B.Comp. Dissertation

CA Report

Analysis and Simulation of Crowd Management

By

Chen Di

Department of Computer Science

School of Computing

National University of Singapore

2016/2017

B.Comp. Dissertation

CA Report

Analysis and Simulation of Crowd Management

By

Chen Di

Department of Computer Science

School of Computing

National University of Singapore

2016/2017

Project ID: H026510

Project Supervisor: Dr. Antoine Fagette, Assoc. Prof. Gary S. H. TAN

Deliverables:

Report: 1 Volume

Abstract

Identifying and predicting individual movement from a live stream video source could be of great benefits in preventing potential hazardous public accidents from happening. In general cases, a human recognition algorithm is capable of identifying and locate individual person from a video input of a surveillance camera. However, due to the time complexity of the algorithm and limitation on hardware performance, the algorithm is not able to provide real time position information. This report will examine an algorithm that make approximated predictions of people's positions to fill in the time gap created by the human recognition algorithm.

Subject Descriptors: (Adapted using ACM's 2012 computing classification system)

Tracking

Matching

Keywords:

Optical Flow Tracking

Implementation Software:

Python 2.7, OpenCV

Acknowledgement

The author sincerely appreciates the advising of Dr. Antoine Fagette, who provided deep insights on the problem and make frequent visit to the school for discussion. Also a great thanks to Prof. Gary S. H. TAN. who has always been supportive in giving relevant papers that are relevant to my study and provide necessary hardware devices for running simulation softwares.

Contents

1 Introduction

1.1 Problem Background

Crowd in public place has always been a problem incubator, once accidents happened, casualty usually tends to grow high. When a public facility like an airport or a train stations is being designed, one of the crucial functionality to be concerned is the capability of handling hazardous accidents that could potentially occurred among crowds of people. However, number of cases that could be taken into consideration during the design phases are limited, furthermore, in real situations, factors that could affect crowd behaviors could vary hugely from pre-planned scenarios, and thus pre-designed static models tend to fail.

Surveillance cameras such as closed circuit televisions (i.e. CCTV) are usually used to monitor crowd status in a public area. With a massive deployment of workable cameras in public places, a huge amount of data, some of which are even live streaming, recording crowd behaviors of a real scenario is collected, and thus made it possible to build more reliable and accurate crowd behavior prediction models upon it.

1.2 Project Background

The overall parental project, which is under the management and sponsorship of Thales Group™ , aims to develop an analytical tool that could provide crowd behaviors/movements predictions using real-time video data from surveillance cameras.

The project constitutes three main parts. “Part 1” is about “Video Analysis”. This section includes identifying individual human in a streaming video, and continuously tracking their movements. For “Part 2”, the focus is on “Behavior Forecast and Decision Support”, where certain learning algorithms are to be applied to generate human behavior prediction models from the video data. Further, a decision system should be developed to simulation a specific event input for the prediction system. (e.g. a bomb blasting event or a celebrity occurrence event). Lastly for “Part 3”, the focus is on “Visualization”, which aims to develop a 3D visualization system to display the predicted crowd movement. A figure on the overall project architecture is provided below.

