Oregon State University

CS Capstone Design Document

December 4, 2018

Pedestrian Counting and Privacy Preservation

Prepared for

Oregon State University

Dr. Fuxin Li

Prepared by

Group 9

Pavement Prometheus

Ian McQuoid

Mazen Alotaibi

Miles Bigelow Davies

Stephanie Allison Hughes

Abstract

The City of Portland needs a tool for gathering data on traffic and pedestrians for internal and public use. The main issue that arose was privacy preservation and the removal of personally identifiable information from the data. The Pedestrian Counting and Privacy Preservation project serves to provide the city with stripped data and analysis of the data. This document's purpose is to present the methods and pieces of the project and the way that the pieces will be developed. The information covered will include the object detection model, facial detection and obfuscation, tracking models, and how the data will be analyzed and made accessible.

Contents

1 Overview

1.1 Purpose

Our team will design a pedestrian/vehicle detection model which is able to obfuscate all identifying features of pedestrians and vehicles for a given video feed. This will allow for storage of the video data without storing identifying information on the pedestrians.

1.2 Scope

The scope for this project is immediately to have a system that results in information on pedestrian movements that can be stored for open access by the public. An update that is not necessary, but is desirable, is the ability to provide data on traffic as well.

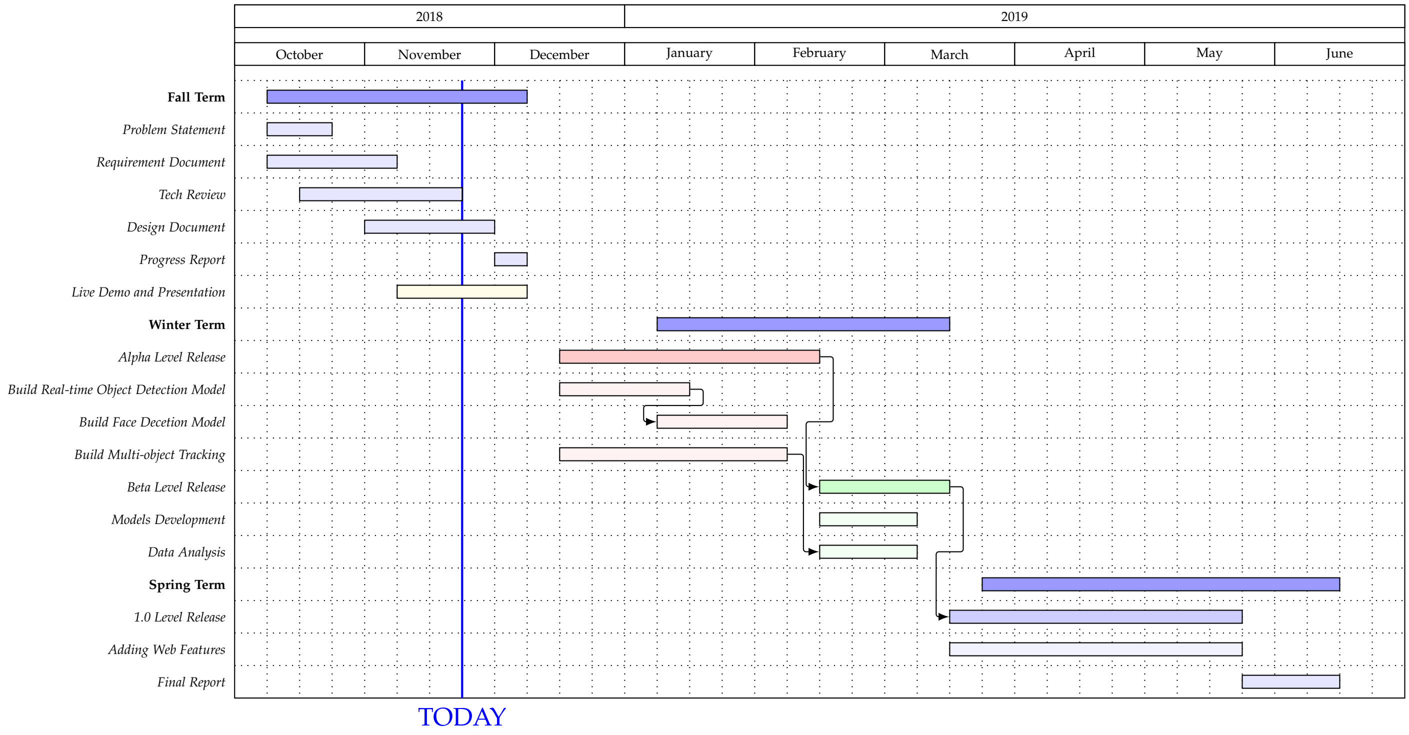
1.3 Glossary

Term	**Definition**
Convolutional Neural Network (CNN)	A class of deep, feed-forward artificial neural networks, most commonly applied to analyzing visual imagery [1].
Recurrent Neural Network (RNN)	A class of artificial neural network where connections between units form a directed graph along a sequence [2].
Long Short-Term Memory networks (LSTMs)	A special kind of RNN, capable of learning long-term dependencies [3].
You Only Look Once (YOLOv3)	A state of the art object detection model which can classify objects with a high degree of fidelity in a time sensitive environment [4].
mean Average Precision (mAP)	The mean for a metric denoting percentage of objects precisely identified, a ubiquitous standard used by object detection models [4].
Car Learning to Act (CARLA)	An open simulator for urban driving. CARLA has been developed from the ground up to support training, prototyping, and validation of autonomous driving models, including both perception and control [5].
VGGFace2	A large-scale face recognition dataset [6].
Facial Keypoints Detection	Facial detection through the use of multiple key points on a person's face [6].
Obfuscation and Mangling	Used interchangeably. The irreparable destruction of data. Specifically used in relation to identifying features of objects.

1.4 Design Stakeholders

The software described in this document, Facial Detector, and Obfuscator, is a project under the advisement of Chanho Kim (Georgia Institute of Technology) and Dr. Fuxin Li (Oregon State University). The client for this project is the City of Portland, which wants a proof of concept for a way to transform the data from their traffic cameras so the city may store the data without storing identifying information about the citizens in the footage.

