Video Analytics for Vision Zero
Task 2017-07

FINAL REPORT

Prepared for the City of Austin
by
The Center for Transportation Research at The University of Texas at Austin
in collaboration with the
Texas Advanced Computing Center at The University of Texas at Austin

COA/CTR Interlocal Agreement
UTA16-000016, Task 2017-07

September 2018

THE UNIVERSITY OF TEXAS AT AUSTIN
CENTER FOR TRANSPORTATION RESEARCH
3925 W. Braker Lane • Suite 4.11080 • Austin, TX 78759 • (512) 232-3100 • ctr.utexas.edu

CONTENTS

Executive Summary

Transportation agencies often own extensive networks of monocular traffic cameras, which are typically used for traffic monitoring. However, the data captured by such cameras can also be of great value for transportation planning and operations applications, particularly when large data sets may be systematically analyzed. We implemented an approach to use data collected by existing monitoring cameras to automatically analyze data at locations where pedestrian safety may be a concern. Our methodology utilizes an artificial intelligence to identify pedestrians in traffic camera feeds. Results are stored and aggregated, and therefore may be queried for further analyses. The approach may leverage hardware such as GPUs and distributed computing clusters to further enable the analysis of large data volumes. Post-recognition analysis utilizes unsupervised learning methods to identify the spatial and temporal patterns of pedestrian positions, which are then correlated to specific scenarios such as usage of crosswalk, compliance with traffic signals, and pedestrian-vehicle interactions. Meaningful applications of this methodology include the identification of potential safety concerns, measuring the effectiveness of proposed safety strategies, and identifying the need for improvements. This report provides preliminary results based on data from City of Austin cameras and discusses outputs such as pedestrian volume estimation and crossing hot-zones identification in the context of Smart Cities, and identifies potential challenges and limitations.

Introduction

Walking is one of the most sustainable modes of transportation, and promoting walking can contribute to the development of healthy and livable community. Urban planning and transportation agencies often make substantive changes to a wide range of built environment features seeking to foster physical activity. The increased choice of non-motorized modes such as walking for transportation can also reduce traffic congestion, decrease energy consumption and improve air quality (Saunier et.al. 2011). However, pedestrians are the most vulnerable group among all non-motorized modes and endure the highest share of fatal road collisions. Pedestrian safety has received significant attention in transportation engineering and planning in recent decades. For this project, we designed and implemented a prototype pipeline to collect and store data from traffic cameras. Collected data was analyzed using a method for automated video content recognition and analysis which was developed in the previous project stage. This report describes the use and extension of our framework to the study of pedestrian movement patterns along and across roadways. The outcomes of this work are expected to support the planning and evaluation of future safety improvements.

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 (Kastrinaki et.al. 2003). However the use of camera video data in system performance/safety assessment or strategic planning is not widespread. Traffic management centers (TMCs) primarily use traffic video data from roadside cameras to identify incidents, prepare the response for emergency situations and manage traffic in special events (Kuciemba and Swindler, 2016). Video data is also used to manually conduct traffic studies such as collecting traffic counts by mode, turning movement counts for traffic signal timing applications, and conducting safety analysis by observing the behavior of traffic in weaving zones (Zangenebpour et al., 2015). In practice, analysis software is often implemented to support real-time traffic operations, commonly focusing on vehicle detection and tracking. Examples include safety analysis for intersections and corridors, identification of unusual events on corridors, generation of traffic counts and queue lengths, and for vehicular emission analysis by estimating traffic speeds (Hu et al., 2004; St-Aubin et al., 2013; St-Aubin et al., 2015; Morris and Trivedi, 2008; Morris et al., 2012).

The analysis of historical data from video camera feeds is less common in practice due to the significant storage and computing resources required to support it. Additionally, the effort involved in

manually extracting meaningful information from video data is prohibitive for most public agencies. Most users discard traffic-monitor data after specified time periods (typically ranging from one day to one year) depending on the recording purpose (Kuciemba and Swindler, 2016). An additional challenge in considering pedestrian data is that regular roadside cameras are installed to have wide and deep view of fields, while pedestrian activities occupy only a small portion of the view. At many locations pedestrians are only sporadically present, especially when compared to vehicle flows. Further, because pedestrians are smaller than cars they are more easily obstructed by other objects within the scene.

Pedestrian safety analysis involves identifying factors leading to unsafe conditions at a particular location, and has traditionally been conducted based on the judgment and experience of traffic safety professionals. The collection and analysis of video data at critical locations provides an opportunity to capture and analyze traffic conflicts based on a permanent, verifiable account of road user behavior, thus reducing the need to rely on ad-hoc decision-making (Sayed et al., 2013). However, if analyses are conducted by human observers, there is a limitation in the number of locations and analysis periods that may be considered. Automated approaches to effectively recognize, analyze and store pedestrian activities over time are needed. In this paper we implement a flexible framework for analyzing historic video from traffic cameras to the study of pedestrian movements and safety (Huang et al. 2017; Xu et al. 2018).

The proposed framework is more general than traditional traffic video analysis tools, which are typically designed to accomplish a single type of analysis. Further, the proposed approach separates the expensive computational steps of object recognition from the subsequent data-intensive analysis, allowing the utilization of different hardware and software resources at various stages for maximum efficiency. The prototype application analyzes video recordings over time and generates two types of visual summaries of pedestrian activities: a visualization of locations where pedestrians are present, and a display of their trajectories. These capabilities and potential applications are exemplified using camera data gathered from three locations in Austin.