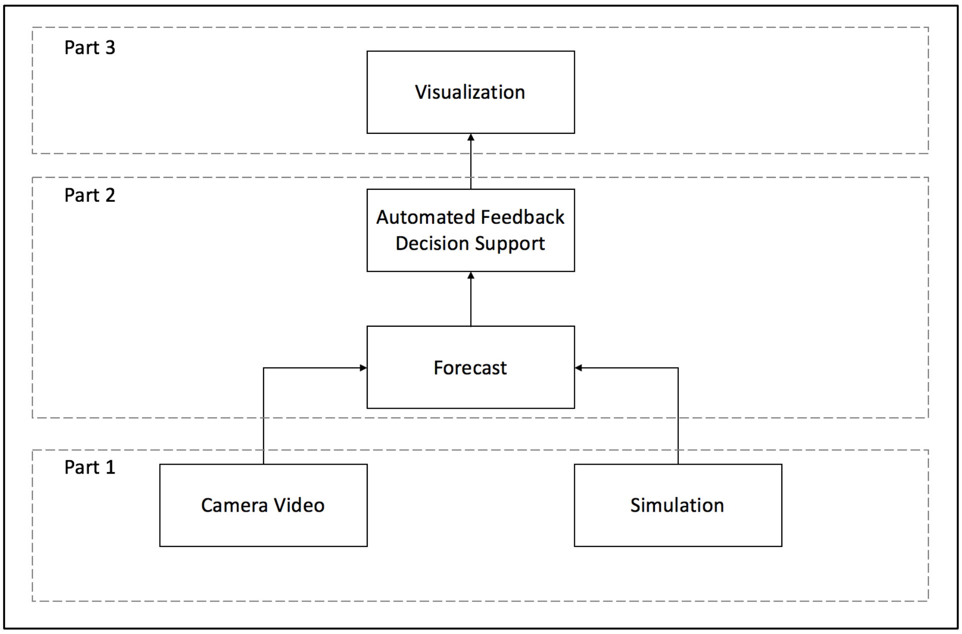

Figure 1 Overall Project Scope

1.3 Project Goal

The project scope of this undergraduate final year project is mainly on solving problems occur in the "Part 1", namely "Video Analysis", of the bigger parental project. Under "Video Analysis", the focus of the research for 2016/2017 fall semester is on tracking individual human's movement and matching their positions in a video.

1.4 Project Progress

For fall semester 2016/17, the first half semester's work was getting familiar with the topic through reading relevant research works such as paper provided by Dr. Antoine, and Prof. Tan. Furthermore, in order to examine how existing crowd simulation tool works, a few simulation scenarios are built using MassMotion® for practice. The second half of the semester, an approach on matching people's position between 2 frames is been proposed and an algorithm on tracking pedestrian position using optical flow has been implemented. Details regarding each of the work accomplished will be examined in the rest of the report.

2 Studies of Existing Materials & Tools

2.1 Crowd analysis in a nut shell

The doctoral work *Video Crowd Detection and Behavior Analaysis* (Fagette, 2014) of Dr. Antoine Fayette is the main study and referencing material for understanding current accomplishments in video based crowd detection & analysis. Problems in the field are examined from 3 perspectives, namely, "Crowd Detecting & Counting", "Tracking" and "Crowd Behavior Understanding".

2.1.1 Crowd Detecting & Counting

Crowd detection can be approached differently depending on the quality of the video or image. For instance, under a dense crowd scenario, or when the camera is far from the object scene, it would be hard to separate an individual pedestrian from background pixels. And thus algorithms that perform human factor feature points detection may not render a good result, a probabilistic approach that approximates the density of the crowd in a certain area could be a more appropriate choice.

In Dr. Antoine's doctoral thesis paper, four ways are generalized for crowd detection (Fagette, 2014). The first way is to perform direct detection on pedestrians, where feature detection such as "head-shoulder" detection (Tu, Sebastian, Doretto, & Krahnstoever, 2008) are usually performed. The second approach aims to detect "non pedestrians" and thus subtract background from the scene, any objects left will then be treated as a human. The third method usually deals with moving crowd. By assuming the static status of background environment, optical flow of the video stream is typically analyzed to detect a crowd that is moving (Reisman, Mano, Avidan, & Shashua, 2004). Lastly, the approach tries to find patterns among textures of the crowd in order to locate it.

Methods for counting the number of people in a crowd also vary depending on the quality of the video. Two approaches are usually taken estimate the size of the crowd. The first approach tries to perform pixel level analysis. The approach tends to give a population size estimation basing on the number of pixels being occupied by the crowd. Such method is capable of rendering a close enough approximation is because there tends to be a linear correlation between the number of pixel and the crowd population size (Hou & Pang, 2008). The second approach looks at the crowd from a texture based perspective. For instance, the more people there are in the image, the less diagonal and course the texture of the crowd tends to be, and thus it could serve as a good indicator of how many people are there in the scene (Haralick, 1979).

2.1.2 Tracking

The movement of the crowd analyzed from two perspectives. The first way is to perform “object level tracking”, where each pedestrian in the scene is tracked separately. The overall moving pattern of the crowd is the aggregation of all individual objects’ movements. The other way is to analyze the crowd from a holistic view. Such approaches typically treat the crowd as a single object and analyze it’s moving pattern from its optical flow data.

The tracking algorithm proposed in Dr. Antoine’s thesis paper, which is also the one implemented in the simulation system currently under the development of Thales Group™ , takes the holistic approach. According to the density distribution of the population, the algorithm assigns a cloud of particles into of the scene as a representation of the crowd, each particles follows a behavior pattern of typical a human being in the crowd, and the whole set of particles serves as tracking points of the crowd.

