DriveCLIP: Zero-shot transfer for distracted driving activity understanding using CLIP

Md Zahid Hasan[1], **Ameya Joshi**[2], **Mohammed Rahman**[1], **Archana Venkatachalapathy**[1]
Anuj Sharma[1],**Chinmay Hegde**[2], **Soumik Sarkar**[1]
[1]Iowa State University [2]New York University
{zahid,shaiqur,archanav,anujs, soumiks}@iastate.edu
{ameya.joshi,chinmay.h}@nyu.edu

Abstract

Distracted driving action recognition from naturalistic driving is crucial for both driver and pedestrian's safe and reliable experience. However, traditional computer vision techniques sometimes require a lot of supervision in terms of a large amount of annotated training data to detect distracted driving activities. Recently, the vision-language models have offered large-scale visual-textual pre-training that can be adapted to unsupervised task-specific learning like distracted activity recognition. The contrastive image-text pretraining models like CLIP have shown significant promise in learning natural language-guided visual representations. In this paper, we propose a CLIP-based driver activity recognition framework that predicts whether a driver is distracted or not while driving. CLIP's vision embedding offers zero-shot transfer, which can identify distracted activities by the driver from the driving videos. Our result suggests this framework offers SOTA performance on zero-shot transfer for predicting the driver's state on three public datasets. We also developed DriveCLIP, a classifier on top of the CLIP's visual representation for distracted driving detection tasks, and reported the results here. `https://github.com/Zahid-isu/DriveCLIP`

1 Introduction

Distracted activity understanding while driving is essential for safety and reliability in transportation. According to National Highway Traffic Safety Administration (NHTSA), almost 3,142 people were killed in 2020 due to distracted driving in the U.S. Therefore, distracted driving action recognition has become a potential research problem. The objective of this study is to identify distracted driving actions by drivers from naturalistic driving videos.

Recently, vision-language-based multimodal pretraining frameworks have exhibited promising results. These multimodal pretraining frameworks use carefully crafted contrastive-loss (see CLIP [1]), which allows us to come up with a very robust embedding of text and vision tasks simultaneously. This approach allowed several domains to use an almost zero-shot approach towards tasks such as object classification and object detection from images [1, 2]. In this paper, we explore using CLIP's vision and language embeddings for distracted driving action recognition.

In distracted driving action recognition, we are provided with a video of the driver from a camera mounted on the dashboard of a car. This video contains a large time window of data on the driver's behavior. Our task is to use this video data to identify if the driver is distracted or not. For this, it seems reasonable to train a video-text model from scratch. However, directly training a language-video model is unaffordable because it requires large-scale video-text pertaining data and a massive number of GPU resources (e.g., thousands of GPU days). A feasible solution is to adapt the pretrained

Machine Learning for Autonomous Driving Workshop at the 36th Conference on Neural Information Processing Systems (NeurIPS 2022), New Orleans, USA.

language-image models to the video domain. Recently, several studies have explored how to transfer the knowledge from the pretrained language-image models to other downstream tasks [2–4].

The proposed approach can avail of two advantages the pretrained CLIP [1] model offers. First, it leverages the training-free zero-shot transfer. It is more feasible and scalable to use natural language supervised video activity understanding rather than fully relying on a supervised approach. The existing computer vision approaches learn to memorize human annotated labels or features, and most of the time, these features are non-transferable. Therefore, this model often does better on a specific dataset but fails to perform well on other datasets. Second, the visual representation can be further finetuned with minimal annotations and training to improve the overall performance.

In summary, the key contributions of this paper are:

1. We explore the idea of using zero-shot transfer of CLIP pretrained weights for the task of distracted driving action recognition without dataset-specific training.
2. With proper prompt engineering, we demonstrate superior performance on several datasets of distracted driver action recognition using zero-shot transfers from CLIP.

2 Related Works

Driver action recognition has been extensively studied over the last few years, but it still undergoes continuous research. Different approaches have been used in several studies. Lots of low-level hand-crafted feature engineering like Cuboids [5], 3DHOG [6], and Dense Trajectories [7, 8] were used to capture the temporal aspects in the video. However, most of these dataset-specific approaches were not end-to-end trained on a large-scale dataset and lacked generalization.

