Detecting Pedestrian Crossing Events in Large Video Data form Traffic Monitoring Cameras

Weijia Xu
Texas Advanced Computing Center
The University of Texas at Austin
Austin, Texas, USA
xwj@tacc.utexas.edu

Natalia Ruiz
Center of Transportation Research
The University of Texas at Austin
Austin, Texas, USA
nruizjuri@mail.utexas.edu

Kelly Pierce
Texas Advanced Computing Center
The University of Texas at Austin
Austin, Texas, USA
kpierce@tacc.utexas.edu

Ruizhu Huang
Texas Advanced Computing Center
The University of Texas at Austin
Austin, Texas, USA
rhuang@tacc.utexas.edu

Joel Meyer
Austin Transportaation Department
City of Austin
Austin, Texas USA
Joel.Meyer@austintexas.gov

Jen Duthie
Austin Transportaation Department
City of Austin
Austin, Texas USA
Jen.Duthie@austintexas.gov

***Abstract*— Pedestrian safety on the road is a priority for transportation system managers and operators. While there are a number of treatments and technologies to effectively improve pedestrian safety, identifying the location where these are most needed remains a challenge. Mid-block locations, where safety countermeasures are often needed the most, are typically harder to monitor. Current practice often requires manual observation of candidate locations for limited time periods, leading to an identification process that is often time consuming, lags behind traffic pattern changes over time, and lacks scalability. As a result, target locations are often selected reactively, after serious traffic incidents reveal an underlying safety issue. We propose an approach to use data collected by existing traffic monitoring cameras to automatically identify pedestrian activities on the road. We propose an algorithm to detect pedestrian crossing events based on the detection of individuals on individual video frames using a deep neural network model. Resulting pedestrian locations and movement trajectories can be visualized on a background image, which is automatically extracted at the analyzed location from the video. We demonstrate and evaluate our approach with a real-world use case. The case study considered in this work uses cameras owned by the City of Austin, Texas to study pedestrian road use before and after the deployment of a pedestrian-hybrid beacon. We explore qualitative and quantitative metrics to describe pedestrian activity and corresponding changes, which may be used to prioritize the deployment of pedestrian safety solutions, or evaluate their performance. We compared the number of crossing events detected per hour with manually reviewed results from a selected day. The result shows 67% overall accuracy, although we observe significant variability across times-of-day. Despite observed limitations, our work illustrates how the value of existing traffic camera networks can be augmented beyond everyday traffic monitoring, and used to collect valuable information on road usage by pedestrians.**

Keywords—big data, traffic camera video analysis, pedestrian safety, deep learning.

I. Introduction

Pedestrians on the road are the most important and yet the most vulnerable link within any transportation system. Pedestrian safety is of critical importance for transportation system managers and operators, and safety measures require frequent revisiting due to the evolving nature of traffic patterns. Over the past years, the total number of pedestrian fatalities has been on the rise, which motivates local transportation agencies to implement additional proven countermeasures to improve pedestrian safety. As approximately 75% of pedestrian fatalities occur at mid-block locations, much focus is placed on strategies to facilitate safer pedestrian crossings. However, a major challenge is to proactively identify mid-block locations where those safety improvements are needed the most, and to do so before fatal incidents take place.

In this paper, we propose a novel approach of identifying and quantifying pedestrian crossing events from existing roadside traffic camera videos. The number of crossing events is expected to be a reflection of the volume of pedestrians crossing at a particular mid-block location, which is an important variable when identifying the need for specialized treatments, such as pedestrian hybrid beacons. Our approach automatically analyzes the content of video data from existing traffic cameras using a semi-automated processing pipeline powered by the state-of-art computing hardware and algorithms.

Our work is directly motivated by real-world needs within the Austin, Texas urban area. Over the past decade the City of Austin (CoA) has experienced fast population growth and has installed pedestrian-hybrid beacons (PHB) at over 75 locations to improve mid-block crossing safety. PHBs have been shown to reduce serious injury and fatal crashes by 15%, and pedestrian crashes by nearly 70% [1]. Each installation location is characterized by documented pedestrian volumes, crash history, and long distances between safe crossing opportunities, among other factors. When studying potential PHB locations, the CoA has traditionally relied on manual observations to quantify pedestrian movements, such as the number of crossings over a given time period. The existing process requires significant effort from CoA staff, and can only acquire limited information at selected locations and for short time periods. Manual methods lack the scalability needed to quantify the evolution of pedestrian behaviour over time, explore a larger set of candidate locations, or systematically evaluate the impacts of a treatment such as PHB.

The work we present here utilizes automated approaches to effectively recognize, analyze, and store records of pedestrian activities over time from video collected by existing traffic monitoring cameras. The collection and analysis of video data at selected locations provides an opportunity to analyze pedestrian movements with verifiable account of road user behavior over extended time periods, and reduces the need to rely on ad hoc decision making [2].