2.1.3 Crowd Behavior Understanding

Behavior analysis also follows two approaches, one imposes examinations on an object level (i.e. individual pedestrian level), the other analyzes the crowd on a holistic point of view. For the object level approach, analysis usually focus on understanding individual human behaviors in the crowd from a sociology perspective. A few models have been raised to generalize human behavior, one of which for example, interprets the factors that drive human’s behavior as “Social Forces” (Helbing & Molnár, 1995), essentially “Social Force” view effects of environment and internal impulses that drive a person to move as a set of 2-D vectors, and the summation of all vectors determines the eventual movement of that individual.

For holistic level approach, motions are usually detected on a group level. For instance, rather than examine individual’s moving behavior, a filter, size of which can vary, is used to extract the general moving tendency of a block of people. By comparing the moving pattern of crowd during an emergency event, the conclusion on whether an accident is happening could therefore be derived (Boghossian & Velastin, 1999).

2.2 Simulation with MassMotion®

MassMotion® is a software that is capable of simulating crowd behaviors in a closed environment (e.g. airport, subway station), furthermore, it’s able to display the simulation result visually. The software is usually used for evaluating crowd moving efficiency of a specific environment design.

Under the scope of this undergraduate final year project, MassMotion® serves as a practice tool to getting familiar with how simulation tool works. The work completed with MassMotion® however does not generate direct benefits towards achieving the project goal.

2.2.1 Build Environment

The software accept 3D models build by Maya® or AutoCAD® as inputs. When working with MassMotion®, a few 3D modeling environments were tested, one of which is a transportation facility, shown in figure 2. When a 3D model gets imported into simulation environment, the software is not able to automatically identify the functionality of each geometry (e.g. a wall, an elevator), therefore objects needed to be assign functions manually.

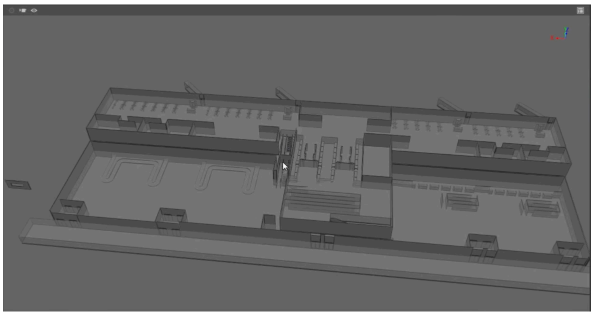

Figure 2. Raw 3D Model

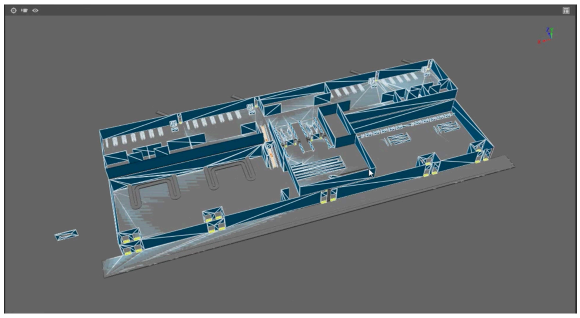

Figure 3. Model with Object Functions

2.2.2 Spawning the Crowd

After the all the objects in the model have their functions mapped, “Entrance” and “Exit” points are marked to specified where the crowd are generated and which destination they are supposed heading to. After that, crowd could then be generated. In MassMotion®, each individual people has its own 3D representation, and it could be spawned in a pre-specified location. The spawning algorithm support multiple crowd generating modes. In the tested scenario, a triangular spawning rate is tested, where the number of people generated increase linearly and come out in a triangular shape, and stop until it the total number of people spawned has reach pre-specified maximum.

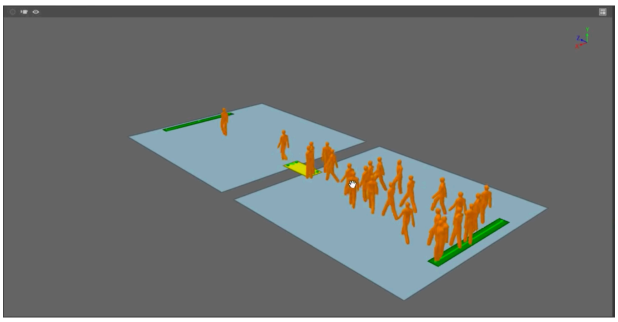

Figure 4. Crowd Representation. Triangular generating algorithm is used in the scene.

