Video Dataset Labeling using Active Learning with Applications in Vehicle Classification and Traffic Flow Rate Measurement

Adonais Ray Maclang*, Miguel Lorenzo Orante†, Rennuel Don Salvador‡,
Dale Joshua del Carmen §, and Rhandley D. Cajote ¶
Electrical and Electronics Engineering Institute
University of the Philippines–Diliman
Quezon City, Philippines
*adonais.ray.maclang@eee.upd.edu.ph, †miguel.lorenzo.orante@eee.upd.edu.ph, ‡rennuel.don.salvador@eee.upd.edu.ph,
§dale.del.carmen@eee.upd.edu.ph, ¶ rhandley.cajote@eee.upd.edu.ph

***Abstract*—Intelligent Transportation Systems (ITS) offer a means to increase efficiency in road management, safety, and traffic enforcement. The Philippines, particularly Metro Manila, is notorious for its high levels of traffic congestion result in significant economic loss. It is possible to accumulate large amounts of traffic video data by installing traffic cameras but the manual preparation of such custom datasets for ITS applications is taxing and laborious. In this paper, we develop an algorithm for vehicle detection, classification, and flow rate measurement algorithm for ITS applications using YOLOv7 and a tracker trained on a custom dataset provided by the UP National Center for Transportation Studies (NCTS) augmented with active learning algorithm. Its detection, classification, and tracking performance were compared to that of a model trained without using active learning. The results indicate that using uncertainty-based active learning algorithm is effective in improving the model's tracking capability. The best results from the active learning models with the tracker was able to achieve a higher HOTA value of 67.423 vs 67.355 (+0.068%) for the first evaluation, and 71.652 vs 70.614 (1.038%) for the second evaluation on the NCTS tracking evaluation sets. For a specific sequence in DETRAC, the improvement is 69.842 vs 69.84 (+0.002%). At the third cycle of training, the active learning model counts better with a total count of 98 vs 100 from a true count of 91.**

***Index Terms*—active learning, detection, classification, tracking, flow rate**

I. Introduction

Transportation has a vital role in the economic development of every country as it facilitates the movement of goods and people from one place to another. The development of efficient transportation systems has a direct impact on the growth and progress of a nation, making it crucial for governments to invest in this sector. Inefficient transportation systems have caused huge opportunity costs, decreased levels of safety for vehicles and pedestrians alike, and overall degradation of the quality of life [1], negatively affecting the country's economy and social development. These issues are evident in urbanized areas such as Metro Manila. Studies in 2018 and 2022 have shown that Manila has a 43% congestion level and that, on average, Filipinos waste around four days per year in its traffic jams, equivalent to 3.5 billion pesos of loss in productivity [2], [3].

Intelligent Transportation Systems (ITS) have the potential to make traffic systems safer and more efficient by incorporating modern information and electronic technologies in the management of traffic elements. ITS that is design carefully can provide real-time traffic information to drivers and commuters alike, allowing them to choose alternative routes and reduce congestion. With recent advancements in both hardware and software technologies, video-based detection systems have become an essential component in ITS and can therefore play a critical role in delivering data for road planning and traffic management applications such as automated incident detection, automated red light reinforcement, more efficient emergency response, etc. [4].

Deep learning approaches in ITS have consistently produced better results than existing statistical and analytical techniques [5] with Convolutional Neural Networks (CNN) being the most commonly used in visual recognition tasks (detection and tracking) [5]. These systems having been designed to learn higher features in image samples by exploiting patterns in adjacent pixels [6]. YOLO utilizes a single convolutional network that produces predictions for multiple objects in bounding boxes and at the same time classifies them. [7]. The current version of YOLOv7 at the time of this writing, surpasses all known detectors in speed and accuracy, from 5 FPS to 160 FPS, and has the highest accuracy among all known real-time detectors, with 56.8% AP at 30 frames per second [8]. This makes it an ideal model for applications that require both high accuracy and low latency. The implementation of these technologies, however, requires a significant amount of video data to be labels and annotated, which can be a challenge for transportation agencies that lack the necessary trained personnel to do this.