We have previously developed a traffic camera video processing pipeline to automatically recognize vehicle trajectories using selected traffic cameras in the CoA camera network [12]. We have applied our framework to support vehicle flow analysis [12] and pedestrian detection [3]. Several technical challenges remain to accurately characterize pedestrian crossing location and quantify the frequency of crossing events. Regular roadside cameras are installed to have wide and deep fields of view, including:

- Pedestrian activities only occupy a small portion of the view, and at many locations are only present sporadically.
- Pedestrians appear smaller in size than cars, and are more frequently subject to obstruction from other objects within the scene.
- There are also cases when drivers visible through open window vehicles can be falsely identified as pedestrians in the road.

Building upon our prior work, we have improved our pedestrian detection and tracking algorithm to address these limitations. Our approach now defines an area of interest within the field of view for the specific targeted locations, the expected pedestrian movement direction and estimated pedestrian travel distance. We also propose additional visualizations to summarize our analyses and provide and intuitive representation of pedestrian activities over time to decision makers. We implemented our improved algorithm to assess the effectiveness of a recent PHB installation in Austin. Specifically, we considered whether the PHB installation changed the number of identified crossings or the position of those crossings along the roadway. Additionally, we assessed the accuracy of the improved algorithm by comparing the algorithm-identified crossing counts to the number of crossings identified manually from selected video recordings.

The rest of the paper is organized as following. In Section II, we review related work on traffic camera video analysis and pedestrian safety. Section III reviews our previously developed video data aggregation framework. In Section IV, we describe the algorithms and techniques used for crossing event detection and visualization. Section V details the result from our analysis of changes in pedestrian crossing behavior in response to a PHB installation. We conclude and discuss our ongoing work in Section VI.

II. Related Work

A. Computational Analysis of Traffic Camera Video

Due to their low maintenance and operational cost, video sensors, such as pan-tilt-zoom (PTZ) cameras, are commonly installed along freeways and arterial streets [4]. However, the use of video data from these cameras for system performance and safety assessment or strategic planning is not widespread. Transportation Management Centers (TMCs) primarily use traffic video data from roadside cameras to identify incidents, prepare the response for emergency situations, manage traffic in special events, and dispatch technicians for maintenance [5]. The video data is also used to manually conduct traffic studies, including collecting traffic counts by mode of transport (e.g. auto, transit, pedestrians), turning movement vehicle counts for traffic signal timing applications, and conducting safety analysis by observing the behavior of traffic in weaving zones [6]. While traffic video data analysis software tools exist, they are mostly used to support real-time traffic operations, commonly focusing on one type of analysis, and often deployed in dedicated, specialized hardware. Examples of video data use include safety analysis for intersections and corridors [7-9], identification of unusual events on corridors, such as wrong-way driving and stalled vehicles [10], generation of traffic statistics including counts and queue lengths, and analysis of vehicular emission by estimating traffic speeds [11]. Such applications are usually labor intensive, and impractical for large-scale implementation.

B. Pedestrain Detection from Video

The detection and tracking of main road users (e.g. pedestrians, cyclists, and vehicles) remains a hot topic in the field of computer vision. The research focus is often on developing an automated process to identify object trajectories, thus avoiding time-consuming manual processing. Automated video data analysis is considerably complex at urban signalized intersections, characterized by a mix of traffic conditions and the presence of various road user types. While significant work exists on the topic of vehicle tracking, fewer studies look into pedestrian tracking. The latter is more complex than vehicle tracking because of pedestrians' non-rigidity, more varied appearance, and less organized movements. Pedestrians have fewer spatial constraints than vehicles and may change their direction of movement frequently, while vehicles have to follow certain lanes with limited turning options [14]. Additionally, pedestrians often move in groups, making even harder the detection and tracking of individual movements [17] [4]. The need for methods and applications for pedestrian tracking and classification of road users has been highlighted and addressed in several articles [15-18].

III. Background and Previous Work

A. Video Collection and Processing Pipeline

We have developed an automated video processing framework that separates the video analysis process into two distinct parts: object recognition and analysis of the identified objects [3]. We use convolutional neural networks to detect and track the motion of objects from each frame in the video stream, and then store and process information using Spark [12, 19]. By combining the best practice of object recognition through deep learning and big data processing for those two parts respectively, the framework can efficiently process large-scale traffic video data automatically and meet evolving analytic needs over time.