3 Connecting Two Frames Problem

In the video analysis part of the parental project, an obstacle occurred in tracking people from a frame to another. Currently an algorithm is capable of recognizing individual people's position from the scene. However, due to the performance limitation, it could possibility takes up to 10s+ to generate people's position from 1 frame of video. When it starts to process the second time, the current frame is 10s+ away from the previous frame, and the people's position in the frame could vary hugely. For instance, for the case shown in figure 5, there are two possible moving patterns, illustrated in figure 6, and figure 7. Thus, in order continuously track each individual person, an algorithm is needed to identify the same person's position in two frames.

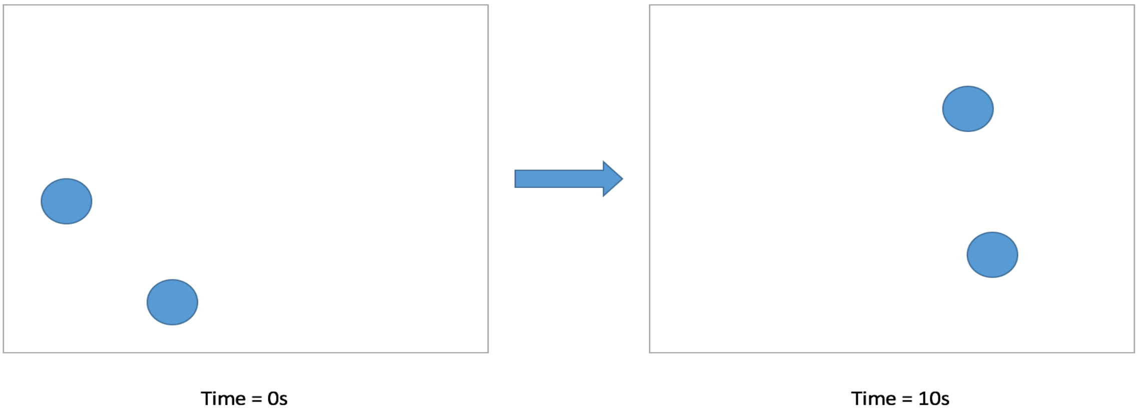

Figure 5. A possible pedestrian case

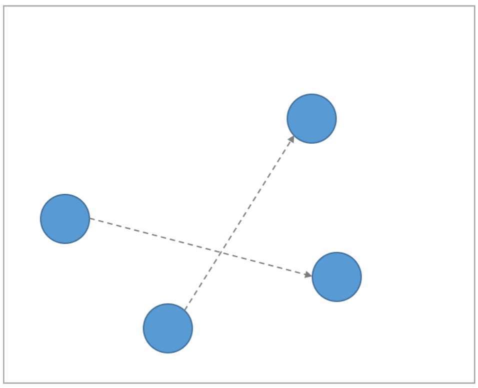

Figure 6. Possibility 1. One possible position displacement of 2 pedestrians.

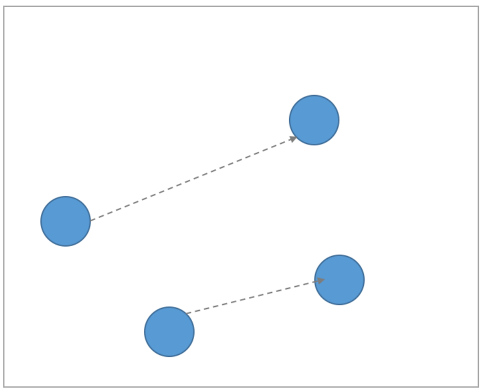

Figure 7. Possibility 2. One possible position displacement of 2 pedestrians.

3.1 Algorithm proposed

The algorithm that's helps fill in the time gap is built mainly based on using information retrieved from dense optical flow calculation. Dense optical flow algorithm (Farneback, 2003) is considerably efficient and is able to give near "real time" results of the changes of position for each pixel between two consecutive frames. The output result will be a 2 dimensional vector that represents the displacement for every pixel. By aggregating optical flow displacement data frame by frame, an approximated final location on a later frame will be generated for every pedestrian occurring in the scene, shown in figure 8.