Background

The following sections provide an overview of the challenges involved in detecting and identifying objects in video data frames, including special considerations when studying pedestrians movements. We also discuss how methodology such as the one implemented in this study may support content-based search and analysis of archived video data.

Object detection and recognition

By definition, video is a representation of moving pictures constructed by sequential frames of images; an image is defined as a collection of red, green, and blue (RGB) pixels used to illustrate visual information (Borkar and S. S. Katariya, 2017). The identification of objects in video data streams is typically conducted at the frame level (i.e., for each of the images that compose the video stream). Given that video streams consist of a large number of frames, content recognition and analysis is a time-consuming process. However, the differences between contiguous frames are often minor, and a considerable amount of redundant information is present in video data. To accelerate the process of identifying relevant detail, we have applied the “key frame” extraction approach, therefore utilizing a compact frame-set as the representation of information in the entire video (Zheng et al., 2015).

In recent years, neural-network-based image-recognition methods have evolved quickly and detection-accuracy has significantly improved (Krizhevsky et al., 2012; Simonyan and Zisserman, 2015). Approaches based on neural networks are now capable of detecting multiple types of objects at different scales (Szegedy, 2013;

Ciresan et al, 2012; Impiombato *et al.*, 2016; Ren et al., 2017). Very recently, methods based on deep learning have been proposed to support video analytics for smart cities (Wang and Sng 2015).

In our approach, we propose to use the "You Only Look Once" (YOLO) library for object detection and recognition. YOLO is unique in its ability to detect objects quickly enough for real-time applications (Impiombato et al., 2016). YOLO's object detection and recognition has been integrated into one neural-network approach, resulting in an algorithm that can effectively take into consideration the global information of a frame and is less sensitive to the effects that shadows have on recognition. To our knowledge, this is the first time YOLO has been applied to the analysis of traffic data.

Pedestrian Tracking

The detection and tracking of main road users (e.g., pedestrians, cyclists, and vehicles) remains a hot topic in the field of computer vision, with a research focus often on developing an automated process to identifying object trajectories and therefore avoiding time-consuming manual processing. While significant work exists on the topic of vehicle tracking, fewer studies look into pedestrian tracking, which is significantly complex at urban signalized intersections because of pedestrians' non-rigidity, more varied appearance, and less organized movements. Pedestrians may change their direction of movement frequently, while vehicles must follow lanes and have limited turning options (Hussein et al., 2015). Additionally, pedestrians often move in groups, making detection and tracking of individual movements even more complicated (Zangenehpour et al., 2015).

Content Based Analysis and Search

Maintaining large quantities of image data requires significant storage resources. Further, large working bandwidths are often needed to transfer the structural details of image characteristics (Hanjalic and Xu, 2005). Content Based Image Retrieval (CBIR) requires complex processing algorithms to enable the transfer of actual physical information, including the large number of pixels, redundant bits, contrast, hue, and other characteristics from the corresponding file (Borkar and Katariya, 2017; Chun et al., 2008; Yuan and Zhang, 2010; Kumar et al., 2015). The identifications of the correct features to be retrieved is feasible because similar images have semantically similar objects (Guo and Prasetyo, 2015). The efficiency of storage and transmission can be enhanced when only the useful contextual contents of an image are retrieved, rather than the full image (Kumar et al., 2015).

Video retrieval follows the same principle of image retrieval except for the similarity measurements of image frames through color and bit mappings of stored videos (Sayed et al., 2013). However, moving from images to video adds several orders of complexity to the retrieval problem by adding a temporal dimension to indexing, analysis, and browsing (Saravanan et al., 2017). Video retrieval algorithms query multiple types of video content by comparing similarities in colors, shapes, pixel contents of frames, and bitmap properties (Borkar and Katariya, 2017; Sze et al., 2005). Content Based Video Retrieval (CBVR) deals with the problem of locating a specific video snippet within a large number of video files by searching content (Mirmehdi et al., 2000; Qiu and Lam, 2003). Using an approach similar to that proposed in this paper, Abdullah et al. utilize cloud computing to build a system to manage and analyze traffic monitoring data. The system supports vehicle detection using a cascade classifier (Abdullah et al., 2014).

Several key issues in CBVR must still be addressed, including bridging the "semantic gap" between low-level features (such as color, texture, shape, motion) and high-level semantic meanings of content (such as people, vehicle speed, indoor and outdoor) (Saravanan et al., 2017). Promising intelligent systems are required to translate low-level feature representation of video into a model of high-level content

representation. Content summarization is another feature urgently needed to enable efficient access to the rich content of video information for spatial and temporal analysis of visual media. Techniques such as key-frame extraction may be used for compact representation and fast browsing of video content. A third aspect of CBVR that requires further research is the dynamic updating of video databases, and corresponding dynamic matching of queries and databases (Ma et al., 2009).

We addressed the high computational requirements of CBVR by taking advantage of distributed computing using Hive and Spark, using the latter and HiveQL as a standard query language for ad-hoc analysis needs. Hive was first introduced in 2009 as a data warehouse solution utilizing a Hadoop cluster and supports multiple types of data sources and file formats, including structured text format, such as CSV and JSON, and serialization formats, such as Avro, Parquet, and ORC (Thusoo et al., 2009). The supported data sources can be ingested in Hive as one or more tables. With files stored in Hadoop cluster, Hive can take advantages of aggregated IO performance backed up by the Hadoop Distributed File System (hdfs) (Shvachko et al., 2010). Hive provides a SQL like query language, known as HiveQL, for users to search tables in the system. HiveQL requests are transformed into a set of distributed computation tasks through a built-in query engine that supports both the MapReduce programming model and Spark programming model(Dean and Ghemawat, 2010; Zaharia et al., 2012). While Spark was initially proposed as an in-memory computing cluster framework and a distributed programming abstraction model, it has evolved to support interactive analysis with SQL-like queries with the recent development of DataFrame and SparkSQL features (Armbrust et al., 2015). SparkSQL follows the same query language specification as Hive, which significantly increases the interoperability between Spark and Hive. Although Hive has been successfully used for business analysis, and in many domains for scientific discoveries, the application of Hive in transportation is still limited (Xu et al., 2016).