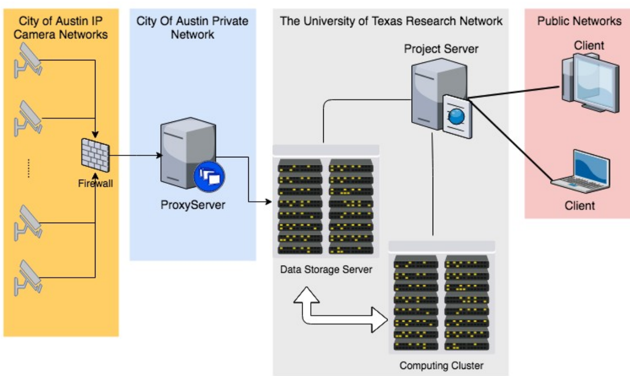

Fig. 1. Camera access and processing pipeline overview.

To implement the framework, we have set up a multi-systems cross-domain video aggregation and analysis pipeline (Figure-1). Raw videos are originated from IP cameras in the CoA private network, which has limited accessibility. To overcome the latter, the CoA set up a proxy server to forward selected video feeds from the IP cameras to a storage cluster hosted at the Texas Advanced Computing Center (TACC). The recorded video can be then be processed by a high-performance computing cluster at TACC. Processed data is saved in a storage server, which is accessed by our project server for results dissemination purposes. The project server also hosts tools and scripts to schedule video recording and processing tasks. The proposed processing approach consists of two main steps: the *video content recognition* step identifies and labels all physical objects from original input video files using a deep-learning based algorithm; the second step is *object tracking*, which “follows” each recognized object across all frames in the input video.

B. Video Content Recongtion

Our video content recognition process is based on Darknet, an open source library of image recognition [12, 13]. The core algorithm utilizes a convolution-neural-network-based object detection system, YOLOv2, to analyze each frame of an input video [13]. For each frame, the algorithm outputs a list of objects including their location in the frame, class label, and confidence of recognition. We have limited recognition to seven class labels that are most relevant, including person, car, bus, truck, bicycle, motorcycle, and traffic light. To improve algorithmic performance and maximize utilization of multi-node computing clusters, we have also adapted the YOLO implementation for parallel execution [12]. Our implementation enables parallel object recognition on multiple frames using p-thread within individual compute nodes, and using MPI for inter-node communication. Specifically, one thread is used to pre-fetch *n* frames, while *n* extra worker threads are assigned to labeling. Since each worker thread is independent, near-ideal linear scaling can be achieved for longer videos [12]. For video recordings from different times/locations, multiple video files can be processed independently across multiple nodes concurrently. A non-maximum suppression (NMS) algorithm with the locally maximal confidence measure is used to remove unnecessary/duplicated objects. In addition to content recognition, the framework outputs a background image (i.e. non-moving features) from each video recording. For more details about the original YOLO algorithm and our implementation, please refer to [12] and [13], respectively.

IV. Methodologies

Although our previous work facilitates video aggregation and has been successfully applied to traffic flow analysis, we observed some limitations when implementing the approach to pedestrian location detection and tracking with the goal of detecting mid-block crossing events. After reviewing the outcomes of our previous implementation, we identified several factors affect crossing event detection performance. We have developed an improved tracking algorithm and implemented features to further improve crossing event detection.

A. Tracking Pedestraints

<u>Pseudo code for Pedestrian Tracking</u>

Input: N = {n_{ij} | i: frame index, j: object index} as the set of recognized objects found in each frame

Output: T= { t_{ij} | i: trajectory index, j: frame index of object within this trajectory} as the set of objects stored by a list of trajectories

Initialize *T* with each object found in the starting frame
for each n_{ij} in *N*
  for each t_k in *T*
    dists <= distance(n_j.location, pred(t_k, *i*))
  if min(dists) < *threshold(frame diference)*
    add n_{ij} to $t_{argMin(min\_dists)}$
  else add n_{ij} as a new trajectory

Fig. 2. Pseudo code for tracking pedestrians.

To track small moving objects, such as pedestrians in particular, we implemented an approach based on predicted positions of objects from previous frames (Figure 2). The algorithm is initialized with the set of recognized “person” objects in each frame. Each object record also includes information about the object location and frame number from the video. At the start of the video frame, for each recognized object, we initialize a trajectory for that object. Recognized objects in the subsequent frame are matched to the closest object (whose bounding box usually overlap with the current object) from the previous frame. Once a trajectory has more than two distinct positions, direction and velocity of the trajectory can be estimated based on positions of latest objects in that trajectory. For subsequent frames, we compute the distance between all newly identified objects with the predicted positions of existing trajectories from previous frame. We used a pre-defined threshold to avoid connecting an object that might be too far away from the closest trajectories [3]. However, the length of trajectories is sensitive to this pre-defined threshold, with smaller values leading to shorter trajectories. Although the length of the trajectories does not affect the visualization of pedestrians location, it has an impact on the detection of road crossing events. To address this limitation, we also implemented a threshold function (Figure 2, line 5). The threshold function will take consideration of elapsed frames between the last frame index recorded in the trajectory and the frame index of the considered object. If the two indices are further apart in time, the threshold is larger than if they are close. When the distance between an object and a trajectory is larger than the calculated threshold value, the algorithm generates a new trajectory. Otherwise, the object position is added to the trajectory whose predicted position is the closest.