In order to develop ITS algorithms applicable in the Philippines, a custom dataset that covers the country's wide variety of vehicles under different road conditions is ideal if not necessary. But this requires a labor-intensive human-in-the-loop system to process all the relevant information in the dataset and to ensure the accuracy of the information derived from it. In the traditional passive learning method, large amounts of labeled data must be produced to enable supervised learning of a model. Active learning algorithms try to maximize the efficiency of labeling data by choosing the data samples that are most informative for the model to learn from. This approach can significantly reduce the

979-8-3503-0219-6/23/$31.00 ©2023 IEEE

amount of labeled data needed, making it more efficient and cost-effective while being able to produce results comparable to those produced by traditional supervised learning methods.

II. Methodology

A. Dataset Preprocessing and Annotation

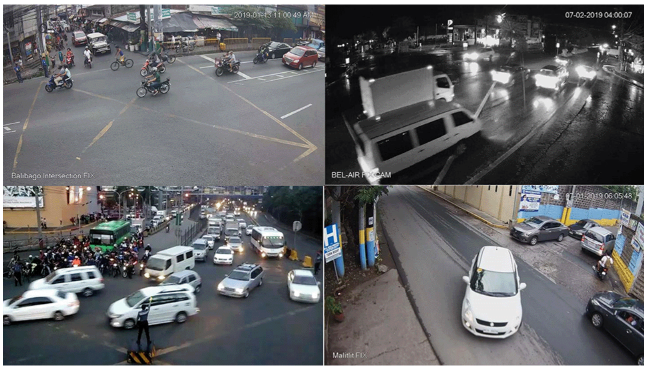

Fig. 1. Sample images from the NCTS dataset

Researchers at the University of the Philippines National Center for Transportation Studies (UP-NCTS) have collected various video data in key areas of Metro Manila for the purposes of improving traffic planning, management, and policy-making. We will refer to this video data collected by UP-NCTS in this paper as the NCTS dataset. This includes the handheld camera and CCTV videos taken from the city of Santa Rosa, Bonny Serrano, and Bicutan Interchange, as could be seen in Figure 1. The NCTS dataset though extensive is still unlabeled and not publicly available. The resolution of the images vary from 640 x 480, 720 x 480, 1280 x 720, and 2224 x 108 for the handcam videos, while the CCTV videos has a resolution of 1920 x 1080. Initially we will take 20,000 unlabeled images (video frames) from this dataset, label them and train our models using active learning.

To create a base model that can better estimate the uncertainty of the unlabeled data by lessening the number of unfamiliar instances, the model was trained using data from different locations and varying lighting conditions, such as sunny, cloudy, and dawn. A total of 1,000 frames were manually labeled initially, and 500 frames were added to each learning cycle until 2,500 frames had been labeled for the active learning dataset. For the random sampling dataset, the initial 1,000 frames without active learning yet will be reused, and an additional 1,500 frames was randomly sampled for a total of 2,500 images.

Computer Vision Annotation Tool (CVAT) was used to label the images. CVAT supports labeling in different formats like YOLO, COCO, KITTI, and others. This makes it easier to annotate and convert formats as needed.

B. Active Learning

Here we discuss the active learning algorithm that was implemented alongside the model training. Its performance in vehicle detection and classification were tested and evaluated. A summary of this is shown in Figure 2.

To determine the informativeness of an unlabeled sample x containing multiple detections, [9] defines a basic uncertainty sampling strategy using the least confidence metric v_{LC} given as,

$$v_{LC}(x) = \arg\max_{x} 1 - P_{\theta}(\hat{y}|x), \tag{1}$$

where $\hat{y} = \arg\max_x P_{\theta}(y|x)$ the prediction with the highest probability under model θ. With this metric, a simple and efficient scoring system proposed by [10] may be used. In this system, aggregation methods namely sun (Sum), average (Avg), and maximum (Max) are devised to score a whole image. Avg takes the average or the sum of the metric over the number of total detections in the image sample. Taking the average may be biased towards images with many detections D [11]; however, this will make the score more comparable between image samples.