Data Workflow and Analysis Framework

The following framework describe the approach used in this project to access, record and analyze video data streams from traffic monitoring cameras from the City of Austin. We also present an overview of the object detection and tracking methods used in this work, which are extension of the techniques presented in further detail in the project report for FY2017. Further technical details are available in published research papers by the authors.

Data Pipeline

The analysis framework implemented in this project requires recording video data streams generated by traffic monitoring cameras, processing the resulting files to generate a database of tracked objects, and querying such database to extract the desired information (Huang et al., 2017; Xu et al., 2018). Splitting the object recognition/tracking step and subsequent analyses allows us to use the most efficient tools for each task: deep learning for object detection and big data processing for the analysis. The resulting framework can efficiently and automatically process large-scale traffic video data and meet evolving analytic needs over time.

To implement the framework, we have set up a multi-systems cross-domain video aggregation and analysis pipeline (Figure 1). Raw videos originate from IP cameras in the City of Austin (CoA) private network. To overcome the network's limited accessibility, 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), where the recorded video can be processed by a high performance computing cluster. 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 recoding and processing tasks.

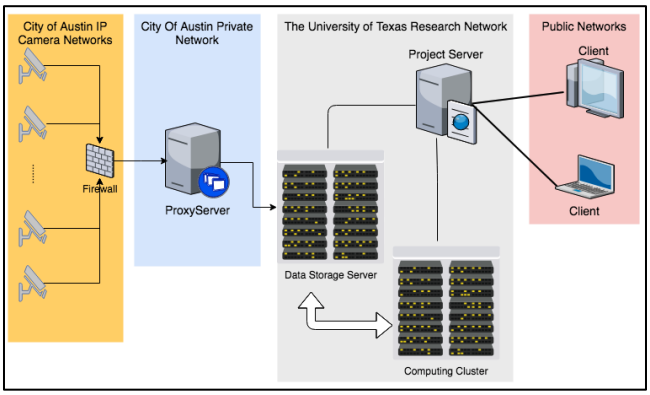

Figure 1. Camera access and processing pipeline overview.

Figure 2 illustrates the main characteristics of the video processing framework. Object Recognition (Labeling) is the first of three main components in the algorithm. It identifies and labels all candidate objects from original input data/image files using a deep-learning based algorithm. Object Tracking (Tracking), which compares recently recognized objects with previously recognized objects, is the second, in which we use background subtraction techniques to differentiate moving objects from stills or background objects and filter out redundant objects. A complete list of all target objects with corresponding detailed information is stored in a structured data file. The structured output data obtained from early components can be registered as Dataframe for further analysis work using Spark framework. Finally, the Object Analysis (Analyzing) module provides efficient Dataframe querying capabilities through Spark/HIVEQL.

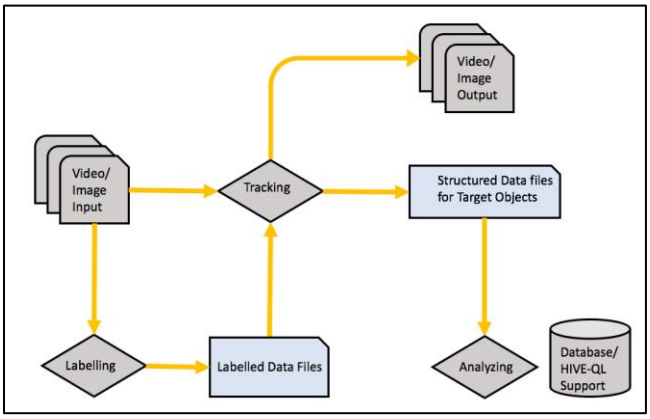

Figure 2. Overview of video processing steps.

Video Processing and Analysis

The video content recognition process is based on Darknet, an open source open source neural network framework (Impiombato et al, 2016). The core algorithm utilizes YOLOv2, a convolution-neural-network-based object-detection system, to analyze each frame of an input video. For each frame, the algorithm outputs a list of objects recognized through a pre-trained model. Each object is defined by its location in the frame, class label, and confidence of recognition. In the CoA application, we have limited recognition to seven class labels: person, car, bus, truck, bicycle, motorcycle, and traffic light.

To improve the performance and maximize utilization of multi-node computing clusters, we have also adapted the YOLO implementation for parallel execution, which enables parallel recognition of multiple frames using pthread within individual compute nodes and inter-node communication using MPI. Specifically, one thread is used to pre-fetch *n* frames, while *n* extra worker threads are assigned to labeling. Each worker thread takes care of one individual frame. Since there is no dependency, ideal linear scaling can be achieved for videos with longer duration. For video recordings from different times/locations, multiple video files can be processed independently and concurrently across multiple nodes. To maximize resource utilization, longer videos can be split into portions and distributed to multiple nodes for parallel processing. A non-maximum suppression (NMS) algorithm with the locally maximal confidence measure is used to remove unnecessary or duplicated objects. In addition to content recognition, we also implemented a method to extract the background (i.e., non-moving features) of the video as part of the process output. Interested readers can refer Huang et al., 2017 and Xu et al., 2018 for more details.