We have implemented this algorithm using the Spark big data processing framework to read and process results files from multiple video recordings and export the detection and tracking of pedestrians as a structured file. A complete list of all tracked objects with corresponding detailed information is stored in a structured data file to derive further information, such as direction of movement.

B. Crossing Event Detection

The original traffic camera video includes multiple pedestrian activities, such as walking down the street, waiting for the bus, and crossing the street. To further isolate mid-block crossing event, we have implemented filters for variables such as location, direction and track length.

Fig. 3. Area of interest can be defined as a polygon and highlighted in red.

Location filter The location filter can be used to select only activities within a user-defined polygon. Trajectories are filtered to present only those that correspond to pedestrians that crossed road. The user can define a convex shape through a list of coordinates for each boundary point. Figure-3 shows an example of a location filter. In Figure-3, the area of interest, defined by the user is represented with a red polygon.

Direction filter each inferred trajectory is classified as horizontal or vertical, and the direction filters can be used to focus the analysis only on the relevant direction. For crossing event detection, we focus on horizontal trajectories, e.g. those that mover across the road. This filter is effective to avoid considering the trajectories of cyclists and motorcyclists in our analyses. Such road users are also recognized as "persons" by the video recognition algorithm, but since they typically move parallel to the road, they may be distinguished from pedestrians crossing the street using the direction filter.

Distance filter As mentioned above, our tracking algorithm sometime generates short tracks. The short tracks are usually caused due to missed pedestrian detection over several frames due to a temporary obstruction of the line-of-view, or to low video quality due to environmental factors. The distance filter allows to focus the analysis on longer trajectories.

C. Visualizing Crossing Event Locations

To help traffic engineers understand pedestrian activities over time, we have implemented several visualization techniques to provide a summary of pedestrian detection and tracking over time.

1) Colored dot plot to show pedestrian locations.

We use a colored dot plot view over the background image to create a visual summary of pedestrian detection over time. In this visualization, only the center location of each detected pedestrian is shown as a colored dot. The dot color can be used to further characterize the detection event. Figure 5 illustrates the visualization of pedestrian locations by time of day.

2) Heatmap with focal statistic smoothing

A heat map visualization may be used to highlights the local significance of a region relative to surrounding regions. In this effort we have adopted a focal statistic smoothing approach to generate the heatmap presented in Figure 4. The selected smoothing technique recalculates raw input data using the weighted neighborhood sum defined in a disc kernel matrix. The approach transforms the raw detection results, highlighting the statistical center of activities through erode (shrink low values) then dilate (expand high values) processing. Figure 4 illustrates such transformation, contrasting the dot plot representing the number of pedestrian detections per pixel (top) with the outcomes of the focal statistics smoothing (bottom). Locations with high expectance of pedestrians are highlighted in red.

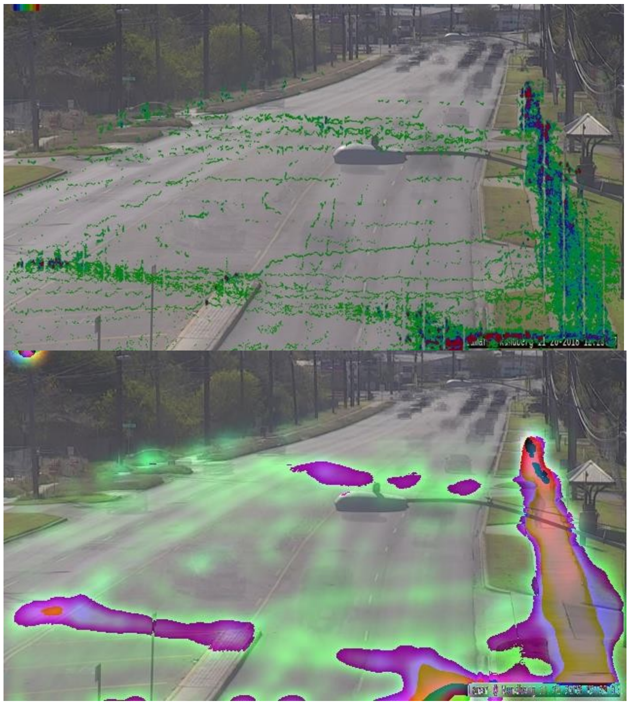

Fig. 4. Focal statics smoothing heatmap of pedestrian detections. On the top, the raw values of pedestrian detection events per pixel. On the bottom, morphed visualization to highlight local centers of activities.