$$v_{Avg}(x) = \frac{1}{|D|} \sum_{i \epsilon D} v_{LC}(x_i) \tag{2}$$

The study by [10] proposed a method of handling the class imbalance in the dataset that may cause a bias towards the majority class and issues in the model's classification accuracy. Detection scores $v_{LC}(x)$ were weighted before aggregation according to the formula,

$$w_c = \frac{\#instances + \#classes}{\#instances_c + 1} \tag{3}$$

Where c is the predicted class. This optimization tries to counter the imbalance by putting more weight into instances of the underrepresented classes.

The YOLOv7 model was retrained after three active learning cycles and evaluated to get its performance metrics and the tracking capability of different variations of the model was tested. The different models were integrated with a tracker using BYTETrack [12] to be able to perform vehicle tracking. The output from the YOLOv7 and BYTETrack network was used to compute the flow rate. This model enhanced by active learning will be compared with the model trained on random sampling. The results from this comparison will provide insights into the effectiveness of active learning in improving the accuracy of vehicle tracking and flow rate computation. Additionally, the findings could have practical implications for traffic management and urban planning.

The active learning method used is an uncertainty-based black-box method that will only be making use of the classification confidence scores from the YOLOv7 model inferences. Once the base model has been trained with the initial 1000 images, confidence scores from the remaining 19,000 images from the unlabeled image pool will be taken and ranked using the simple least confidence metric found in (1). Using the Average (Avg) aggregation method in (2) to rank the unlabeled images, the top-500 highest-scoring images were then used and manually labeled again to ensure that the labels are correct, with the lower confidence score starting at 0.001 per detection. It is important to note that there will also be a class imbalance due to the disproportionate distribution of vehicles in the dataset. Weights calculated in (3) were multiplied to each vehicle detection per image to try to address this issue.

979-8-3503-0219-6/23/$31.00 ©2023 IEEE

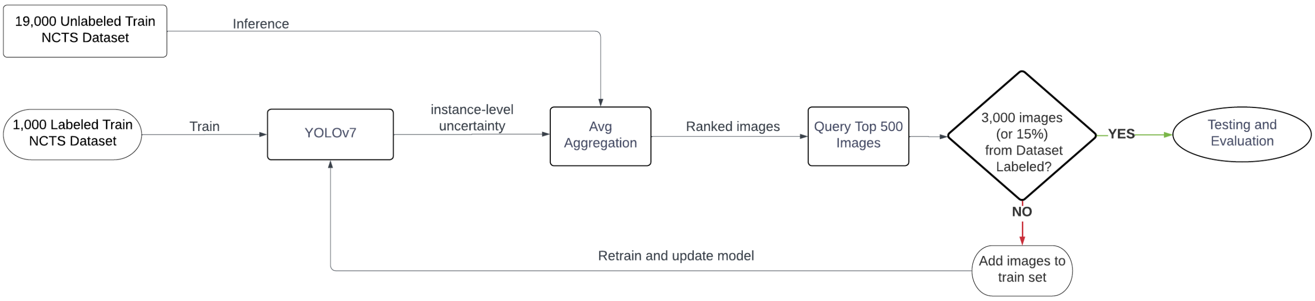

Fig. 2. Overview of the Active Learning Algorithm.

C. YOLOv7 Training and Testing

A total of 2500 images was used to train different models on the active learning algorithm and on randomly sampled data while 1,000 images will be used for testing. The mean average precision (mAP) of the model from each active learning cycle will be compared to verify its improvement. A separate set containing 500 images from the Tagapo CCTV is used to test the model's performance in a location not included in its training data. The resolution of the images will be scaled down to 640 x 640 as it's the default image size used for the YOLOv7 model.