1.5 Design Timeline



.

2 References

[1] G. Hu, Y. Yang, D. Yi, J. Kittler, W. Christmas, S. Z. Li, and T. Hospedales, "When Face Recognition Meets with Deep Learning: an Evaluation of Convolutional Neural Networks for Face Recognition," *ArXiv e-prints*, Apr. 2015.

[2] A. Shekhar, "Understanding the recurrent neural network," Available at https://medium.com/mindorks/understanding-the-recurrent-neural-network-44d593f112a2 (2018/04/14).

[3] C. Olah, "Understanding lstm networks," Available at http://colah.github.io/posts/2015-08-Understanding-LSTMs/ (2015/08/27).

[4] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," *ArXiv e-prints*, Apr. 2018.

[5] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, "CARLA: An Open Urban Driving Simulator," *ArXiv e-prints*, Nov. 2017.

[6] "Vggface2 about," Available at http://www.robots.ox.ac.uk/~vgg/data/vgg_face2/ (2018/10/30).

[7] C. kim, F. li, and J. Rehg, "Multi-object Tracking with Neural Gating Using Bilinear LSTM," *Oregon State Univesity*, Mar. 2018.

3 Design Viewpoints

3.1 Introduction

The Pedestrian Counting and Privacy Preservation project has four primary design viewpoints which will need to be considered and implemented. Our project will require a robust object detection system, which is able to identify objects in both an accurate and time-efficient manner. This object detection in turn will feed into a face recognition system that will ultimately be responsible for both improving detection accuracy and meeting privacy preservation goals. We will also be implementing a multi-object tracking system that will allow identity retention of an object across multiple frames. In addition to collecting important data on pedestrian and vehicle behavioral patterns, this tracking system will also assure validity in such measures as aggregate object counts; preventing a single object from being counted twice between frames. All of these components will feed directly into a data analysis and access system, which will be responsible for the extraction and serialization of any pertinent information filtered for storage on a server.

3.2 Viewpoint: Real-time Object Detection

3.2.1 Viewpoint Description

Real-time object detection is a necessary component of the pedestrian counting and privacy preservation system. Through the implementation a proper object detection algorithm, it is possible to aggregate data on pedestrian and vehicle traffic using a camera without requiring any video feed storage for further analysis. This would effectively maintain privacy for all parties picked up in a camera feed. Real-time object detection will be part of a collective effort by Miles Davies and Stephanie Allison Hughes, who will both be responsible for systems pertaining to object detection and face detection. Object and face detection, in turn, will be broken down into subsections involving setting up detection algorithm, and the requisite deep learning framework for its training and validation. Criteria involved for evaluating the success of object detection will relate to both the speed and accuracy of said detection. This will ultimately be a negotiation between the two criteria, as a faster processing time will temper our possible accuracy, while an improved accuracy will require a longer processing time.

