m (assuming there is no other source of loise). At 50m, the drone's hum is essentially buried in background noise, but it is still visible if anyone looks up. These are potential sources of problem because the pedestrian dynamics may be affected by the presence of drone (e.g. avoid walking directly under, walking faster to move away from it, etc.). The view angle is important because the features visible to the camera varies as the view angle varies. Lastly, the feature resolution matters because it determines what the machine learning algorithm is given to work with.

2 Statement of next steps

2.1 Plan for carrying out the remainder of project

Date	Goal
2018/07/30	Complete extracting pedestrian models from papers
	During two months period (June-July), active participating members will extract a model from one paper together every five days.
2018/08/30	Complete tuning the feature detection algorithm Collect and process as many sets of data as possible (minimum eight)
	The feature detection algorithm will be tuned once Charles (in charge of ML portion) is back on campus, and should take no more than a couple of weeks.
2018/09/31	Complete error analysis of coded models, begin summarizing results for final report
2018/10/31	Complete first draft for final report, perform additional data collection or paper review as necessary
2018/11/30	Complete final report, search for venue for publication/presentation of work

As noted from the date, we made the plan with one month of overhead in case certain components get bogged down and require additional time. That being said, through the issues and delays we faced during the first half of the project, we now have a good idea of how to approach each of the components, so we believe that the timeline given above is still reasonable.

References

[1] Emiliano Cristiani, Benedetto Piccoli, and Andrea Tosin. Multiscale modeling of granular flows with application to crowd dynamics. *Multiscale Modeling & Simulation*, 9(1):155–182, 2011.

[2] Dirk Helbing and Peter Molnar. Social force model for pedestrian dynamics. *Physical review E*, 51(5):4282, 1995.

[3] ZuWhan Kim. Real time object tracking based on dynamic feature grouping with background subtraction. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[4] Paul A Langston, Robert Masling, and Basel N Asmar. Crowd dynamics discrete element multi-circle model. *Safety Science*, 44(5):395–417, 2006.

[5] S Mejia, JE Lugo, R Doti, and J Faubert. Evaluation of a thermal image for pedestrian modeling using the least-action principle. In *Soft Computing: Theories and Applications*, pages 377–387. Springer, 2018.

[6] Mehdi Moussaïd, Dirk Helbing, and Guy Theraulaz. How simple rules determine pedestrian behavior and crowd disasters. *Proceedings of the National Academy of Sciences*, 108(17):6884–6888, 2011.

[7] Rahul Narain, Abhinav Golas, Sean Curtis, and Ming C Lin. Aggregate dynamics for dense crowd simulation. In *ACM Transactions on Graphics (TOG)*, volume 28, page 122. ACM, 2009.

[8] Constantine Papageorgiou and Tomaso Poggio. Trainable pedestrian detection. In *Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on*, volume 4, pages 35–39. IEEE, 1999.

[9] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[10] Weiguo Song, Xuan Xu, Bing-Hong Wang, and Shunjiang Ni. Simulation of evacuation processes using a multi-grid model for pedestrian dynamics. *Physica A: Statistical Mechanics and its Applications*, 363(2):492–500, 2006.

[11] Yusuke Tajima and Takashi Nagatani. Scaling behavior of crowd flow outside a hall. *Physica A: Statistical Mechanics and its Applications*, 292(1):545–554, 2001.

[12] Theano Development Team. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May 2016.