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: object index within this trajectory} as the set of objects stored by a list of trajectories

1: Initialize *T* with each object found in the starting frame
2: for each n_{ij} in *N*
3: for each t_k in *T*
4: dists <= distance(n_{ij}.location, pred(t_k, *i*))
5: if min (dists) < *threshold*

Figure 3. Pseudo code for tracking pedestrians.

Pedestrian Identification and Tracking

The recognition algorithm can identify new individuals in a frame when they are in reasonable proximity to the camera, and provided that the view is unobstructed. However, tracking individuals across frames remains challenging. The vehicle tracking algorithm proposed relies on the intersection over union value between identified objects across frames, which is not effective when tracking pedestrians (Xu et al., 2018). For the CoA application, we used an approach based on proximity of predicted positions of objects from different frames.

As shown in Figure 3, the algorithm is initialized with the set of recognized "person" objects in each frame. For each recognized object in the first frame we initialize a trajectory. A recognized object in the subsequent frame is associated to the closest trajectory. Once a trajectory has at least two distinct positions, direction and velocity of the trajectory can be estimated. In subsequent frames, we compute the distance between all identified objects and the predicted positions of existing trajectories at that frame. If an object's minimum distance to a trajectory is greater than that in a pre-defined threshold, the algorithm generates a new trajectory. Otherwise, the object position is added to the trajectory whose predicted position is the closet.

A Case Study of Pedestrian Safety

In the context of this study more than 1,000 video files were recorded for the purpose of algorithmic development and testing. This section describes the implementation of the proposed methodology to the

analysis of pedestrian activity at four locations, and exemplifies some of the visuals that may be generated based on the processing and analysis outputs.

Locations and video recordings

The use case analyzed for the CoA application consists of locations where pedestrians frequently cross a street in order to understand the impact of measures designed to promote street crossings at designated safe areas, such as crosswalks. We selected four camera locations (Table 1) and used the video aggregation pipeline to record video segments throughout the day. Automated recording capabilities were developed to support the collection of longer video data streams, which may be used to conduct meaningful before/after analyses. As an example, the intersection of Anderson Lane and Burnet Road will be equipped with a Pedestrian Hybrid Beacon (PHB), and the methods discussed in this report will be used to compare pedestrian behavior before/after the deployment of such devices.

Table 1. Data collection summary

	Lamar & 24th	Lamar & 38th	Lamar & 45th	Anderson & Burnet
Time range	2018.04.28 00:44~23:59	2018.04.20 06:00~15:30	2018.04.20 06:00~15:00	Various dates in August 2018
Number of files	24	10	16	~1,000 / ongoing
Average size per video file	~315MB	~592MB	~530MB	~ 300MB
Total size	~7.5GB	~6GB	~8.5GB	~300GB
Average durations	~15 mins	~30 mins	~29.5 mins	15 mins

Experimental Design and Results

After labeling all video content, all “person” objects were selected and used for tracking. A drawback of this preliminary approach is that an object is considered a person as long as it shares visual similarities to human features. Therefore, identified objects include pedestrians, cyclists and motorists. This limitation may be addressed through further algorithmic refinement in future project stages.

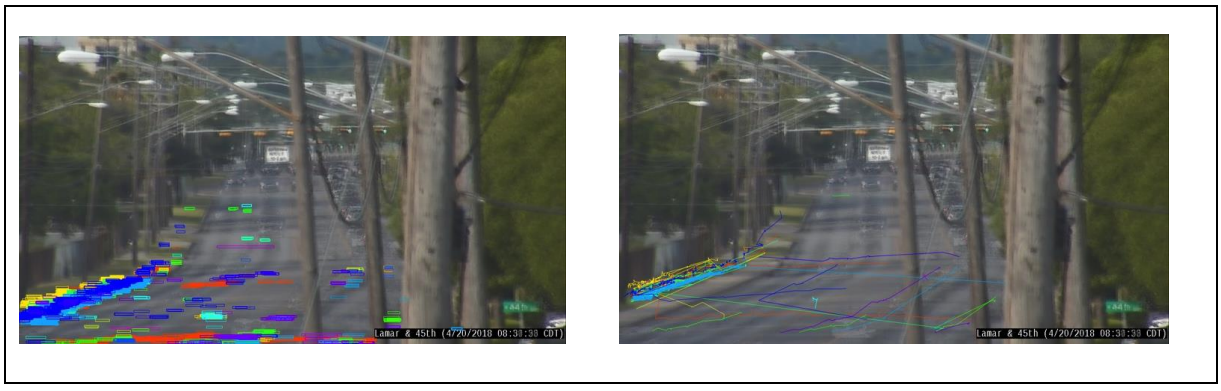

Figure 4. Visual summaries from recordings at Lamar Boulevard and 45th Street, including location summary (left) and tracking summary (right).

Figures 4–6 exemplify the two types of visual summaries generated based on labeling and tracking results. The background image in such figures is extracted from the data stream during the processing stage. The first visual summary in Figure 4 is a location summary showing the positions of "person" objects across videos files. The second visual summary is a tracks, summary displaying identified trajectories. Results from different video files are rendered using different colors. Figure 6 shows the two visual summaries for recordings at Lamar Boulevard and 45th Street. The results from Lamar Boulevard and 38th Street and Lamar Boulevard and 24th Street are presented in Figures 5 and 6, respectively.