D. Vehicle Tracking

ByteTrack [12] is a multiple object tracker that uses a simple, effective, and generic association method that tracks by associating every detection box instead of only the high-scoring ones, which improves the tracking performance. The similarities of low-score detection boxes are utilized with tracklets to recover the object detections and filter out the background detections. This makes it highly effective in crowded scenes where objects are closely packed together and overlapping. The tracker ranks first on benchmarks such as MOT17 [17] and MOT20 [18] while having the highest running speed among the trackers on the leaderboard as of this writing.

ByteTrack is used as the tracking algorithm with the YOLOv7 model trained in the NCTS dataset. The flow rate was obtained using the formula in (4) with the information obtained from the YOLOv7 and ByteTrack network, utilizing the number of vehicles it has counted. The algorithm uses the maximum and minimum ID set by the tracker for each frame for a specific interval. This is multiplied by the framerate of the video sample to get the flow rate in vehicles/sec.

$$flow\_rate = \frac{ID_{max} - ID_{min} + 1}{frame\_interval} \cdot fps \tag{4}$$

To measure and better visualize the amount of traffic, the average amount of time vehicles stay within the video frame k can be measured as in (5). For each vehicle tracked by the system with an ID, it will count for the number of frames that ID was present in the system, and this value will be summed together with all vehicles counted. The models will be compared to see how much active learning can improve the overall performance of the system compared with random sampling.

$$k = \frac{\sum_{i=1}^{N}(vehicle\ duration\ in\ frames)_i}{total\ number\ of\ vehicles(N)} \tag{5}$$

E. Evaluation

The performance of YOLOv7 models will be evaluated by comparing its mAP with the test set of 1,000 images, as well as the separate set of 500 images from Tagapo to see its performance in an unfamiliar environment. Together with ByteTrack, the tracking performance will be primarily evaluated on the HOTA metric, as well as the CLEAR-MOT and IDF1 metrics [13]. The counting will also be measured to see how it deviates from the true count to see the accuracy of flow rate measurement since the flow rate is simply the counting of vehicles per given time segment.

Higher Order Tracking Accuracy (HOTA) is a multiple object tracking evaluation metric that gives equal importance to detection and association accuracy, measuring how well the trajectories of matching detections align while penalizing detections that do not match. The identity metric IDF1 focuses on measuring association accuracy rather than detection accuracy by combining ID-Precision and ID-Recall. Lastly, the CLEAR-MOT metrics, Multiple Object Tracking Precision (MOTP) and Accuracy (MOTA), are overall performance measures showing how well the locations of the objects are estimated (MOTP) and how many mistakes the system made in terms of false negatives, false positives, and ID mismatches (MOTA).

III. Results and Discussion

The training and testing of the YOLOv7 models were done on a machine with Ryzen 5 4600H and GTX 1650 with 16GB of RAM. This was our primary system where the evaluation on tracking was also performed, achieving almost 15 fps with a resolution of 1600 x 700 on our evaluation video for tracking. Another machine with Intel Core i7-8750H and GTX 1050 with 8GB of RAM was also used to help in training for the third cycle of both the active learning and random sampling model. The evaluation of the whole system is done using TrackEval [14]. TrackEval is a codebase for different tracking evaluation metrics, which include the HOTA, CLEARMOT, and Identity metrics, that provides support on a number of different tracking benchmarks.

The training was done using the default parameter settings of the YOLOv7 model, which can be found in the YOLOv7 repository [8] from the directory "yolov7/data/scratch.p5 yaml" containing the values of the 30 parameters used in training. The models were trained for 150 epochs instead of the default value of 300 epochs for faster training.

In this section, the models trained with the help of the active learning algorithm will be denoted as A1 (first cycle), A2 (second cycle), and A3 (third cycle). Similarly, the models generated and trained using randomly sampled data will be denoted as R1, R2, and R3.