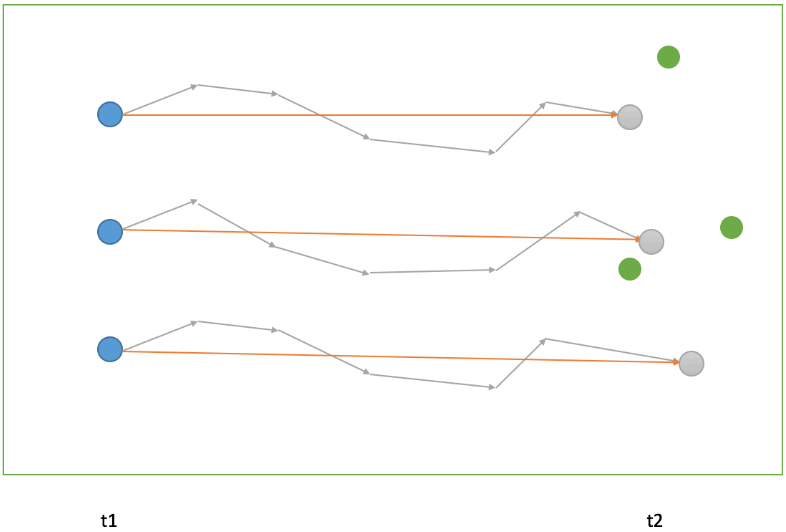

Figure 8. Aggregation of optical flow displacement. Blue dots indicate positions at an early frame, green dots indicate positions at a 10 seconds later frame, grey dots indicate the resulting position from the aggregation of optical flow displacements.

After getting the approximated location at the destination frame, a matching algorithms is used to associate optical flow displacements with actual human position from recognition algorithm. The goal of the "association algorithm" is to find the most possible matching pair between approximated position and actual human position, shown in figure 9. One intuitive greedy approach is to calculate the nearest point for each approximated optical flow position, however the algorithm is possible to populate extreme long associating for some points, illustrated in figure 10.

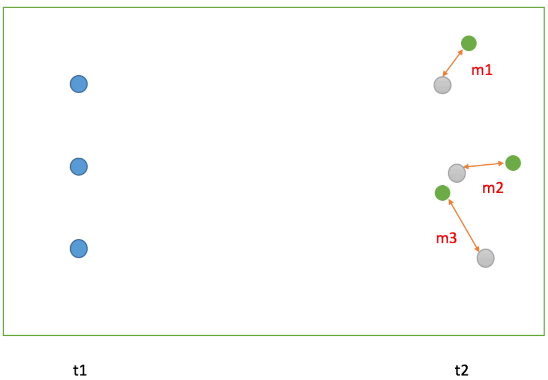

Figure 9. The "association problem". The algorithm aims to minimize the association distance of each pair of points.

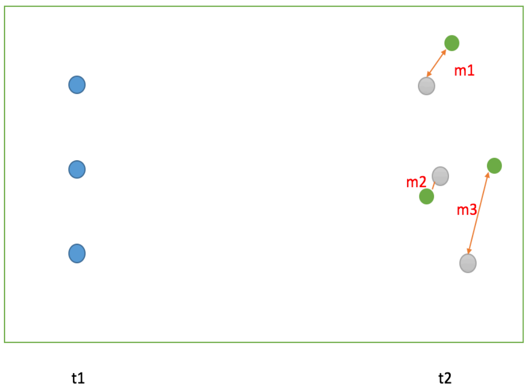

Figure 10. A fail case for finding nearest point approach. In the scene, grey point in the middle gets calculated first, and finds its nearest green point, resulting in the bottom grey points no choice but find a point far from itself.

In order to reduce the case of having abnormally long association caused by points calculating sequence. The algorithm proposed tries to tackle on reducing the maximum association first by giving priority to match to points who are prone to generate long association. As shown in figure 11, a "scanning range" is added around each of the optical flow approximation points. The size/width of the range is determined by the average displacement error throughout the optical flow aggregation process. For points that does not have any "potential association points" in their scanning range, they have the priority to choose their association points first because they tends to pick long association. For the set of priority points, each of them will find their nearest points. Notice that in such situation, they still face the problem of picking "long association", which is again caused by inappropriate point picking up sequence, however, since only a minor set of points are involved in the "priority points", and the set of candidate points available to them is all the candidate points in the scene, the chance of having extreme long associate is been significantly reduced. And once all the priority points finished their choosing process, it's considered one iteration.