3) Calendar view of pedestrian detection over time.

To facilitate comparisons of pedestrian moving patterns by time of day over multiple days, we propose to visualize key metrics in calendar format. Graphs such as scatter plots, histograms, and violin plots can be generated based on daily data, and presented in a calendar-like grid layout. This view gives an intuitive summary of data over different time scales. Figure 9 presents an example of calendar visualization, which is be discussed further in next section.

D. Pedestrian activity metrics

We have computed and used two quantitative metrics to characterize pedestrian activity.

Pedestrian detections by minute provides a representative estimate of pedestrian counts by minute by filtering the raw detection results. For each frame, we count all objects classified by the deep learning algorithm as pedestrians which have bounding boxes centered in the roadway. Because we do not assemble trajectories for this analysis, we do not know how many unique individuals are present in the roadway during each minute. Instead, we estimate the minute-level pedestrian count by selecting a single representative frame from each minute of video. The representative frame is the frame with the highest count of pedestrians. This allows us to determine the maximum number of pedestrians present in the roadway without accidentally counting some individuals multiple times. This metric is displayed in Figure 9.

Hourly pedestrian crossing activity The hourly pedestrian crossing activity metric results from aggregating minute-by-minute pedestrian crossing events by hour, producing a time series that illustrates how activity changes over the course of the day and between days.

V. Case Studies

A. Data Overview

TABLE I. DATA COLLECTION SUMMARY (2019)

	Anderson Ln. @ Burnet Rd.		**Lamar Blvd. @ Rundberg Ln.**	
Date Time range	*Sep. 19 – 21 (Wed – Fri) Sep. 29 – Oct. 2 (Sat – Tue)*	*Feb 12 – 15 (Tue – Fri) Feb 23 – 25 (Sat – Mon)*	*Feb. 14 - Mar. 2*	*Mar.16 -21*
Average size per video file	*~ 300MB*	*~ 300MB*	*~315MB*	*~315MB*
Total size	*1300GB*	*1400GB*	*941GB*	*522GB*
Average durations	*15 mins.*	*15mins.*	*15 mins.*	*15 mins.*

We have used our implementation to analyse pedestrian street-crossing patterns at the intersection of Anderson Lane and Burnet Road (hereafter Anderson location) in Austin, Texas. For the Anderson location, analyses were conducted before and after the activation of a PHB device intended to allow for safe pedestrian crossings in the area near an active bus stop. We have automated video captures in 15 minutes interval at each location during the period from September 2018 to May 2019. Since the camera angle and direction can be dynamically adjusted by staff at CoA for other purposes, video recording not of interests to this study were discarded. The time range and size of video recordings selected for computational analysis are summarized in Table-1.

B. Pedestrain Activities during the testing period.

The pedestrian activities for selected periods are summarized in Figure 9 (September 2018 before PHB activation) and Figure 6 (February 2019, after PHB activation). Figure 9 presents a time series of the maximum pedestrian counts for each one-minute slice in all analyzed videos (blue dotted line). The corresponding 30-min rolling average is shown in red in those plots as well. These two visualizations illustrate the magnitude of pedestrian activities over different times of day and across multiple days. The results suggest a relatively consistent level of pedestrian activity, with an average of 2 pedestrian detections per minute both before and after the PHB device activation.

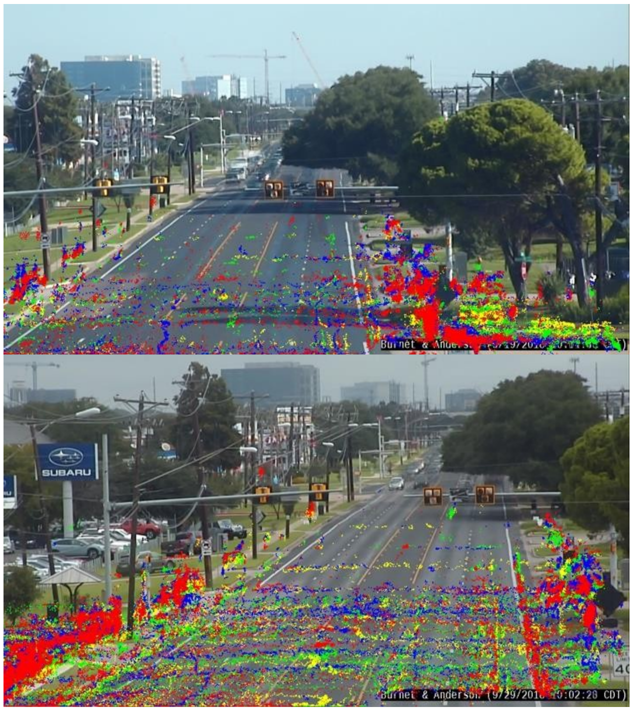

Fig. 5. Dot plot view to show locations of pedestrian activity detection before the PHB installation. The activities are colored for four time periods of the day: Yellow: 7-10 am; Green: 10:00-13:00; Blue: 13:00~16:00; Red: 16:00 ~19:00 (top) Sep 19-21 2018 (Wed ~ Fri) (bottom) Sep. 29 ~ Oct 2 2018 (Sat ~ Tue)