A. Active Learning Algorithm

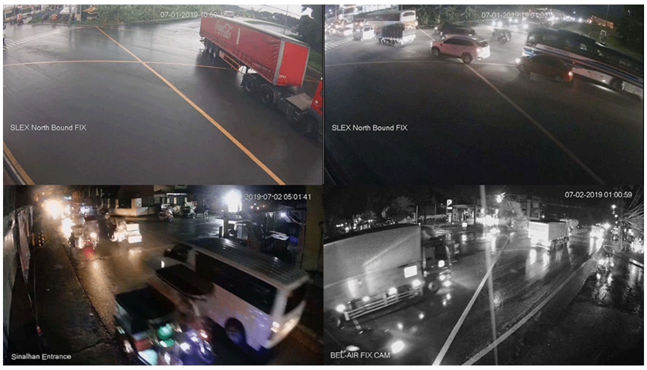

Fig. 3. Sample images picked by the Active Learning Algorithm

Figure 3 shows some of the images picked by the active learning algorithm. It can be observed that the majority of the images are of dark, rainy, or occluded conditions. This suggests that the active learning algorithm is able to identify challenging scenarios where the model may struggle and select images that will improve its performance in those conditions.

In table I, it can also be observed how the algorithm tries to prioritize images containing underrepresented classes such as buses and trucks because of the implemented weighting system. This is shown in the table below, where the distribution of classes in the random sampling does not differ much from the base model, while the active learning algorithm reduces the number of cars and motorcycles while increasing the number of instances for other classes.

TABLE I
COMPARISON OF THE CLASS DISTRIBUTION BETWEEN MODELS

Model	bus	car	jeep	motor	trike	truck	van
Base	3.59	35.6	6.92	26.74	11.39	7.51	8.24
R1	3.48	36.16	6.82	25.84	11.65	7.68	8.36
R2	3.53	35.77	6.84	25.99	11.91	7.59	8.38
R3	3.51	35.6	6.88	25.86	12.03	7.58	8.53
A1	3.72	33.72	6.93	25.21	11.95	9.44	9.03
A2	3.65	32.97	6.73	24.51	12.67	10.5	8.98
A3	4.35	31.14	7.16	20.97	16.24	11.33	8.80

B. YOLOv7 Detection and Classification

As can be observed in Table II, the YOLOv7 models can be seen improving their mAP when the number of images increases. In here, R0 and A0 are the models trained only on the first 500-image set of each method. Looking from 100 images to 1000 images of the base model, the mAP increases as well and the active learning algorithm when having 500 images total (adding 250 images picked by the active learning algorithm to the previous 250 images) has a mAP advantage of 1.5% (65.8% vs 64.3%) to see the effectiveness of the active learning even at an earlier stage. After 1000 images, the mAP improvements become smaller. The mAP on the first cycle of active learning improves but lowers on the second and third cycles. Random sampling improves the map on the second and third cycles but has no changes at the first cycle, having slightly lower mAP@0.5 (84.6 vs 85.5) compared to the base model. One of the reasons for slight decline in performance during active learning on later parts is that the 2nd and 3rd cycle had 21.1% and 18.4% of its training set have background images, unlike the 1st cycle having 12.4% and the random sampling cycles less than 10%. Training YOLO models recommends having background images limited to about 10% of the training set. The decrease in mAP may also be due to having lesser amounts of training data, where during the third cycle of training there are 31,781 detections for random sampling and only 19,002 instances for the active learning.

TABLE II
TRAINING RESULTS OF DIFFERENT MODELS ON NCTS DATASET

Model	mAP
100 images	0.509
250 images	0.605
Base	0.673
R0	0.643
R1	0.673
R2	0.677
R3	0.682
A0	0.658
A1	0.678
A2	0.667
A3	0.675

TABLE III
EVALUATION OF DIFFERENT MODELS ON TAGAPO TEST SET

Model	mAP
100 images	0.323
250 images	0.364
Base	0.436
R0	0.425
R1	0.456
R2	0.477
R3	0.474
A0	0.447
A1	0.528
A2	0.466
A3	0.441

As in Table III, when using the Tagapo test set, a similar trend is shown, but the A1 model had the best mAP out of them all. This shows that the first active learning cycle of the model gives the biggest advantage compared to random sampling with a starting low confidence score of 0.001 for a different environment as well as a similar environment from the first test set. The second and third cycle has the random sampling having a slight advantage compared to active learning.

