Supplementary Material: Ordered Atomic Activity for Fine-grained Interactive Traffic Scenario Understanding

Nakul Agarwal[1] Yi-Ting Chen[2]
[1]Honda Research Institute USA [2]National Yang Ming Chiao Tung University
nakul_agarwal@honda-ri.com ychen@cs.nycu.edu.tw

1. OATS Dataset

1.1. Data Collection and IRB Approval

We collect data on the road using an equipped vehicle in the San Francisco Bay Area region, and therefore human subjects (i.e. pedestrians) are part of our dataset. We do not have IRB approval but will anonymize any identifying information of humans and vehicles (e.g. blur license plates and faces) before publicly releasing the dataset, as has been done in the past with other similar datasets.

1.2. Dataset Statistics

We show detailed statistics of OATS in Table 1 based on terminology described in Figure 1. We show the number of activities, individual actors, actions and also the types of scenarios. Note that around 90% of our scenarios contain four stop signs at the intersection (4W4S), i.e. no traffic light, since stop signs make the scene more interactive between traffic participants.

1.3. Beyond 4-Way Intersections

Our proposed traffic language is not just limited to 4-way intersections and can be easily extended to other road topologies as shown in Figure 2. The ego vehicle is always in Z1 (shown by red star) and subsequent zones (shown by orange star) and corners are marked in an anti-clockwise fashion w.r.t. the ego vehicle. Due to this, our proposed traffic language can easily handle other road topologies. We don't use other road topologies in our dataset as i) they are not as abundantly available as 4-way intersections and ii) even when they are available, they lack rich interactions between traffic agents and ego vehicle in the scene.

1.4. Beyond Crossing Actions

While we focus on crossing scenes in this work, our language can also be extended to represent other important scenarios by further discretizing the road topology. For example, lane changing of ego-vehicle can be represented as A-B:E where A and B are neighboring lanes. An extension of this, A-B-A:E, can represent an ego-vehicle going into an oncoming lane to go around a blockage. U-turns can be represented by Zx-Zx:C, and jaywalking is just pedestrian crossing at a non-intersection, so the road topology would change and not the annotation format. We don't claim that our proposed language is a cure-all for traffic scene understanding, but it is the first step in this direction, and we hope it will stimulate the community to advance this field.

2. Implementation Details

Our appearance and motion models are inspired by [4][1] and [3, 5][2][3] respectively. We adopt stochastic gradient descent with ADAM to learn the network parameters and train the model for 50 epochs using a learning rate ranging from 0.0002 to 0.0001. All feature layers are jointly updated during training. We fix the input resolution to 224×224 and use 32 frames as

[1] https://github.com/wjchaoGit/Group-Activity-Recognition
[2] https://github.com/abduallahmohamed/Social-STGCNN
[3] https://github.com/yysijie/st-gcn

input for all the experiments. We set $N = 20$ for all experiments. Since some of the classes in our dataset do not have enough samples for training, we only train and evaluate on 35 classes with enough samples shown in Table 2. We also do not use ego vehicle classes in our experiments, i.e. action units starting with 'E', as i) our focus in this work is to understand driving scenarios based on activities of other traffic agents and ii) because of the former reason, ego vehicle cannot be effectively represented in our current graph for solving the proposed tasks. For multilabel atomic activity recognition and activity order prediction, we train on two out of the three splits and test on the remaining one, and do this iteratively until each split has been a test set once. For scenario retrieval, we use 13 high frequency classes out of the 35 for experiments: Z3-Z1:C, Z2-Z4:C, Z1-Z3:C, C2-C1:P, C2-C3:P, Z4-Z2:C, C3-C4:P, C3-C2:P, C4-C1:P, C1-C2:P, Z3-Z4:C, C1-C4:P, C4-C3:P. We do this by taking a subset from s1, s2 and s3 comprising only the above 13 classes and forming new splits s1', s2' and s3' respectively. We then treat two out of the three newly formed splits as database and the remaining one as query, and do this iteratively until each split has been a query set once. Since there are only 2 actors, i.e., 'C' (cars) and 'P' (pedestrians) in these 13 classes and 'P' only operate on corners (C1, C2, C3, C4) and 'C' only operate on zones (Z1, Z2, Z3, Z4), we see identical results for actions and activities in Table 6 in the main paper. This is because if the action is retrieved correctly, then the actor will always be retrieved correctly.

3. Quantitative Results

We show classwise results of our method on all three splits of the OATS dataset in Table 2, corresponding to results in Table 2 in the main paper. We also show results on all three splits corresponding to two different fusion methods of motion and appearance features in our network: i) Tracklet level fusion and ii) Average fusion, in Table 3. Since both our appearance and motion-based GCN are formed using tracklets, we have a correspondence of a particular agent across both the GCNs and also across time. Our GCNs give a feature representation of $B \times T \times N \times C$ where B is the batch size, T is the number of frames, N is the number of nodes in the graph and C is the channel dimension.

Tracklet level fusion. For this, we first concatenate features across the channel dimension, do max pooling across the nodes, pass it through a fully connected layer and then take an average across the number of frames.

Average fusion. We first do 2D average pooling of the motion features across the nodes and frames, then concatenate with the appearance features by replicating the same averaged motion features across number of frames. Then, we pass this concatenated feature representation through a fully connected layer and finally take an average across number of frames.

