DLSU RESEARCH CONGRESS 2020
"Building Resilient, Innovative, and Sustainable Societies"
June 17-19, 2020

Development of Annotation Tools for building Labeled Datasets for a Vision-based Traffic Monitoring System

Jonathan Cempron and Joel Ilao
Center for Automation Research, College of Computer Studies
De La Salle University, Manila, Philippines
**Corresponding Author: joel.ilao@dlsu.edu.ph*

Abstract: High-quality annotated datasets are valuable resources in developing learning-based systems, and in objectively measuring their corresponding performance. However, data collection for dataset creation and annotation is tedious, especially when involving large datasets. In this paper, we describe the tools that have been used to expedite the creation of vehicle counts from vehicular traffic surveillance videos, image datasets of different car categories, and trajectory information of passing cars intended for training a vehicle detector of a vision-based traffic monitoring system. Software tools were developed specifically for the creation of the aforementioned datasets.

Key Words: Datasets; Annotation; Intelligent Transport System; Computer Vision

1. INTRODUCTION

In the field of computer vision, object classifiers categorize a picture containing a single object according to pre-set categories while object detectors both localize and classify visually identifiable objects in images. These systems are typically implemented using models trained on examples of pre-labeled images. Many implementations of object classifiers and detectors are based on supervised machine learning approaches that use traditional techniques, i.e. SVM, decision trees, regression, clustering, and more recently using neural networks and deep learning technologies (Liu et al., 2020).

The performance of object detectors is dependent on the quantity and quality of training datasets. As shown in Table 1, datasets, particularly those used for benchmarking, are typically large and contain thousands to millions of images, depending on the application and object categories considered. Furthermore, the process of building datasets, from raw image collection to annotation, is also laborious and time-consuming, typically requiring several man-hours because of the large video collections with several ground-truth labels covering many object categories in cluttered scenes. In ensuring the annotation quality, image annotation behaviors and patterns also need to be considered in designing the annotation software's user interface and in streamlining the corresponding annotation process (Chen, Wang, Xie & Lu, 2018). These issues underscore the need for annotation tools that are convenient-to-use and intuitive and make the annotation tasks quicker and more efficient through the automatic generation of initial annotations from detectors pre-trained from seed data.

In this paper, we describe the annotation protocol and the annotation tools specially designed to produce labeled image frames extracted from vehicular traffic videos, intended for training a vehicle detector of a traffic monitoring system, from the vantage point of ordinary road-side traffic surveillance cameras (Cempron and Ilao, 2019). Section 2 describes the dataset and the features of the annotation tool relative to corresponding annotation tasks. Section 3 describes the adopted annotation protocol and samples of the resulting annotated dataset. Section 4 contains the Conclusions and Future Work. Due to space limitations, however, details of the implementation of the annotation tools will not be presented.

Table 1. Select Image Datasets used for Object Recognition

Name	Remarks	Size
LabelMe (Russel, Torralba, Murphy & Freeman, 2008)	web-based tool for easy annotation and sharing	187,240 images, 62,197 annotated images, and 658,992 labeled objects
ImageNet (Deng et al., 2009)	Organized according to WordNet hierarchy, with each node depicting hundreds and thousands of images	~14M images
MS COCO (Ty et al., 2014)	contains images of common objects in everyday scenes	330K images of which >200K are labeled, 80 object categories, 1.5M object instances
Google's Open Images (Krasin et al., 2017)	rich annotations, complex scenes with several objects per image	~9M images, >6000 object categories
PASCAL VOC (Everingham et al., 2015)	image datasets were released as part of an annual competition and workshop held from 2005 to 2012	27,450 detection objects, 11,530 images, 20 different classes[1]

[1] The dataset sizes vary depending on the year of the PASCAL VOC challenge. The reported size is from the last (2012) challenge.

2. METHODOLOGY

There are three annotation tasks performed: Count annotation, Car image classification annotation, and Trajectory annotation. Each annotation task is discussed in the following sub-sections.

2.1 Raw Data

The raw data used in our project are video recordings of select roads or highways. A typical recording lasts from a few days to a few months. The total file sizes range from a few gigabytes to a few terabytes respectively. One day of recording usually yields 4 video files with 6 hours each. Each video file is then chopped into 5-minute chunks. The raw data given to the annotators are then partitioned into 5-minute video chunks.

2.2 Count Annotation

The count annotations by human annotators serve as a reference for comparison in determining the accuracy of the developed vehicle counter. Two annotators are given the same set of videos and they are tasked to count the vehicles on the videos given to them. The inter-rater correlation of the annotators will be used as a target correlation for the analysis of the accuracy of our automated vehicle counter.

The tools we recommend to be used by the annotators for the counting tasks are multimedia player VLC[2] and Microsoft Excel[3]. The actual counting task was performed manually. An alternative is to use readily available annotation tools like LabelMe[4] but such tools have extra features that cater to generic CV applications which could hamper the annotation process.