3.2.2 Design Concerns

Our primary design concerns for our real-time object detection relate to both the speed and accuracy of our model. Because our system will primarily involve interesting information extraction while simultaneously excluding any identifying details, we require an object detection algorithm that maximizes both detection accuracy and detection speed. An object detection algorithm which meets both requirements will allow us to eschew any form of video storage for more conventional methods of processing, as all necessary information can be extracted in real-time. By eschewing video storage, this system will have no persistent data that contains personally identifying information which might violate the privacy of any party recorded in the camera feed.

3.2.3 Design Elements

Design elements include the setup and choice of a base object detection algorithm, a deep learning framework for both the training and validation of our selected detection system, a comprehensive dataset to train the model on, and any requisite computer hardware for training the model. After selecting and implementing a base object detection algorithm in our chosen deep learning framework, our team will need to download a dataset to begin training and validation our detection algorithm against. Our team can then investigate methods to improve either the accuracy or speed of our model to compare against the base setup; whether through the reduction of object classes available for detection or through some novel restructuring of the model's neural networks or weights. For the selection of a base object detection algorithm to implement, our team research has indicated You Only Look Once (YOLOv3) is the best choice for maintaining an adequate detection accuracy with an impressive framerate speed[4]. The speed and accuracy of this detection system appears to be the result of a union between the DarkNet 53 system, for feature extraction, and Feature Pyramid Networks (FPN), which uses a bottom-up and top-down pathway for improved small object detection accuracy. Our selected deep learning framework, in turn, will be the open-source machine learning library PyTorch. We selected this deep learning framework for a variety of reasons, including ease of use, active community support, and the customizability of its Neural Networks.

3.2.4 Relationship

The real-time object detection system will be the first layer of processing performed on a camera feed for data extraction; and will be interfaced directly with the facial detection, tracking, and data analysis and access systems. Through a combination of the object, facial, and tracking detection we will be able to present information for data analysis and serialization for extraction and storage to a server. Through the interfacing with the face and tracking subsystems, any shortfall in the accuracy of the real-time object detection system will ideally be assuaged. Allowing for a comprehensive, accurate, detection system performing in a real-time environment.

3.3 Viewpoint: Face Detection

3.3.1 Viewpoint Description

Our group will conduct facial detection using a feature-based method by training a convolutional neural network (CNN) and validating our results. For the feature-based method, the facial features are detected through examining the edge, intensity, color, shape, etc. of a feature. Training of a facial detection model allows our group to use the video footage from the city of Portland and define the faces we are looking for. The data we use is from low-quality footage where the environment can take different forms, so it is important that we can train the data in those various settings. The validation of facial detection is responsible for ensuring that the results from our program are true to the information that is actually within the video footage. The final criteria the facial detection program is to achieve at least 70 percent accuracy of detecting faces.

3.3.2 Design Concerns

Main concerns of the design are the the quality of video to work with and ensuring consistent detection accuracy. The video footage provided by the city of Portland is low quality and is taken from various perspectives. Low quality footage may not allow us to make out distinct facial features, so we will need to make sure we train our model with different quality face images. With the video taken from different perspectives, the CNN must also be trained with multiple sides of the face from several different camera angles. Another design concern is ensuring high accuracy of results within various times of day and weather conditions. The video footage is taken from outside environments where rain, fog, snow, and other elements may alter the visibility of the scene. This stresses the importance of training the program with many different environments and consistently testing the accuracy of the detection to understand what may need to be adjusted. To avoid these design concerns, our group must have in-depth CNN training and validation.