Before the era of transformers majority of the proposed approaches relied on different CNN networks [9–11] as feature extractors of video data. Some CNN-based frameworks collected facial detection [12], optical flow [13], and body pose estimation [14] based features to study video actions. However, these models fail to generalize well, only focus on one feature, and might collapse if the feature extractor fails. Although 3D CNNs [15, 16] can learn temporal features from video frames, it imposes a burden of computational cost and difficulty of implementation in the real-world video domain. Another common approach is ensembling multiple CNN networks [17, 18] like ResNet50 [19], Vgg19 [20], Xception[21], Inception-v3 [22] as different feature extractor and do weighted ensemble to create a global feature. Also, two streams' fusion-based approaches [23–25] can learn RGB appearance and temporal features individually that can be merged later. In recent times, vision transformer(ViT) based approaches [26–28] have taken over after CNN for the video action recognition tasks.

Despite decent performance on benchmark datasets, these models are critically dependent on large-scale datasets that require huge data annotation effort. Moreover, most of these models are trained to map a set of predefined categories that thwart learning the transferable visual concepts on unseen data. Instead of learning narrow visual features within a uni-modal framework, learning from natural language supervision will offer more flexibility and generality. Recently vision-language-based pretraining approaches based on CLIP [1] like VideoCLIP [3], and ActionCLIP [2] show advanced performance in video action recognition and understanding tasks.

3 Our approach

The main motivation of our work is to understand drivers distracted activity from videos and build an end-to-end driver monitoring system. By distracted driving activity, we mean any activity that diverts drivers' attention from safe driving. For example, talking on the phone, drinking, or eating. However, distracted driving activity detection became much more complicated because of the low-quality video data and the human labor involved in annotating the videos. Therefore, we need a generalized framework that performs well without solely relying on the labeled data. To accomplish this goal, we proposed a contrastive vision language-based pretrained framework. Figure 1 depicts the overview of our approach. The base model [1] is pretrained on around 400 million image-text pairs from the web, which excels in most visual understanding-based tasks. One key advantage of using a pretrained network over other approaches is connecting the "already learned" visual representation to language, which offers zero-shot transfer. We adopted the pre-trained visual embedding for zero-shot transfer

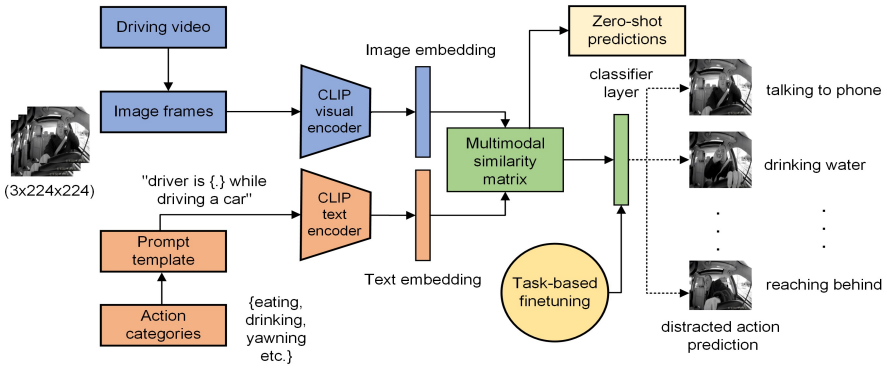

Figure 1: DriveCLIP multimodal framework exploits the pretrained CLIP pipeline to extract video-text semantic information.

Table 1: Distracted driving prompt categories

Prompt ID	Distracted actions	Prompt descriptions
0	adjusting	"driver is adjusting his or her hair while driving a car"
1	drinking	"driver is drinking water from a bottle while driving a car"
2	eating	"driver is eating while driving a car"
3	picking	"driver is picking something from floor while driving a car"
4	reaching	"driver is reaching behind to the backseat while driving a car"
5	singing	"driver is singing a song with music and smiling while driving"
6	talking	"driver is talking to the phone on hand while driving a car"
7	yawning	"driver is yawning while driving a car"

on driving datasets and fine-tuned a linear classifier for the distracted activity detection task. We referred to task-based fine-tuning as a step of training a simple classifier layer on top of the pretrained embedding for a specific task (object detection, action recognition, etc.) and evaluated the whole model's performance on a corresponding dataset.

4 Experiments

4.1 Dataset

We tested our approach on three driving datasets- SynDD1 [29], StateFarm [30] and AUC [17]. These datasets are commonly used as a benchmark for studying distracted driver behavior. The SynDD1 dataset consists of annotated videos of eighteen distracted action categories with three camera views. For this work, we merged similar categories and considered eight distinct actions shown in table 1. The StateFarm dataset has ten activity categories with images captured from the right-side camera view. The AUC dataset also contains ten activity categories having a right-side camera view.