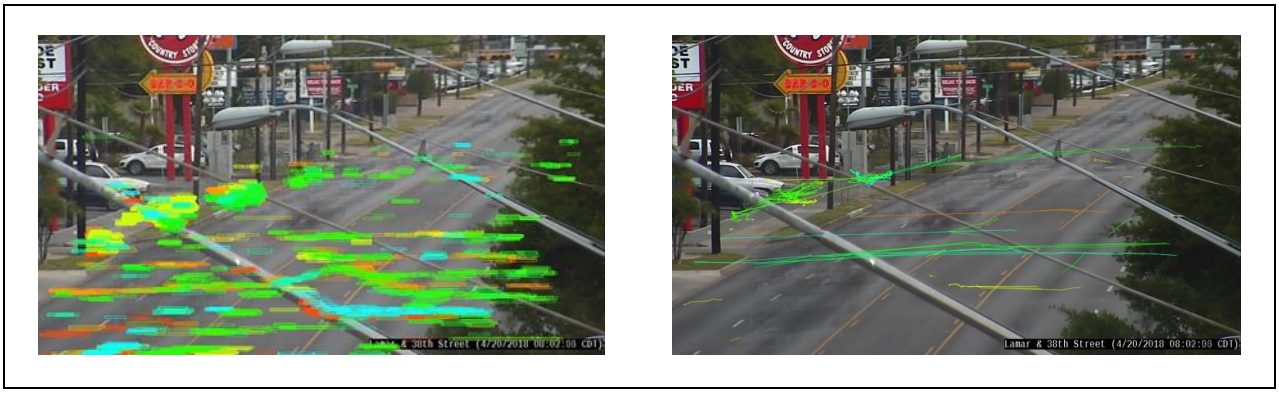

Figure 5. Visual summaries from recordings at Lamar Boulevard and 38th Street, including location summary (left) and tracking summary (right).

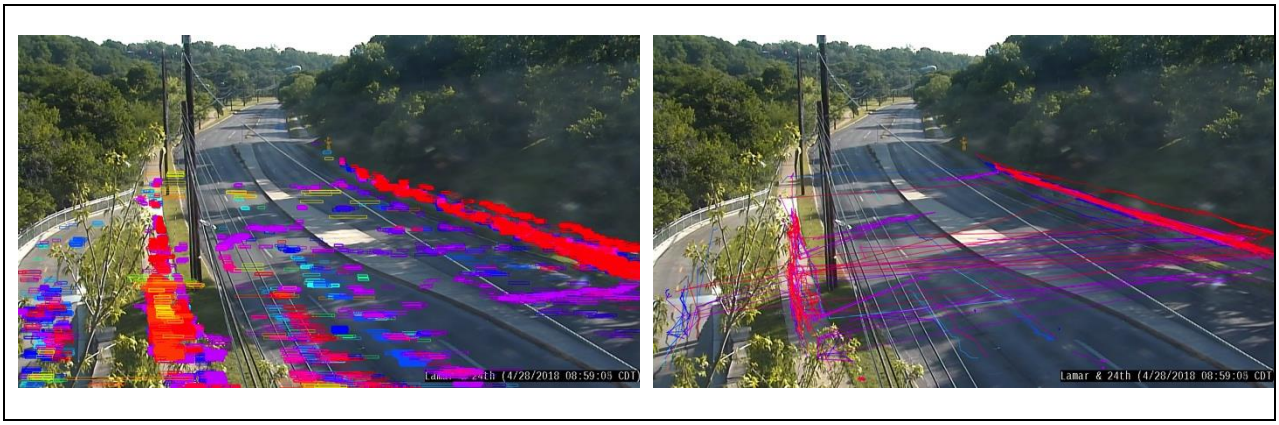

Figure 6. Visual summaries from recordings at Lamar Boulevard and 24th Street, including location summary (left) and tracking summary (right).

Location summary views are generally more cluttered, with seemingly many pedestrians identified in the middle of road. However, our approach is likely to be overestimating the presence of pedestrians on the street because "person" objects are not limited to pedestrians; they also include cyclists, motorists, and drivers with open roofs/windows. Misidentifications are most likely responsible for the tracks seen to follow the roadway direction.

The location summary view complements the track view by identifying spots frequented by pedestrians, or where pedestrians tend to stand for longer time intervals. For example, in Figure 5, clusters at the top left of the frame correspond to a bus stop. The track view in Figure 5 also suggests locations where pedestrians cross the street on their way from/to the stop, which is not always on the crosswalk.

At the Lamar Boulevard and 24th Street location there is a park with jogging trails on the left side of the road and an apartment complex on the right (Figure 6). Both summary views show significant pedestrian activity along the roadside and crossing the road.

In addition to visual summary, we can also create histograms to show summary statistics of activities over a period of time for a given location, using the activity index (AI) formula, as follows:

$$\textbf{\textit{Activity Index}}\ (\textbf{\textit{AI}}) = \frac{\textbf{\textit{Number of Person Idenfication}}}{\textbf{\textit{Number of Frames of videos}}}$$

The AI can be used as a singular numerical indicator of pedestrian usage of the road for a given video. A higher value indicates more personal activities presented in the video for unit time. The measure can be used for comparison purposes across different locations and times. Figure 7 shows a histogram from 07:00 to 22:00 at the intersection of Lamar and 24th.