For testing the vehicle classification, a confusion matrix is generated. For the base model, the results show that vehicles can be classified accurately, where the true positive ranges from 0.63 to 0.88, and that from backgrounds, there can be false positives of cars and motorcycles. False negatives can often occur with buses and tricycles, where the model thinks

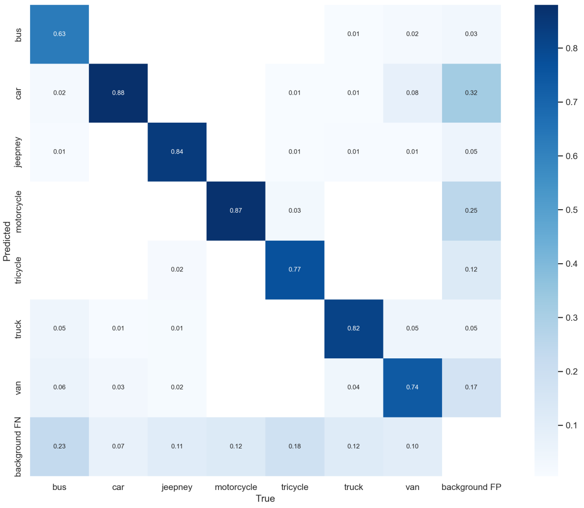

Fig. 4. Confusion Matrix for the Base Model

of them as a background when they should have been labeled. There are no significant changes from the models in terms of the confusion matrix.

C. YOLOv7 + ByteTrack Testing

TABLE IV
COMPARISON OF TRACKING RESULTS BETWEEN MODELS

Model	HOTA	MOTA	MOTP	IDF1	Count	True Count
A1	66.254	79.360	79.736	89.881	17	17
A2	66.556	76.066	79.378	88.310	17	17
A3	67.423	78.585	79.473	89.329	17	17
Base	66.127	77.907	78.962	87.837	18	17
R1	65.725	79.264	78.456	88.456	18	17
R2	65.453	79.651	78.519	88.794	18	17
R3	67.355	78.973	78.934	89.701	17	17

In Table IV, the tracking threshold used was 0.8 for the comparison of models, as it yielded the best HOTA value using the base model by tracking only the car class for simpler evaluation. The first two cycles of active learning improve the HOTA compared to random sampling while worsening for the random sampling. In the third cycle, however, random sampling improves over previous cycles, but the active learning algorithm still yields the best result for the HOTA value as shown by the increasing trend from each cycle. The random model took until the third cycle before it can match the active learning models. This shows that flowrate can be accurate with no errors when the active learning models are used, while the random sampling models can deviate by up to 5.88% and needs more images to be more accurate in counting.

TABLE V
DETRAC EVALUATION IN [15]

Det Thres	Recall	Prec	FP	FN	IDs	MOTA	MOTP
0.3	73.362	93.812	291	1602	2	68.490	74.559
0.7	59.628	95.985	150	2428	0	57.133	75.019
1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Comparing the system to a similar paper for ITS in the Philippines using YOLOv4 and a KLT Tracker, the sequence MVI_20011 of UA-DETRAC dataset [15] in Figure V was used to show the difference between their performance in terms of the CLEAR-MOT metrics, at least between the MOTA and MOTP metrics common between them [16].

TABLE VI
COMPARISON OF TRACKING RESULTS BETWEEN MODELS ON DETRAC

Model	HOTA	MOTA	MOTP	IDF1	Count	True Count
A1	69.842	80.732	79.992	90.124	54	49
A2	69.740	79.853	79.998	89.715	54	49
A3	68.952	79.824	79.904	89.561	53	49
Base	68.636	77.144	79.718	88.798	55	49
R1	69.465	78.435	79.890	89.518	57	49
R2	67.598	78.066	79.311	88.586	54	49
R3	69.84	80.703	79.442	90.122	56	49