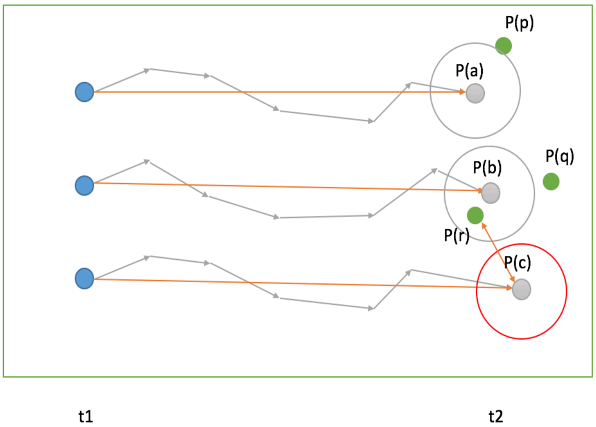

Figure 11. Scanning Range Algorithm. Since point "C" does not have any potential matching candidates in its scanning range, it has the priority to choose a nearest point first.

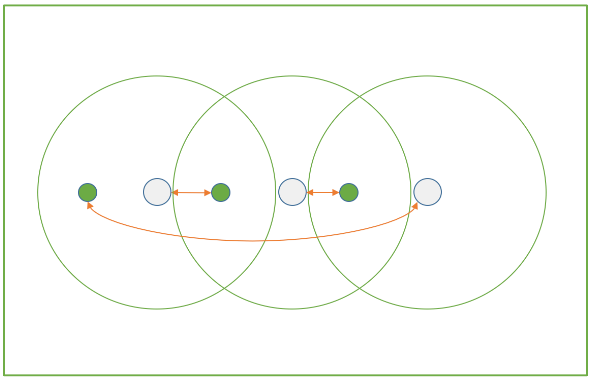

Figure 12. Fail case for inappropriate pick up sequence for non-empty scanning range points. When grey point in the middle is picked up first, followed by the middle points, resulting in the right side point have no chance but to pick up the left most point.

After one iteration of "priority matching", some of the human positions are been taken from the scene, this could probably result in generating new "empty scanning range" points. The "priority match" algorithm will stop when no "empty scanning range" points are left. And will be re-activated once new "empty scanning range" points occurred. By doing so, points that tends to have long association will have it reduced. The ideology of reducing long association is established on the heuristics that humans are unlikely to perform extremely long displacement in a crowd, the movement of each individual in a crowd tends to be alike and regular.

However, the proposed algorithm may still fail for in appropriate pick up sequence for "non-empty scanning range" points, as illustrated in figure 12. But the failed case would occur only when points are chained and they are picked up from a single direction, such scenario tends to be low in real tracking scenes, as pedestrians in most cases are distributed relatively sparse in the scene.

3.2 Implementation of Dense Optical Flow Tracking

The current and implementation of the algorithm tackles the aggregation of optical flow displacements. OpenCV dense optical flow API is been used to perform the displacement extraction. For the given position for each human being, it is a one pixel 2-D coordinates, and is locating at where the pedestrian's head is. The aggregation of optical flow displacement will be performed on that pixel. However, considering that each human's body does not stay "frozen" when moving, displacement could vary differently for different parts of human body, for instance the head itself could be turning to another direction and thus pixel displacements on it, which could be circular, may not represent the movement of the whole body. Instead of picking the displacement of a specific point, the average displacement of nearby neighboring points around the tracking point, where $\mathrm{D(avr)} = \frac{1}{n}\sum_{k=1}^{n} D(\mathrm{k})$, is used. A visual illustration of velocity filter square, which has 4 padding distance, is shown in figure 13. A few neighboring range is being tested, the resulting error is shown in table 1.