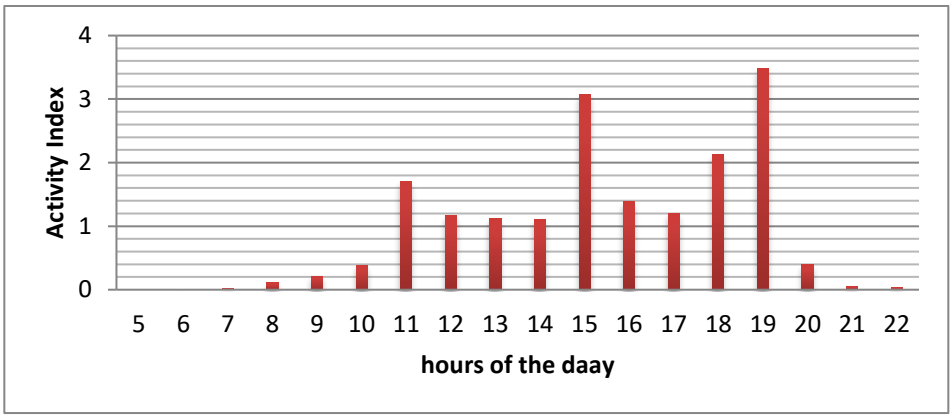

Figure 7. Activity index distribution at Lamar and 24th on April 28, 2018.

The detection and tracking of pedestrians can be exported as a delimited file for further analysis. Figure 8 shows an example of an export of pedestrian tracks that include road crossing. Each row represents a pedestrian track and corresponding video recording name, size (in number of frames where the track is recognized), start and end frame, and start and end location on the video. Crossing events are identified in the last columns.

file	size	end_frame	end_xmin	end_ymin	end_xmax	endy_max	from_frame	from_xmin	from_ymin	from_xmax	from_ymax	isCrosing
/work/03076/rhuang/maverick/CTR/A	222	26732	962	386	977	431	26196	946	370	960	419	FALSE
/work/03076/rhuang/maverick/CTR/A	74	25128	315	391	331	437	24536	158	472	179	526	FALSE
/work/03076/rhuang/maverick/CTR/A	52	24553	168	689	222	719	24305	30	542	55	608	FALSE
/work/03076/rhuang/maverick/CTR/A	209	23056	0	549	28	645	22789	20	524	51	606	FALSE
/work/03076/rhuang/maverick/CTR/A	292	22832	26	521	62	606	21967	190	458	209	523	FALSE
/work/03076/rhuang/maverick/CTR/A	443	11802	995	386	1014	463	10250	60	468	84	517	TRUE
/work/03076/rhuang/maverick/CTR/A	691	5594	948	376	964	429	3830	980	456	1007	499	FALSE
/work/03076/rhuang/maverick/CTR/A	339	3808	980	458	1005	501	2935	1011	482	1035	538	FALSE
/work/03076/rhuang/maverick/CTR/A	116	2198	1028	519	1055	558	1769	1040	543	1073	617	FALSE
/work/03076/rhuang/maverick/CTR/A	70	1670	1047	540	1082	639	1500	1052	553	1085	631	FALSE
/work/03076/rhuang/maverick/CTR/A	109	1760	1038	544	1071	619	1491	1051	556	1080	608	FALSE
/work/03076/rhuang/maverick/CTR/A	139	1346	1059	574	1099	667	1125	1058	592	1102	673	FALSE
/work/03076/rhuang/maverick/CTR/A	694	1618	1052	546	1083	645	628	1114	620	1154	688	FALSE
/work/03076/rhuang/maverick/CTR/A	315	818	1099	610	1139	690	153	993	405	1013	456	FALSE

Figure 8. Example tracking result output file.

Limitations

The work conducted during this project was critical to refine the methods proposed in the previous FY and provided a better understanding of the potential of these techniques. Through this work we also gained better insights on the current limitations of both the data gathering and framework and analysis techniques.

The video recording pipeline is sensitive to the stability of all involved networks, and to that of computing resources. In some cases, the instability causes disruption of video recordings. The results of detection and tracking algorithms are also sensitive to various parameters, such as the detection cutoff threshold (based on detection confidence) that is used to determine which objects are included in the analysis and can directly affect the number of labeled objects. Other threshold values are used in the tracking algorithm to measure the distance among objects, and may affect the number of predicted tracks. We also found that people walking in groups are difficult to track consistently because their distance to the camera is far greater than their distance to other objects. As a result, the current approach is not appropriate to estimate pedestrian volumes.

However, despite some limitations, the current framework can be used to better understand how pedestrians use the road and how the use pattern evolves over time or across locations. Results can also assist urban planners and traffic engineers in identifying location-based solutions to enhance pedestrian safety. Researchers can also generate a visual summary based on analysis needs for different locations or aggregated time periods. Figure 9 shows an example at the Lamar Boulevard and 24th Street location, divided into four different time periods. While pedestrian volume counts are not collectable, the qualitative change in pedestrian patterns over time is clear, and further quantification of the observed changes may be feasible.