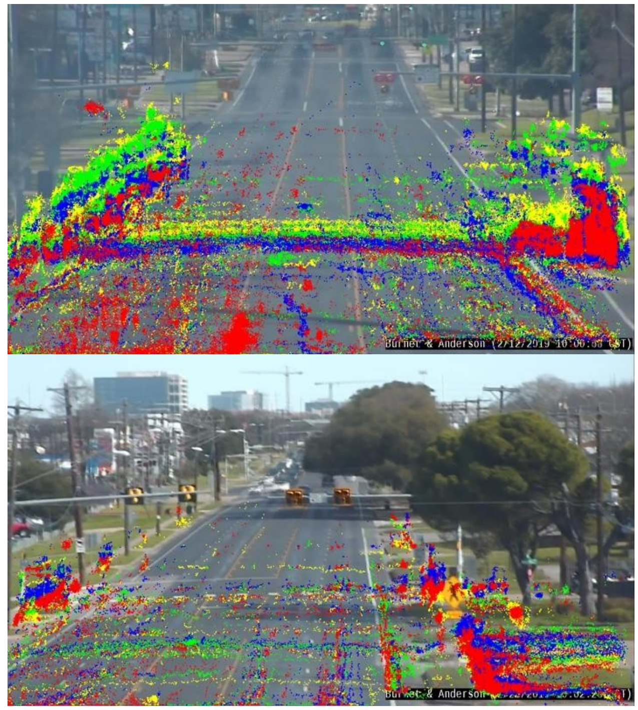

Fig. 6. Dot plot view to show locations of pedestrian activity detection after PHB installation. The activities are colored for four time periods of the day: Yellow: 7-10 am; Green: 10:00-13:00; Blue: 13:00~16:00; Red: 16:00 ~19:00 (top) Feb 12-15 2019 (Tue ~ Fri) (bottom) Feb 23 ~ Feb 25 2019 (Sat ~ Mon)

To visualize the location of pedestrian activities, we plotted all pedestrian detection in Figure 5 and Figure 6. In Figure 5, the analysis is divided due to slightly different views of camera adjustment. Figure 5 shows the activity summary before the PHB activation. Figure 6 shows the activity summary after the PHB activation. The activities are colored for four time periods of the day: Yellow: 7-10 am; Green: 10:00-13:00; Blue: 13:00~16:00; Red: 16:00 ~19:00. The Yellow and Red groups correspond to peak traffic hours while the Blue and Green groups are for off-peak hours. In both cases, major pedestrian activities are centered around two bus stops on both sides of the street, suggesting that this is the major motivation for road crossing.

The trend of concentrated crossing locations shown in Figures 5 and 6 is further highlighted in the smoothed heatmap view (Figure 7). Deep blue color indicates high likelihood of pedestrian crossing while green color indicates less likely location to observe pedestrian crossing. The crossing events seem more concentrated at the PHB location.

C. Evaluation

To evaluate the accuracy of the crossing event detection, we used four reviewers to analyze 38.5 hours of video at the Anderson location traffic camera. Each reviewer worked independently on a different day of video. Collectively, reviewers took 68 hours to complete the work, with an average processing time of 26 minutes o review a single 15-minute video. Three full days of video, with 7-8 hours of video per day, were reviewed: September 19 2019, September 29 2018, and February 13 2019. Two partial days of video were also reviewed: February 12 2019 and February 14 2019.