3.3.3 Design Elements

Design elements of creating a facial detection program includes training a CNN and validation of results. To implement feature-based facial detection, first the key facial features that will be tracked must be determined. Our group will take snapshots of the traffic video footage and use those frames to train with initially. The image is examined to find common variances in the face, grouping the components together to ensure they match. By having a threshold of the number of key points of features detected, we can then label that area as a detected face. Using the feature-based method is more likely to detect facial features despite the orientation of the face. If a person is faced to the side or in a different direction, the program is not fazed as it is not looking at the overall face but the distinctive features. This makes the speed of the program quicker as it would not need to run extra functions to detect faces from different angles, just the primary program. The results of the program should output a JSON file containing the detected facial feature data.

Once the CNN is trained to detect faces, the results are validated by testing the accuracy between our program versus that of industry-leading facial detection software, Face by Microsoft. Face uses the distinctive features of a face to mark a section of an image to be a person.The software examines 27 different elements of a face including the face edges, hair, eyes, eyebrows, mouth, nose, chin,and more. Each facial feature is measured with an x and y value delivered in a JSON file. This technology has a 70 percent accuracy, focusing on the facial features, making it a very accurate implementation. With the same accuracy goals and output type as our program, it makes it perfect to test against.

3.3.4 Relationship

Facial detection plays in integral part in our project as the faces must be detected to protect the privacy of the pedestrians. A pedestrians face is a unique, identifying aspect of a person, so if it is not obscured, the person could be easily tracked. If just the pedestrians were detected and the entire bodies were obscured as a block, larger amounts of video footage would be lost. By just detecting the faces, the crucial identifying information is obscured without altering too much of the footage. The point of the project is to keep the privacy of the pedestrians while retrieving of traffic data. For people to keep their privacy and live free lives, faces must be detected to be obscured.

3.4 Viewpoint: Multi-object Tracking

3.4.1 Viewpoint Description

Multi-object tracking tasks to localize multiple objects, maintaining their identities, and predict their individual movements given an input video. Mazen Alotaibi will be the main designer of the subsystem relating to building the multi-object tracking system supervised by Dr. Fuxin Li and Chanho Kim because of their research experience in developing multi-object tracking computer vision systems. The main purpose of this system is to collect and pre-process data to describes the behavior of pedestrians, bikers, and cars in traffic. The collected data should include the bounding boxes of all relevant objects as well as unique identifiers for each object in the image. The expected outputs of the computer vision system are heat maps of objects, pose estimate, and movement tracking to be used in the data analysis stage.

3.4.2 Design Concerns

Our main concerns in the multi-object tracking system are related to generating data-sets for training the computer vision system, building the computer vision system, and validating the result of the computer vision system over rain, snow, and other weather conditions as well as changes of lighting conditions. Generating data-sets is challenging because of two reasons: legality of using videos without the consent of captured pedestrians and labeling the captured objects within video frames.Because of the complexity of recording unconsenting individuals, it is crucial that preserving privacy is our highest priority. Our system should seek to not only localize people within the image, but also to censor their faces in order to preserve their anonymity. Because we might participate in a Multiple Object Tracking competition to test our computer vision system, we expect our computer vision system to have high accuracy and speed when tracking multiple objects within any video format. Lastly, because our computer vision system will be used on surveillance cameras that monitor inner cities and highways, our computer vision system's accuracy of detection and tracking stay be high over rain, snow, and other weather conditions as well as changes of lighting conditions.

3.4.3 Design Elements

Due to the requirement of having the computer vision system ready for production after develop, our optimal solution for the development framework is PyTorch. The preview version PyTorch 1.0 was chosen as our framework for our computer vision system due to its speed and simplicity. Pytorch 1.0 utilizes the machine learning framework Caffe2 for its backend operations. PyTorch and Caffe2 are collaboratively developed and managed by Facebook and Microsoft and is built with scalability and simplicity in mind. In addition, PyTorch allows us to import our computer vision system into production C++ code without the need to rewrite or change any of the PyTorch code that has been written in Python. Moreover, the expected output of the computer vision system should be in JavaScript Object Notation (JSON) format because that what the City of Portland expect when obtaining the data for production use. For the development of the computer vision system structure, our computer vision system will use YOLOv3 for object tracking and Dr. Li's computer vision system [7], which uses LSTM cells to obtain the spatial information of detected object over time.

3.4.4 Relationship

Multi-object tracking is the most complex part in our project because obtaining more useful information that describes pedestrians, bikers, and cars in traffic is needed for the data analysis stage. In addition, developing the a decent multi-object tracking system for our project requires a lot of development and understanding of building computer vision models.