Table 2. Sample Annotated Count

Vidnum	Truck	SUV	Jeep	Sedan	Van	Motor	Non-vehicle	Bus	TOTAL
500	6	35	18	48	6	22	55	0	190
501	5	31	24	41	11	11	57	1	181
502	5	34	27	38	8	9	48	1	170
503	5	28	17	32	6	9	43	0	140
504	4	42	16	34	5	8	36	1	146
505	4	38	15	20	3	8	18	1	107
506	4	28	7	19	5	11	9	1	84
507	3	21	8	20	3	6	30	0	91
508	2	25	13	22	5	7	18	3	95
509	4	19	15	17	5	14	30	4	108
510	5	14	7	24	2	11	31	1	95
511	5	24	16	26	6	11	41	3	132
512	4	25	9	34	9	10	30	5	126
513	5	24	17	21	6	9	41	1	124
514	1	13	5	24	4	13	34	1	95
515	5	16	12	21	3	8	43	3	111
516	3	22	8	19	7	9	39	4	111
517	6	25	12	30	3	21	21	2	120
518	3	20	9	15	7	10	19	3	86

The output of the count annotation is a CSV file containing entries with corresponding identifiers to a video, filename or video number, and the corresponding vehicle counts per classification. A sample annotation count is shown in Table 2.

2.3 Car Image Classification Annotation

The car image classification annotation task aims to build a dataset that will be used in training the vehicle classifier module of our traffic monitoring system. Annotators are provided with a set of videos and an annotation tool. The output of the car image classification annotation task is a set of image files, wherein each image contains a vehicle cropped from an input video. The classification of the vehicle on the image is embedded in its filename.

Figure 1. Region of Interest selector Annotation Tool interface

The input to the car image classification annotation tool is a video file. The annotation tool is used to traverse across the frames of the video. It also has a feature to crop portions of a video frame (refer to Figure 1), ask the annotator for the vehicle's classification, and save the cropped portion into an image file wherein the classification is embedded in the filename. This annotation tool saves the image files in the *.jpg* format. This annotation tool is a software we developed and we used the OpenCV library to create this software. A design consideration of this annotation tool is to require minimum interaction to the user such as automatic traversal of frames and a cursor based interface for selection of region of interest.

[2] Open-source video player downloadable from https://www.videolan.org/vlc/index.html

[3] A popular spreadsheet application downloadable from https://products.office.com/en/excel

[4] LabelMe website http://labelme.csail.mit.edu/Release3.0/

2.4 Trajectory Annotation

Trajectory annotation information is used in analyzing the accuracy of our traffic monitoring system's ability to faithfully track a vehicle. Aside from the annotation tools, we had also developed a CV based multiple object tracking algorithm that produces trajectories of objects from a scene. A trajectory contains information such as: points as to where, and a time informaton of when an object appeared in a scene.

The annotators are provided with a set of video files, automatically-generated trajectory files using an initial tracking system, and a trajectory annotation tool. Annotators are given two tasks: (1) to review and delete visually-observable erratic automatically-generated trajectory files, and (2) to create new trajectory files for each vehicle that is missed by the initial tracking system, and to replace the erratic trajectories that they have deleted.

Figure 2. Sample Trajectory Plot

A trajectory is composed of a triple of files: a snippet video containing the vehicle being tracked, a trajectory plot, and a binary file containing the trajectory of the vehicle when imposed into the source video. A sample trajectory plot image is shown in Figure 2. A trajectory plot, together with a snipped video is what the annotators use to review the trajectories.

The trajectory annotation tool is used to create new trajectories. The input to the annotation tool is a video file. And the annotator will traverse through the frames of the video, select portions of some frames where the vehicle appears, similar to what is shown on Figure 1. Based on the selected portions, the tool will generate the triple of files that compose a trajectory. This annotation tool is a software we developed and we used the OpenCV library to create this software.

3. RESULTS AND DISCUSSION

We employed 7 annotators and gave them a task to count vehicles from 280 video files divided into sets. Each set consists of 40 5-minute long videos. An annotator is provided with two different sets, but in order to measure the inter-rater agreement, pairs of annotators were given similar sets. The resulting inter-rater correlation of 0.4308 (40%) will serve as the benchmark for measuring our vehicle counter's performance. The human performance will serve to benchmark the traffic monitoring system performance.