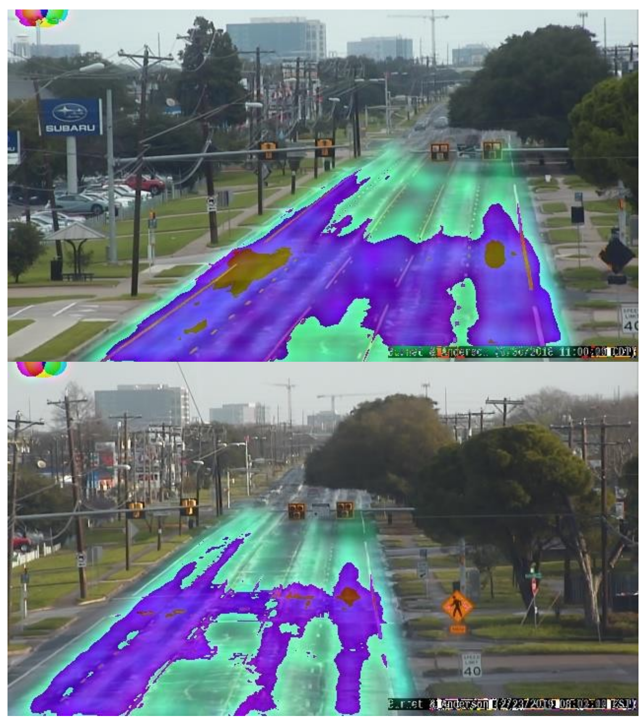

Fig. 7. Smooth heatmap view to show frequency of pedestrian crossing location shifts before (top, Sep. 29~ Oct 2 2018) and after (bottom, Feb 23 ~ Feb 25) PHB activation. Deep blue color indicates high likelihood of pedestrian

crossing while green color indicates less likely location to observe pedestrian crossing.

Reviewers classified 2,481 activities across five different activity types (walking, standing, crossing, cycling and other). The most frequently observed activities were walking and standing; street crossings accounted for approximately 20% of observed activities. A single individual may be counted across multiple activity types as their actions changed over the course of the video. There were 412 observations of one or more pedestrians crossing the street at a time. Using these curation results, we also calculated that pedestrians cross Burnet Road, which is approximately 60 feet wide, at a median rate of 3.5 feet/second. This is consistent with the literature, and may inform the computation of future site-specific metrics.

To assess the accuracy of the automated crossing event prediction, we compared prediction results for the number of crossing events per hour on 02-13-2019 at Anderson location with human reviewed results (Figure 8). The average prediction accuracy is about 67%. The comparison results show a wide margin of errors over different hours of the day, from less than 5% to more than 50%. A major factor contributing towards this large variability of prediction is the fact that traffic conditions change over different hours but our model are derived from aggregated results from all data.

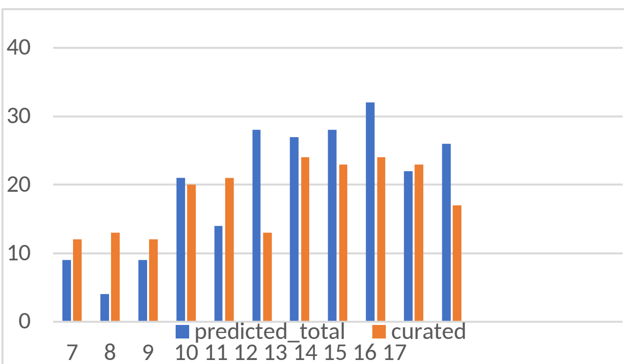

Fig. 8. Comparison of crossing event prediction with human reviewed results at Anderson location.

VI. Conclusion

In this paper, we present an approach that utilizes existing traffic monitoring cameras within an intelligent transportation system to understand pedestrian movement patterns and safety. The use case presented in this work illustrates the potential of implementing deep-learning methods to the study of pedestrian street-crossing patterns. We also propose several metrics of pedestrian activity to support the systematic analysis of before and after scenarios and the quantification of the benefits of pedestrian safety treatments. While preliminary, our results suggest that meaningful metrics may be derived automatically from data recorded through traffic monitoring cameras, which could enable agencies to conduct more thorough analyses on a larger number of locations.

Artificial intelligence technologies can greatly reduce the effort involved in analyzing video data, and frameworks such as the one presented here can facilitate research traditionally based on manual video data analysis and promote further work on video data applications and integration. A unique advantage of our framework is to convert video recordings into query-able information, which can accommodate multiple subsequent use cases without re-processing [3]. We have exemplified their potential to support useful analyses with minimal effort compared to manual processing. An additional benefit of this approach is that processed data can be combined with other datasets to conduct more complex analyses. For example, video data may be combined with loop detector data and signal timing data to understand pedestrian compliance with traffic signals. Traffic data from Bluetooth or Wavetronix sensors may support a more comprehensive assessment of pedestrian behavior by providing contextual information including prevalent vehicle speeds and traffic volumes.

Acknowledgment

This work is based on data provided by the City of Austin under the Vision Zero project, which also provided partial support for this research. We would like to thank Kenneth Perrine and Chris Jordan for their help in setting up video recording environment. The computation of all experiments was supported by the National Science Foundation, through award # 1726816, Stampede2 (OAC-1540931) and XSEDE (ACI-1953575) awards.