3.5 Viewpoint: Data Analysis and Access

3.5.1 Viewpoint Description

Data analysis and access is a key subsystem in fulfilling the end goal presented to our group by the City of Portland. Ian McQuoid and Mazen Alotaibi will be the two designers of the subsystems relating to data aggregation, analysis, and access. The main purpose of this system is to provide the City with reasonable access to the data provided by sensors around the city, after stripping all personally identifying information, and to make the data provided more intelligible by parsing the video or photographic information gathered into a serialized format. This formatted data will allow the City to make decisions about the roadways and traffic with greater speed and accuracy. The criteria our group will use to interpret and evaluate the system will be concerned with the ergonomics relating to the access of the information and the accuracy to ground truth models for the analysis and representation of the serialized data.

3.5.2 Design Concerns

Our main concerns in the Data subsystems are related to the persistent data structure, the data access scheme, and the data content. The City of Portland largely uses the JavaScript Object Notation (JSON) as a key standard for transmission of data objects between departments. Because of the City's JSON use, the City of Portland requires analyzed data to be in JSON form with a web-based interface. The first concern raised is the requirement relating to the basic data format, while the second concern relates to the structure the analyzed data will be stored in. The data content is the broadest concern in the design process. The content of the analyzed data must be verifiable, accurate, and interesting. The basis for the content concern relates directly to the applications for the analyzed data. Information relating to traffic makeup, size, lane usage, and speed are all directly applicable to traffic analysis, so final data content in the database must provide comparable or directly related information.

3.5.3 Design Elements

The direct data access format must be in JSON form as defined by the City's concern. The format specification affords the team the opportunity to use the MongoDB persistent data structure for storing final data-sets. The MongoDB database program has default interfaces in place which will be the base for access to stripped and analyzed data. The resulting system will be presented as a web-based interface backed by the MongoDB program for storing and accessing data. The interface will be implemented using a general back-end written in Node JS which will create a simple HTML web page for accessing and downloading JSON-formatted data from a MongoDB database. The presented solution will directly address both of the primary concerns presented by the City of Portland. The final design concern is related to the final content of our stored data. The primary pictorial and video data will be presented directly to the user and will not be stored in the database; however, analyzed and stripped information needs to be stored in the database. Before data can be stored in the database, pertinent information needs to be taken from the photographic or video data collected by the City's sensors. The data system will use a mixture of direct storage of information and inferential stripping techniques to gather the required information. Primarily, the object detection and tracking systems will provide concrete data on number, makeup, and trajectory of the vehicles and pedestrians. The concrete data will be stored directly in the MongoDB structure. Further, the data analysis subsystem will be comprised of categorization, comparative, statistical, traffic theory based, and neural network algorithms. Incoming data-sets will first be split into geographic and time-stamped categories and will be passed into the predictive algorithmic structures for analysis. The outputs from the system will be comprised of predictions on congestion, traffic incidents, and recommendations for traffic flow organization. The final results will be stored in the database structure for access.

3.5.4 Relationship

The data analysis and access system will directly interface with the three detection and tracking systems. All three of these systems will present the data system with uniform information in the form of both photographic or video data as well as serialized information about the geographic and temporal placement of the data-sets. When possible, serialized information such as the number of detected objects will be passed to the data system and parsed, categorized, and stored by the access subsystem. All non-serialized data will be passed directly to the analysis subsystem, as the photographic and video data will not be accessible through the MongoDB interface. The two subsystems of access and analysis will interface with each other in a symbiotic relation with the analysis system pulling data from the access system as well as pushing analyzed and serialized data into the database.

3.6 Conclusion

By addressing and implementing each of these viewpoints we will create a robust system capable of collecting census data on pedestrians, vehicles, and their traffic patterns while simultaneously protecting any personally identifying information from being stored and put at risk. This system will comprise of an object detection system which first identifies pedestrians and vehicles alike. While a face recognition will be responsible for detecting pedestrians for masking and improved detection accuracy, and a multi-object detection system will retain memory of unique objects to both maintain data integrity and collect pedestrian and vehicle behavior. The data in turn will be piped though a data analysis and access system responsible for the serialization and ultimately storage of data on a server. This system, fully realized, could be implemented on a city by city basis to provide informed decisions on transit planning, census counts, and pedestrian behavior while simultaneously preserving the privacy of all parties involved.