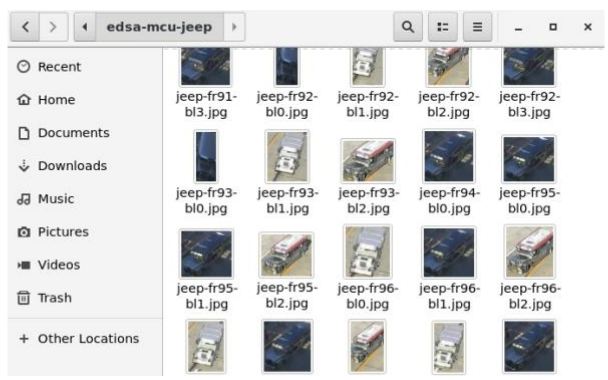

Figure 3. File manager screenshot of car image annotation

The annotators are next given 27 videos from which they would extract images of vehicles using the car image classification annotation tool. This yielded a dataset with 13,741 images of vehicles. Figure 3 shows a screenshot of a file manager with some of the images created from the car image annotation task.

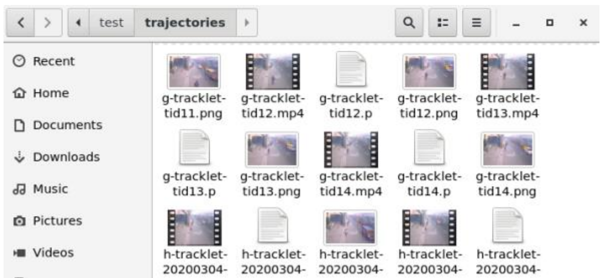

Figure 4. File manager screenshot of trajectories

The annotators are next given 4 videos each and tasked to review the automated trajectories and create new trajectories, when necessary. Figure 4 shows a file manager screenshot showing the files for automatically-generated trajectories are found in the same folder as the trajectories made by the human annotators.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we described techniques that we have employed in the creation of three kinds of datasets in developing a traffic monitoring system. The datasets created are vehicle counts, car image classifications, and vehicle trajectories. Existing software have been used to aid the annotation task such as VLC media player and Microsoft Excel, but we also developed two annotation tools for car image classification annotation task and trajectory annotation task. The use of the developed annotation tools has greatly aided in speeding up the annotation task compared to when using existing available software such as the Microsoft snipping tool.

Two of the annotation tools developed are made to create a dataset used to measure the performance of traffic monitoring systems. A continuation of this work is to also develop tools for measuring the performance of traffic monitoring systems.

5. ACKNOWLEDGMENTS

The TITAN project, under which the annotation tools were developed, was funded by the Department of Science and Technology-Philippine Council for Industry, Energy, and Emerging Technology Research and Development (DOST PCIEERD) and De La Salle University Research Coordination Office (DLSU URCO).

The authors also acknowledge the main annotators of the project: Aaron Fajardo, Candace Claire Mercado, Christian Dequito, Mohammad Minaga, Raphael Zapanta, Rhei Justin Juan, and Saira Kaye Manalili, and the Philippine Science Junior High School Interns who also did some annotation tasks: Aaron Joseph Yu, Alvaro Joseph Loste, Frodel Bennedict Cyril Gonzalez Pascua, Jerico Wayne Bayod, Jilliane Evangelista, Joel Miguel Leysa, Rheana Mindo, and Thristan Raotraot.

6. REFERENCES

Chen, J., Wang, D., Xie, I., & Lu, Q. (2018). Image annotation tactics: transitions, strategies, and efficiency. Information Processing & Management. Volume 54, Issue 6, pp. 985-1001

Cempron, J. and Ilao, J. (2019). VEMON: Vision-based Vehicle Counter for Traffic Monitoring. DLSU Research Congress 2019. Manila, Philippines

Deng, J., Dong, W., Socher, R., Li, L., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. 2009 IEEE Conference on Computer Vision and Pattern Recognition. DOI:10.1109/cvpr.2009.5206848

Everingham, M., Eslami, S. M. A., Van Gool, L., Williams, C. K. I., Winn, J. and Zisserman (2015). The PASCAL Visual Object Classes Challenge: A Retrospective. International Journal of Computer Vision, 111(1), 98-136

Ty, L. et al. (2014) Microsoft COCO: Common Objects in Context. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds) Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science, vol 8693. Springer, Cham

Krasin I., Duerig T., Alldrin N., Ferrari V., Abu-El-Haija S., Kuznetsova A., Rom H., Uijlings J., Popov S., Kamali S., Malloci M., Pont-Tuset J., Veit A., Belongie S., Gomes V., Gupta A., Sun C., Chechik G., Cai D., Feng Z., Narayanan D., Murphy K. (2017). OpenImages: A public dataset for large-scale multi-label and multi-class image classification. arXiv:1811.00982v2

Liu, L., Ouyang, W., Wang, X. et al. Deep Learning for Generic Object Detection: A Survey. (2020). Int J Comput Vis 128, 261–318

Russell, B., Torralba, A., Murphy, K., Freeman, W. T. (2008). LabelMe: a database and web-based tool for image annotation. International Journal of Computer Vision