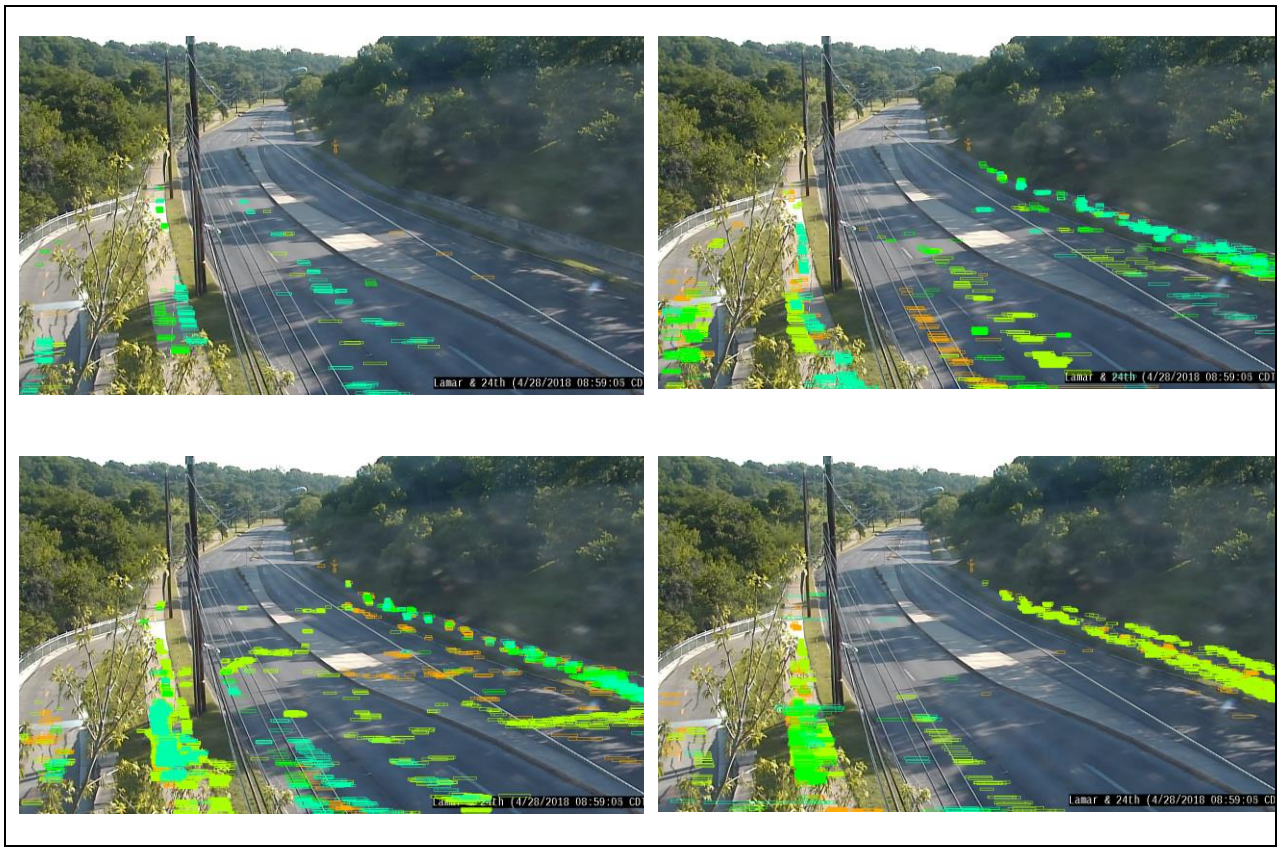

Figure 9. Aggregation over different times of day, from top left to bottom right: 06:00–10:00, 10:00–13:00, 13:00–17:00, 17:00–21:00.

Figure 10 shows track detection (right) and hotspot summaries (left) at Anderson Lane and Burnet Road. In this case, only one pedestrian crossing event is indicated in the crossing tracking summary (shown on the right). However, the hotspot view clearly indicates two possible crossing events (shown on the left). Inspection of the original video shows that the second track is a relatively short track from a pedestrian running across the road quickly. Due to its short duration, the track was missed in the current model.

Figure 10. Hotspot view of pedestrian detection location (left); (right) inferred pedestrian crossing event based on tracking algorithm. The missed crossing event in the pedestrian crossing summary (right) is due to significant speed differences (pedestrian running in missing track).

Conclusions and future opportunities

Artificial intelligence technologies can greatly reduce the effort involved in analyzing video data. Applications such as the one developed for CoA can accelerate research and analysis that has been traditionally based on manual video data analysis and can promote further work on video data application and integration. A unique advantage of the framework is to convert video recordings into query-able information that can accommodate multiple subsequent use cases without reprocessing. While the framework and specific applications are still in development, we have explored their potentials to support useful analyses with minimal effort compared to manual processing.

The approach provides a space saving alternative for raw video data storage, as the output of recognized objects can be much smaller than the raw video files. The storage requirement is significantly reduced when the raw video is no longer needed. The data also becomes anonymized because identifiable information is not stored with recognized objects. Therefore, preserving useful key traffic information over a large region for long duration of time becomes a more practical task.

The output information can also be combined with other datasets to conduct more complex analyses. For example, video data may be combined with loop detector data and signal timing plan 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, such as vehicle speeds and traffic volumes.

Further effort includes algorithmic refinement, and the extension of the web access interface We plan to enhance our object recognition system by introducing more object characteristics, and optimize our method to handle videos taken under severe weather conditions (e.g., night, rain, fog) with alternative solutions.

The use cases presented in this work illustrate both the benefits and limitations of the proposed methodology. The video aggregation pipeline has the potential to support long-term road-usage monitoring. The flexibility of the data selection and filtering capabilities is expected to enable additional practical applications. In addition to the visual summaries described in this study, quantitative outputs can be generated to facilitate the comparison of conditions across different locations or time ranges and to evaluate the impact of infrastructure changes and construction scenarios, among others.

Acknowledgments

This work is based on data provided by the City of Austin under the Data Rodeo project, which also provided partial support for this research. The authors are grateful for this support. 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 Stampede2 (OAC-1540931), and XSEDE (ACI-1953575) awards.

References

Abdullah, T., A. Anjum, M. F. Tariq, Y. Baltaci, and N. Antonopoulos, "Traffic monitoring using video analytics in clouds," in *Proceedings of the 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing*, 2014, pp. 39–48.