The system developed in this paper as seen in Table VI is able to achieve a better tracking performance, with its best MOTA higher at 80.732 (vs 68.49), and best MOTP of 79.998 (vs 75.019) which shows how a better network can improve overall performance. Only cars have been tracked in both evaluations so they can be compared directly. Furthermore, the flow rate and counting may deviate from the true count by 8.2% or up to 10.2% (from active learning) or 16.3% (from random sampling) depending on the final model used. Although the third cycle of the random sampling model had the highest mAP from our testing for the NCTS dataset, it still performed worse in terms of counting, with the first cycle of the random sampling model being the worst. The increased error rate can be attributed to the different environment used for evaluation, unlike the NCTS testing set where the error rate can go as low as 5.85% or none at all. A screenshot with labels from the sequence MVI_20011 of the UA-DETRAC dataset is shown below, where only the region of interest is being evaluated excluding areas like the parked cars.

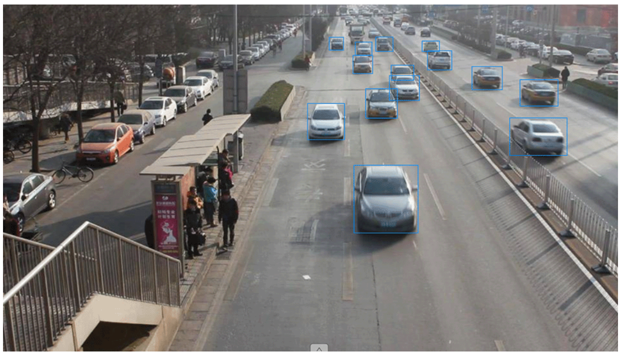

Fig. 5. MVI_20011 sample frame

Shown in figure 6 is a sample of the traffic flow rate in action, where the number of vehicles per second is being measured. This value measures how many new vehicles enter the frame per second. The metric k is also shown to measure how much traffic is currently present in terms of frames; the accuracy of this value is affected by how accurate tracking is, and missed IDs on some frames can affect this even if the same ID re-appear on the same vehicle.

Overall, the results show that active learning is effective in improving the tracking performance of the system. The

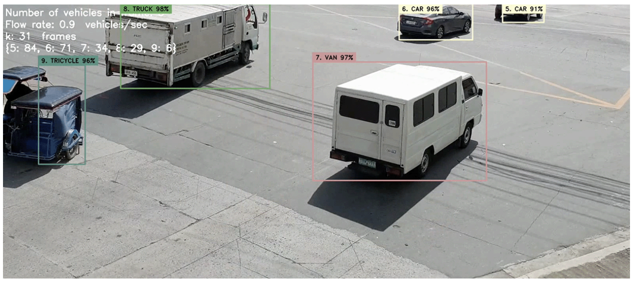

Fig. 6. Sample of Flow Rate and K measurement

active learning algorithm tends to pick rainy or low light conditions more as expected, but it also comes with the addition of many background images, and these background images at later cycles may no longer be as helpful, unlike the first cycle of active learning where there was a noticeable mAP improvement over random sampling. There also tend to be fewer instances of training data for the active learning algorithm in an attempt to balance these classes because the class distribution of the NCTS dataset is biased toward cars. Despite being weaker in terms of mAP at later cycles during the detection stage, the active learning models still perform better together with the tracker combined as compared to the random sampling models as shown during the tracking evaluation.

IV. Conclusion

An object detection model that is able to detect, classify and track vehicles was developed with the help of an active learning algorithm. The effectiveness of the active learning algorithm in the application of ITS is compared by using a model trained on randomly sampled data. After three random cycles and three active learning cycles, it was shown that despite being slightly less accurate at later cycles during the detection stage, the active learning models were able to track the vehicles better compared to the random sampling models. The lack of significant difference between the models in their detection accuracy may be attributed to the inherent imbalanced class distribution, excessive vehicle variation per class, and occlusions in the NCTS dataset that may have limited the model's ability to generalize.

In the future, it is highly recommended to explore more complex active learning algorithms as well as other object detection/classification methods. The implementation of Region of Interest in training may also be investigated for potential improvements in the performance of the model.