References

1. FHWA. Proven Safety Countermeasures: Pedestrian Hybrid Beacons. https://safety.fhwa.dot.gov/provencountermeasures/ped_hybrid_beacon/ .
2. Sayed, T., M. H. Zaki, J. Autey (2013). Automated safety diagnosis of vehicle–bicycle interactions using computer vision analysis, *Safety Science*, vol. 59, pp. 163–172.
3. Weijia Xu, Natalia Ruiz-Juri, Ruizhu Huang, Jennifer Duthie, and John Clary. 2018. Automated pedestrian safety analysis using data from traffic monitoring cameras. In Proceedings of the 1st ACM/EIGSCC Symposium on Smart Cities and Communities (SCC '18). ACM, New York, NY, USA, Article 3, 8 pages. DOI: https://doi.org/10.1145/3236461.3241972
4. Kastrinaki, V., M. Zervakis, K. Kalaitzakis (2003). A survey of video processing techniques for traffic applications, *Image and Vision Computing*, vol. 21, issue 4, pp. 359–381.
5. Kuciemba, S., K. Swindler (2016). Transportation Management Center Video Recording and Archiving Best General Practices. Federal Highway Administration report no. FHWA-HOP-16-033.
6. Zangenehpour, S., L. F. Miranda-Moreno, N. Saunier (2015). Automated classification based on video data at intersections with heavy pedestrian and bicycle traffic: methodology and application. *Transportation Research Part C: Emerging Technologies*, vol. 56, pp. 161–176.
7. Hu, W., X. Xiao, D. Xie, T. Tan, S. Maybank (2004). Traffic accident prediction using 3-D model-based vehicle tracking. *IEEE Transactions on Vehicular Technology*, vol. 53, issue 3, pp. 677–694.
8. St. Aubin, P., L. Miranda-Moreno, N. Saunier (2013). An automated surrogate safety analysis at protected highway ramps using cross-sectional and before-after video data. *Transportation Research Part C: Emerging Technologies*, vol. 36, pp. 284–295.
9. St. Aubin, P., N. Saunier, L. Miranda-Moreno (2015). Large-scale automated proactive road safety analysis using video data. *Transportation Research Part C: Emerging Technologies*, vol. 58, part B, pp. 363–379.
10. Morris B. T., M. M. Trivedi (2008). A survey of vision-based trajectory learning and analysis for surveillance. *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, issue 8, pp. 1114–1127.

11. Morris B. T., C. Tran, G. Scora, M. M. Trivedi, M. J. Barth (2012). Real-time video-based traffic measurement and visualization system for energy/emissions. *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, issue 4, pp. 1667–1678.
12. Huang, L., W. Xu, S. Liu, V. Pandey, N. Ruiz Juri (2017). Enabling versatile analysis of large scale traffic video data with deep learning and HiveQL. In Proceedings *2017 IEEE International Conference on Big Data (Big Data)*, Boston, MA.
13. Redmon, J, S. Divvala, R. Girshick, A. Farhadi (2016). You Only Look Once: Unified, Real-Time Object Detection. In Proceedings *IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV.
14. M. Hussein, T. Sayed, P. Reyad, and L. Kim, “Automated Pedestrian Safety Analysis at a Signalized Intersection in New York City Automated Data Extraction for Safety Diagnosis and Behavioral Study,” *Transportation Research Record*, 2015.
15. K. Ismail, T. Sayed, and N. Saunier, “Automated analysis of pedestrian-vehicle conflicts: a context for before-and-after studies,” *Transportation Research Board*, 2010.
16. F. M. Puscar, T. Sayed, A. Y. Bigazzi, and M. H. Zaki, “Multimodal Safety Assessment of an Urban Intersection by Video Aanlysis of Bicycle, Pedestrian, and Motor Vehicle Traffic Confilicts and Violations,” 2018.
17. S. Zangenehpour, L. F. Miranda-Moreno, and N. Saunier, “Automated classification based on video data at intersections with heavy pedestrian and bicycle traffic: Methodology and application,” *Transportation Research Part C: Emerging Technologies*, 2015.
18. T. Sayed, M. H. Zaki, and J. Autey, “Automated safety diagnosis of vehicle–bicycle interactions using computer vision analysis,” *Safety science*, vol. 59, pp. 163–172, 2013.
19. M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, “Spark : Cluster Computing with Working Sets,” in *HotCloud’10 Proceedings of the 2nd USENIX conference on Hot topics in cloud computing*, 2010, p. 10.

Fig. 9. Summary of Pedestrian detection per minute at Anderson Location during selected days in Sep. 2018. (bottom) Summary of Pedestrian detection per minute at Anderson Location during selected days in Feb. 2019. In both top and bottom panels, the blue points represent the number of pedestrians detected within one minute of video and the red line represents a 30-minute rolling average of the count.