Armbrust M. *et al.*, "Spark sql: Relational data processing in spark," in *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*, 2015, pp. 1383–1394.

Borkar S.V., and S. S. Katariya, "Content based video retrieval scheme using halftone block truncation coding," in *Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017 2nd IEEE International Conference on*, 2017, pp. 1089–1094.

Chun, Y. D., N. C. Kim, and I. H. Jang, "Content-Based Image Retrieval Using Multiresolution Color and Texture Features," *Multimedia, IEEE Transactions on*, 2008.

Cireşan, D., U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," *International Conference of Pattern Recognition*, 2012.

Dean, J. and S. Ghemawat, "MapReduce: a flexible data processing tool," *Communications of the ACM*, vol. 53, no. 1, pp. 72–77, 2010.

Guo, J. M. and H. Prasetyo, "Content-based image retrieval using features extracted from halftoning-based block truncation coding," *IEEE Transactions on image processing*, vol. 24, no. 3, pp. 1010–1024, 2015.

Hanjalic A., and L. Q. Xu, "Affective video content representation and modeling," *IEEE Transactions on Multimedia*, 2005.

Hu, W., X. Xiao, D. Xie, T. Tan, and S. Maybank, "Traffic accident prediction using 3-D model-based vehicle tracking," *IEEE Transactions on Vehicular Technology*, 2004.

Huang, L., W. Xu, S. Liu, V. Pandey, and N. R. Juri, "Enabling versatile analysis of large scale traffic video data with deep learning and HiveQL," in *Big Data (Big Data), 2017 IEEE International Conference on*, 2017, pp. 1153–1162.

Hussein, M., 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.

Impiombato, D. *et al.*, "You Only Look Once: Unified, Real-Time Object Detection," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

Kastrinaki, V., M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image and Vision Computing*, 2003.

Krizhevsky, A., I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *ImageNet Classification with Deep Convolutional Neural Networks*, 2012.

Kuciemba, S. and K. Swindler, "Transportation Management Center Video Recording and Archiving Best General Practices," 2016.

Kumar, A., F. Nette, K. Klein, M. Fulham, and J. Kim, "A Visual Analytics Approach Using the Exploration of Multidimensional Feature Spaces for Content-Based Medical Image Retrieval," *IEEE Journal of Biomedical and Health Informatics*, 2015.

Ma, X., D. Schonfeld, and A. Khokhar, "Dynamic updating and downdating matrix SVD and tensor HOSVD for adaptive indexing and retrieval of motion trajectories," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*, 2009, pp. 1129–1132.

Mirmehdi, M. "Segmentation of color textures," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000.

Morris, B. T. and M. M. Trivedi, "A survey of vision-based trajectory learning and analysis for surveillance," *IEEE Transactions on Circuits and Systems for Video Technology*. 2008.

Qiu, G. and K. M. Lam, "Frequency layered color indexing for content-based image retrieval," *IEEE Transactions on Image Processing*, 2003.

Ren, S., K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

Saravanan, M. S. G. , M. T. Sivaprakasam, and D. Somasundaram, "A Review on Content Based Video Retrieval, Classification and Summarization," *Asian Journal of Applied Science and Technology (AJAST)*, vol. 1, no. 9, pp. 40–45, 2017.

Saunier, N., A. El Husseini, K. Ismail, C. Morency, Jean-Michel Auberlet, and T. Sayed, "Pedestrian Stride Frequency and Length Estimation in Outdoor Urban Environments using Video Sensors," *Transportation Research Record*, 2011.

Sayed, T., 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.

Shvachko, K. , H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*, 2010, pp. 1–10.

Simonyan, K. and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *International Conference on Learning Representations (ICRL)*, 2015.

St-Aubin, P., L. Miranda-Moreno, and N. Saunier, "An automated surrogate safety analysis at protected highway ramps using cross-sectional and before-after video data," *Transportation Research Part C: Emerging Technologies*, 2013.

Sze, K. W., K. M. Lam, and G. Qiu, "A new key frame representation for video segment retrieval," *IEEE Transactions on Circuits and Systems for Video Technology*, 2005.

Szegedy, C. "Deep Neural Networks for Object Detection," *Nips 2013*, 2013.

Thusoo, A. *et al.*, "Hive: a warehousing solution over a map-reduce framework," *Proceedings of the VLDB Endowment*, vol. 2, no. 2, pp. 1626–1629, 2009.

Wang, L. and D. Sng, "Deep learning algorithms with applications to video analytics for a smart city: A survey," *arXiv preprint arXiv:1512.03131*, 2015.

Xu, W. *et al.*, "Supporting large scale connected vehicle data analysis using HIVE," in *Big Data (Big Data), 2016 IEEE International Conference on*, 2016, pp. 2296–2304.

Xu, Weijia, Ruiz-Juri, Natalia, Huang, Ruizhu, Duthie, Jennifer and Clary, John. (2018). Automated pedestrian safety analysis using data from traffic monitoring cameras. In proceedings of 1st ACM/EIGSCC Symposium On Smart Connected Communities, June 2018, Portland, Oregon

Yuan, H. and X. P. Zhang, "Statistical modeling in the wavelet domain for compact feature extraction and similarity measure of images," *IEEE Transactions on Circuits and Systems for Video Technology*, 2010.

Zaharia M., *et al.*, "Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing," in *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, 2012, 2.

Zangenehpour, S., 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.

Zheng, R., C. Yao, H. Jin, L. Zhu, Q. Zhang, and W. Deng, "Parallel key frame extraction for surveillance video service in a smart city," *PLoS ONE*, 2015.