V. Acknowledgement

The researchers would like to thank Sheila Flor Javier of the UP National Center for Transportation Studies for allowing us to use the NCTS video dataset to explore efficient labeling algorithms.

References

[1] G. Dimitrakopoulos and P. Demestichas, "Intelligent transportation systems", IEEE Vehicular Technology Magazine, vol. 5, no. 1, pp. 77–84, 2010. DOI: 10.1109/MVT.2009.935537.

[2] Manila is the world's 8th city with longest hours spent in traffic — study. [Online]. Available: https://www.philstar.com/business/2022/09/08/2208311/manila-worlds-8th-city-longest-hours-spent-traffic-study.

[3] "PH loses ₱3.5B a day due to Metro Manila traffic – JICA," CNN. https://www.cnnphilippines.com/transportation/2018/02/23/JICA-P3.5-billion-traffic.html

[4] M. Bommes, A. Fazekas, T. Volkenhoff, and M. Oeser, "Video Based Intelligent Transportation Systems ˆa State of the Art and Future Development", en, Transportation Research Procedia, Transport Research Arena TRA2016, vol. 14, pp. 4495–4504, Jan. 2016, ISSN: 2352-1465. DOI: 10. 1016 /j.trpro.2016.05.372.

[5] A. K. Haghighat, V. Ravichandra-Mouli, P. Chakraborty, Y. Esfandiari, S. Arabi, and A. Sharma, "Applications of Deep Learning in Intelligent Transportation Systems", en, Journal of Big Data Analytics in Transportation, vol. 2, no. 2, pp. 115–145, Aug. 2020, ISSN: 2523-3564. DOI: 10.1007/s42421-020-00020-1

[6] T. A. Yang and D. L. Silver, "The Disadvantage of CNN versus DBN Image Classification Under Adversarial Conditions", en, Proceedings of the Canadian Conference on Artificial Intelligence, Jun. 2021. DOI:10.21428/594757db.b65acd40.

[7] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection", en, in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA: IEEE, Jun. 2016, pp. 779–788, ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.91.

[8] C. Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors, arXiv:2207.02696 [cs], Jul. 2022. [Online]. Available: http://arxiv.org/abs/2207.02696.

[9] B. Settles, "Active Learning", Synthesis Lectures on Artificial Intelligence and Machine Learning, vol. 6, no. 1, pp. 1–114, Jun. 2012, Publisher: Morgan & Claypool Publishers, ISSN: 1939-4608. DOI: 10.2200/ S00429ED1V01Y201207AIM018.

[10] C. A. Brust, C. Käding, and J. Denzler, "Active Learning for Deep Object Detection." arXiv, Sep. 26, 2018. doi: 10.48550/arXiv.1809.09875.

[11] E. Haussmann et al., Scalable Active Learning for Object Detection, arXiv:2004.04699 [cs], Apr. 2020. DOI: 10.48550/arXiv.2004.04699.

[12] Y. Zhang et al., "ByteTrack: Multi-Object Tracking by Associating Every Detection Box." arXiv, Apr. 07, 2022. doi: 10.48550/arXiv.2110.06864.

[13] R. Khandelwal, "Evaluation Metrics for Multiple Object Tracking," Medium, May 11, 2022. https://arshren.medium.com/evaluation-metrics-for-multiple-object-tracking-7b26ef23ef5f.

[14] J. Luiten et al., "HOTA: A Higher Order Metric for Evaluating Multi-Object Tracking," International Journal of Computer Vision, pp. 1–31, 2020.

[15] "The UA-DETRAC Benchmark Suite." https://detrac-db.rit.albany.edu/

[16] R. C. Castro, H. Mamugay, and K. Panti, "Vehicle Detection and Classification using YOLOv4 and KLT Tracker."

[17] P. Dendorfer et al., "MOTChallenge: A Benchmark for Single-Camera Multiple Target Tracking." 2020.

[18] P. Dendorfer et al., "MOT20: A benchmark for multi object tracking in crowded scenes." 2020.