4.2 Experimental setup and prompt engineering

In this study, we adapted the CLIP [1] framework in the driving domain. The zero-shot transfer and task-based finetuning branches were explored to evaluate the "task learning capability." To investigate the representation learning capability, we fit a linear classifier on top of the representation extracted from the model and measured its performance. For training the linear probe (see CLIP [1]), we used an NVIDIA Tesla T4 GPU with batch size 100. Since the original CLIP model employs semantic information in text labels rather than the traditional one-hot labels, it is necessary to provide a textual description or prompt for each action category. Therefore, We used full sentences as prompts for describing the classes instead of a single word. The prompt descriptions are shown in table 1. As reported in [1], prompt ensembling can increase the overall performance. We tried manually fine-tuning the prompts for running the experiments, which can be further optimized in the future.

Table 2: Zero-shot results on SynDD1 dataset (see model configs 4)

CLIP visual backbone	Top-1 accuracy	Top-5 accuracy
ViT-L/14	**53.50**	**74.50**
ViTL/14@336px	54.00	65.00
ViT-B/16	47.50	64.50
ViT-B/32	38.00	45.50

Table 3: Linear probe performance on SynDD1, StateFarm, and AUC dataset (see model configs 4)

CLIP visual backbone	accuracy on SynDD1	accuracy on StateFarm	accuracy on AUC	Benchmark StateFarm (2020)	Benchmark AUC (2019)
ViT-L/14	**85.666**	**98.226**	**88.609**	97.00 [31]	95.98[17]
ViT-B/32	75.478	96.988	84.786	method:	method:
ViT-B/16	73.567	96.205	84.327	CNN	AlexNet
RN101	65.197	90.553	73.624	ensemble	ensemble

5 Results

5.1 Zero-shot transfer

For zero-shot transfer, the unseen frames were inferred using the CLIP model. During the forward pass, the framework computes the visual and textual features from the input frames and the text prompts, respectively. Then, it computes the cosine similarity between the visual and textual features and provides a top-5 ranking of the most probable classes. The zero-shot results on the SynDD1 dataset are shown in table 2. The table shows that the vision transformer-based CLIP model with ViT-L/14 visual backbone performs better than the other variants for zero-shot transfer.

5.2 Linear probe experiments

To justify the task-based fine-tuning, we trained a linear classifier on top of the CLIP visual embedding for the datasets mentioned above. We separately plugged in four visual backbones as feature extractors and showed their corresponding linear probe results in table 3. Note that, for comparing the results with benchmark approaches, we considered the same camera views and classes. The best-performing visual encoder with ViT-L/14 architecture achieved around 85.67% accuracy on the SynDD1 dataset for ten classes. The average F1 score is 0.4109 for the original eighteen classes of the SynDD1 dataset, which outperforms the recent benchmark [32] of 0.3492. Also, this model outperforms some of the recent StateFarm benchmarks [31, 33, 34] that used CNN ensembling and attention-based approaches. It achieved around 88.61% accuracy for the AUC dataset compared to the recent benchmarks of 95.5-95.9%[17, 35]. Apart from the three vision transformer backbones, one ResNet-101 backbone (RN101) was also tested; however, it performed poorly compared to the vision transformers.

5.3 Class-level analysis

Table 6 shows the class-level F1-scores of DriveCLIP on the SynDD1 dataset. From this table and the confusion matrix shown in figure 2, we can conclude that "eating," "singing," and "yawning" were the confusing classes for this framework compared to other obvious classes. We found that these actions produce image features that are similar looking and need further optimization.

6 Conclusion and Future works

In this work, we proposed a vision-language-based framework to detect distracted driving behavior. This framework demonstrated its efficacy on three distinct datasets with superior performance compared to traditional deep models for the same task. Future work would be, optimizing the vision-language approach in terms of prompt learning can excel the existing approaches at distracted driving detection for naturalistic driving scenarios.

References

[1] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[2] Mengmeng Wang, Jiazheng Xing, and Yong Liu. Actionclip: A new paradigm for video action recognition, 2021.

[3] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding, 2021.

[4] Han Fang, Pengfei Xiong, Luhui Xu, and Yu Chen. Clip2video: Mastering video-text retrieval via image clip, 2021.

[5] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *2005 IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 65–72, 2005.

[6] Alexander Klaser, Marcin Marszałek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *BMVC 2008-19th British Machine Vision Conference*, pages 275–1. British Machine Vision Association, 2008.

[7] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60–79, 2013.

[8] Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *Proceedings of the IEEE international conference on computer vision*, pages 3551–3558, 2013.

[9] Bhakti Baheti, Suhas Gajre, and Sanjay Talbar. Detection of distracted driver using convolutional neural network. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1145–11456, 2018.