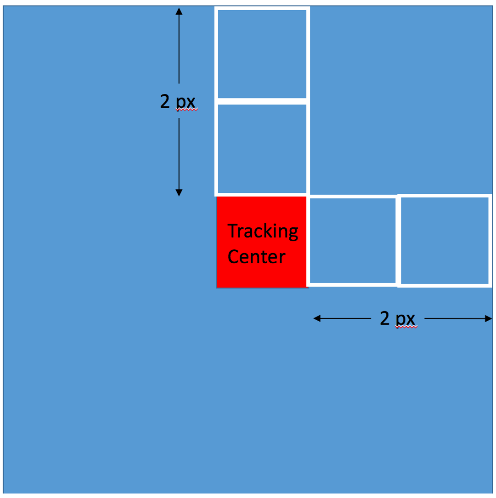

Figure 13. Displacement Filter Square. Each small square represents a pixel. The "tracking center" is initialized at the position given by the first human position frame. Here all 4 padding is 2-pixel wide.

	Initial frame, Time: 0 (s) Initial Position: (215, 300) Ending frame, Time: 74 (s) Ending Position: (599, 357)
All 0px	Final position: (602.492, 365.389) Difference: (+3.492, +8.389)
All 1px	Final position: (602.786, 365.389) Difference: (+3.786, +8.389)
Left & Right & Bottom: 1px; Top: 0px;	Final position: (603.055, 364.674) Difference: (+4.055, +7.674)
All 2px	Final position: (602.368, 365.166) Difference: (+3.368, +8.166)

Table 1. Test Result.

A demo of the optical flow tracking is shown in the table below. The total tracking time is around 70 seconds. The dense optical flow tracking algorithm performs a continuous tracking of a person (left bottom corner) moving in the video stream. The location of where the optical flow is tracking is mark by the red circle.

t = 0(s)

t = 11 (s)

t = 22 (s)

t = 33

t = 44 (s)

t = 55 (s)

At time = 0 seconds, the red circle is placed at the person's location. Later in the video stream, the tracking program no longer has the person's position in subsequent frames, the and red circle area's position is purely calculated from the aggregated displacement of the displacement filter. It can be seen that the position predicted by the dense optical flow always stays inside the person's body, also from multiple test result in table 1, the final error is around 3 to 8 pixels away from the actual position. Considering that each human in the scene is around 50 pixels long, the error generated by dense optical flow is only at around 12% of a human's height.

4 Conclusion

4.1 Summary of work

During fall semester 2016/17, half of the time were spent on reading relative research materials and getting familiar with existing crowd simulation tools. During the rest of time, an algorithm aiming to match people in two separate frames was proposed, and an implementation on tracking people 's position using optical flow was implemented with OpenCV using Python.

4.2 Limitations of work

For the "matching & association" problem, the algorithm proposed in this report is trying to solve only one of the mutation of the actual "matching & association" problem. The algorithm assumes that for the two frames for matching has the same group of people (i.e. same number of people, no people leaving or adding). There are 2 other cases. Case 1: the initial frame has less people than the final frame. Case 2: the initial frame has more people than the final frame. Under those two other cases, the current algorithm would need to be modified to taking care of the change in people's number and identity. Besides, the proposed matching algorithm could still fail in certain cases as shown earlier in this report.

4.3 Future work for next semester

For the rest of the final year project time, the first focus is on implementing the rest of the matching algorithm into a workable program with a reasonable performance. Furthermore, the algorithm need to be improved to handle the failing case stated in this report.

Another important task to complete is to either modified the current algorithm or to propose new algorithms in order to handle the 2 other cases of matching problem, in other words, the case of people leaving the scene or new people showing up in the scene.

5 References

Fagette, A. (2014). *Video Crowd Detection and Behavior Analysis.* Singapore.

Reisman, P., Mano, O., Avidan, S., & Shashua, A. (2004). *Crowd detection in video sequences.* Proc. of the IEEE Intelligent Vehicles Symposium.

Tu, P., Sebastian, T., Doretto, G., & Krahnstoever, N. (2008). *Unified crowd seg- mentation.* Proc. of the 10th European Conference on Computer Vision.

Hou, Y.-l., & Pang, G. K. (2008). *Automated people counting at a mass site.* IEEE.

Haralick, R. M. (1979). *Statistical and structural approaches to texture.*

Helbing, D., & Molnár, P. (1995). *Social force model for pedestrian dynamics.* Physical Review E.

Boghossian, B. A., & Velastin, S. A. (1999). *Motion-based machine vision techniques for the management of large crowds.*

Farneback, G. (2003). *Two-Frame Motion Estimation Based on Polynomial Expansion.*