[10] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition, 2018.

[11] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification. In *Proceedings of the European conference on computer vision (ECCV)*, pages 305–321, 2018.

[12] Xuepeng Shi, Shiguang Shan, Meina Kan, Shuzhe Wu, and Xilin Chen. Real-time rotation-invariant face detection with progressive calibration networks, 2018.

[13] Neslihan Kose, Okan Kopuklu, Alexander Unnervik, and Gerhard Rigoll. Real-time driver state monitoring using a cnn based spatio-temporal approach, 2019.

[14] Peng Li, Meiqi Lu, Zhiwei Zhang, Donghui Shan, and Yang Yang. A novel spatial-temporal graph for skeleton-based driver action recognition. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 3243–3248. IEEE, 2019.

[15] Ali Diba, Mohsen Fayyaz, Vivek Sharma, M Mahdi Arzani, Rahman Yousefzadeh, Juergen Gall, and Luc Van Gool. Spatio-temporal channel correlation networks for action classification. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 284–299, 2018.

[16] Jonathan Stroud, David Ross, Chen Sun, Jia Deng, and Rahul Sukthankar. D3d: Distilled 3d networks for video action recognition. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 625–634, 2020.

[17] Yehya Abouelnaga, Hesham M. Eraqi, and Mohamed N. Moustafa. Real-time distracted driver posture classification. *CoRR*, abs/1706.09498, 2017.

[18] Chen Huang, Xiaochen Wang, Jiannong Cao, Shihui Wang, and Yan Zhang. Hcf: a hybrid cnn framework for behavior detection of distracted drivers. *IEEE access*, 8:109335–109349, 2020.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[20] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2014.

[21] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1800–1807, 2017.

[22] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[23] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[24] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.

[25] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1933–1941, 2016.

[26] Anurag Arnab, Mostafa Dehghani, Georg Heigold, Chen Sun, Mario Lučić, and Cordelia Schmid. Vivit: A video vision transformer, 2021.

[27] Haoqi Fan, Bo Xiong, Karttikeya Mangalam, Yanghao Li, Zhicheng Yan, Jitendra Malik, and Christoph Feichtenhofer. Multiscale vision transformers, 2021.

[28] Gedas Bertasius, Heng Wang, and Lorenzo Torresani. Is space-time attention all you need for video understanding?, 2021.

[29] Mohammed Shaiqur Rahman, Archana Venkatachalapathy, Anuj Sharma, Jiyang Wang, Senem Velipasalar Gursoy, David Anastasiu, and Shuo Wang. Synthetic distracted driving (syndd1) dataset for analyzing distracted behaviors and various gaze zones of a driver, 2022.

[30] https://www.kaggle.com/competitions/state-farm-distracted-driver-detection/data.

[31] Ketan Ramesh Dhakate and Ratnakar Dash. Distracted driver detection using stacking ensemble. In *2020 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS)*, pages 1–5. IEEE, 2020.

[32] Manh Tung Tran, Minh Quan Vu, Ngoc Duong Hoang, and Khac-Hoai Nam Bui. An effective temporal localization method with multi-view 3d action recognition for untrimmed naturalistic driving videos. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3167–3172, 2022.

[33] Chen Huang, Xiaochen Wang, Jiannong Cao, Shihui Wang, and Yan Zhang. Hcf: A hybrid cnn framework for behavior detection of distracted drivers. *IEEE Access*, 8:109335–109349, 2020.

[34] Yaocong Hu, Mingqi Lu, and Xiaobo Lu. Feature refinement for image-based driver action recognition via multi-scale attention convolutional neural network. *Signal Processing: Image Communication*, 81:115697, 2020.

[35] Ardhendu Behera, Zachary Wharton, Alexander Keidel, and Bappaditya Debnath. Deep cnn, body pose, and body-object interaction features for drivers' activity monitoring. *IEEE Transactions on Intelligent Transportation Systems*, 23(3):2874–2881, 2022.

A Appendix

A.1 DriveCLIP configuration

Table 4: Linear classifier hyperparameter list

Item	Description/values
Input image size	(3x224x224)
Batch size	100
Maximum iteration	1000
L2 regularization strength	0.42919
Hyper-parameter sweep range	[10^{-6}- 10^{6}]
Log-steps	50
Hyper-parameter solver	lbfgs
Search approach	GridSearchCV and HalvingGridSearchCV (scikit-learn)
Dataset used for search	Validation split
Model configuration	Visual-encoder: Vision transformer (ViT-B/32, ViT-B/16, ViT-L/14, ViTL/14@336px), ResNet (RN-101) ViT-B/32 (L=12, N=12, d=768, p=32) ViT-B/16 (L=12,N=12, d=768, p=16) ViT-L/14 (L=24, N=16,d=1024, p=14) ViTL/14@336px (ViTL/14 pretrained at 336 pixel resolution, input image size=3x336x336) Text encoder: CLIP text transformer (L=12, N=8) where L denotes the layers, N refers to the number of attention heads, d represents the embedding dimension, and p is the patch size
Temperature value(initial)	0.07
Maximum text sequence length	77 words
Maximum vocabulary size	49408

A.2 Class-level analysis

Table 5: Experiment details for class-level analysis

Item	Details
Model configuration	Visual encoder: ViT-L/14 Text encoder: basic CLIP text encoder
Dataset	SynDD1[29]
Class no.	08
Total drivers	14
Total frames	4405
Camera view	Dashboard
Train driver IDs	Total=10 ['76189', '61597', '25470', '79336', '56306', '65818', '49381', '76803', '24491', '19332']
Valid driver IDs	Total=2 [38058, 35133]
Test driver IDs	Total=2 [24026, 42271]
Maximum iteration	1000
L2 regularization value	0.42919

Table 6: Precision, recall, and F1-score for class-level analysis

Action classes	Precision	Recall	F1-score
"driver is adjusting his or her hair while driving a car"	0.51	0.74	0.60
"driver is drinking water from a bottle while driving a car"	0.70	0.86	0.78
"driver is eating while driving a car"	0.60	0.31	0.41
"driver is picking something from floor while driving a car"	0.76	0.96	0.85
"driver is reaching behind to the backseat while driving a car"	0.79	0.94	0.86
"driver is singing a song with music and smiling while driving"	0.44	0.30	0.36
"driver is talking to the phone on hand while driving a car"	1.00	0.77	0.87
"driver is yawning while driving a car"	0.56	0.42	0.48

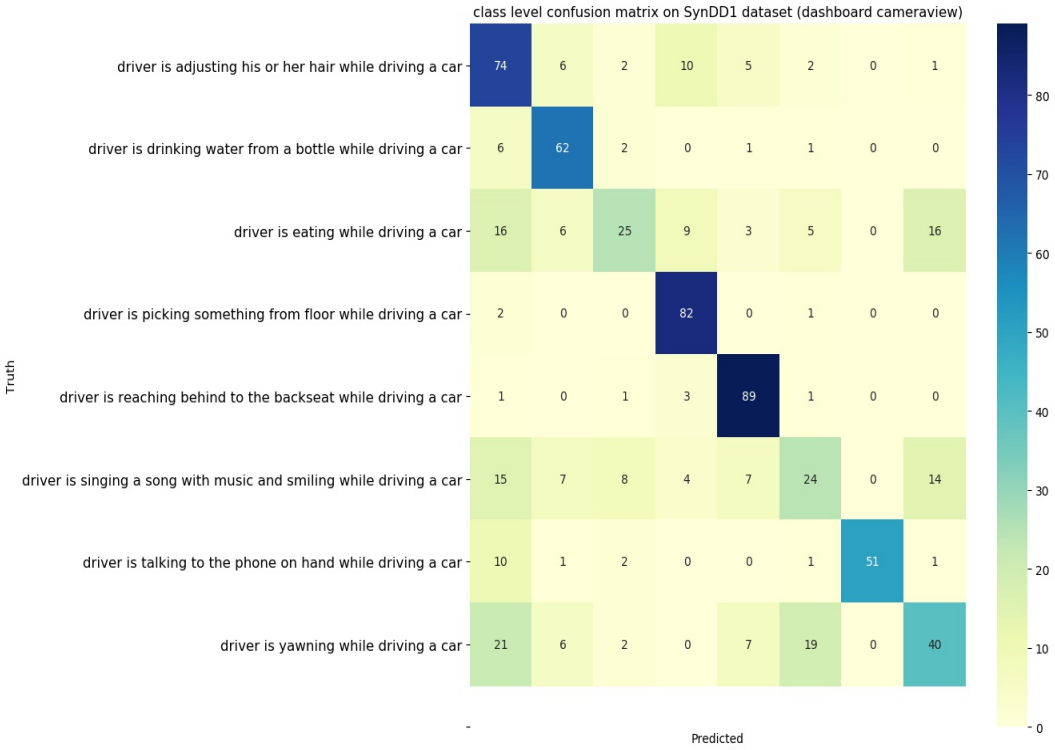

Figure 2: Class-level confusion matrix on SynDD1 dataset (dashboard camera view).