We find *Tracklet level* fusion to perform worse as shown in Table 3 primarily because of inconsistency in individual tracklets. This is one of the major reasons why object level methods [4, 1] do not consider tracklets in the graphs and trajectory prediction and motion-based graph methods [3, 5, 2] simply avoid tracklets which are not present across all frames in the graph. We cannot avoid such tracklets in our dataset because i) we are trying to solve the problem from a video level to ease annotation burden and thus lack ground truth tracklets, ii) such inconsistent tracklets are often the activities represented in the ground truth and iii) our weakly supervised phrase grounding algorithm is based on tracklets.

4. Qualitative Results

We show qualitative results of our method for multilabel atomic activity recognition in Figure 3 and Figure 4, with failure cases in Figure 5. As mentioned as one of the limitations in the main paper, our method is unable to successfully handle group classes like P+, C+ and K+ as we do not explicitly model these in our framework.

References

[1] Fabien Baradel, Natalia Neverova, Christian Wolf, Julien Mille, and Greg Mori. Object Level Visual Reasoning in Videos. In *ECCV*, 2018.

[2] Vineet Kosaraju, Amir Sadeghian, Roberto Martín-Martín, Ian Reid, S Hamid Rezatofighi, and Silvio Savarese. Social-bigat: Multimodal trajectory forecasting using bicycle-gan and graph attention networks. *arXiv preprint arXiv:1907.03395*, 2019.

[3] Abduallah Mohamed, Kun Qian, Mohamed Elhoseiny, and Christian Claudel. Social-stgcnn: A social spatio-temporal graph convolutional neural network for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14424–14432, 2020.

[4] Jianchao Wu, Limin Wang, Li Wang, Jie Guo, and Gangshan Wu. Learning actor relation graphs for group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9964–9974, 2019.

[5] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018.

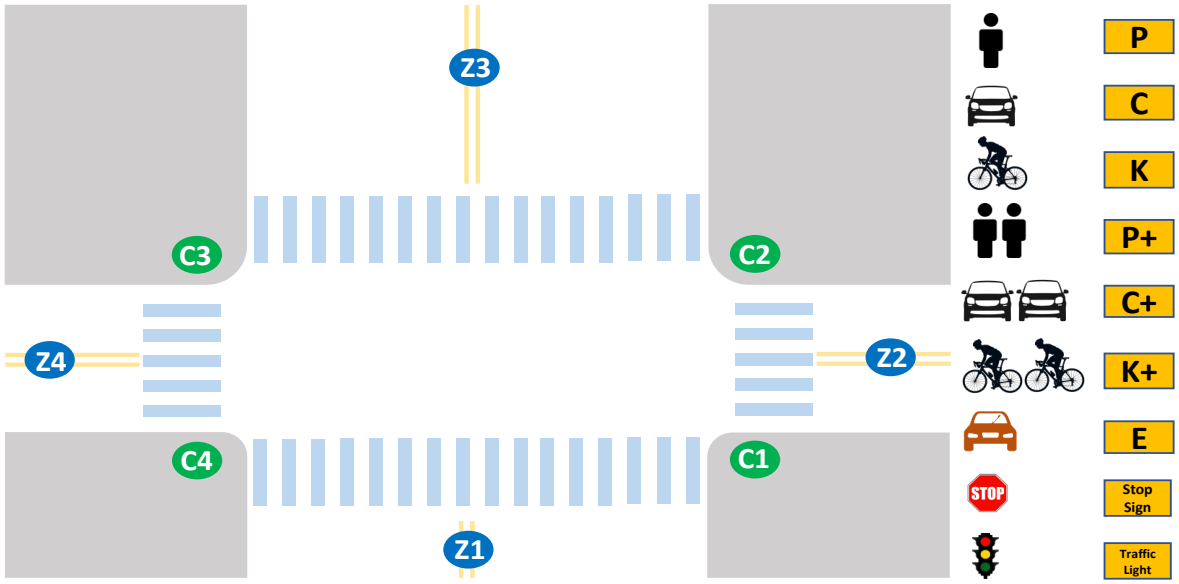

Figure 1: Intersection description and configuration for the OATS dataset.

Table 1: Statistics of the OATS Dataset.

Action Units	#
Z1-Z3:E	648
Z3-Z1:C	560
Z2-Z4:C	495
Z4-Z2:C	466
Z1-Z3:C	347
Z1-Z2:E	258
Z1-Z4:E	331
Z3-Z4:C	221
Z4-Z3:C	207
C2-C3:P	198
C3-C2:P	188
C2-C1:P	181
C3-C4:P	167
C1-C2:P	166
Z2-Z1:C	161
C1-C4:P	160
C4-C1:P	156
Z2-Z3:C	153
Z3-Z2:C	140
C4-C3:P	128
Z1-Z4:C	123
Z4-Z1:C	112
C2-C1:P+	106
C1-C4:P+	104
C3-C4:P+	104
C2-C3:P+	98
Z1-Z2:C	97
C3-C2:P+	96
C1-C2:P+	89
C4-C3:P+	85

Action Units	#
C4-C1:P+	47
Z3-Z1:C+	35
Z2-Z4:K	31
Z3-Z1:K	30
Z2-Z4:C+	23
Z4-Z2:K	23
Z1-Z3:K	21
Z4-Z2:C+	15
Z1-Z3:C+	14
Z3-Z2:K	11
Z4-Z3:K	7
Z1-Z2:C+	4
Z3-Z4:K	4
Z3-Z4:C+	4
Z2-Z3:C+	4
Z1-Z2:K	4
Z2-Z1:C+	3
Z2-Z3:K+	3
Z4-Z3:C+	3
Z2-Z3:K	2
Z3-Z2:C+	2
Z4-Z1:K	2
Z1-Z4:K	2
Z3-Z1:K+	2
Z2-Z4:K+	2
Z1-Z2:K+	1
Z4-Z1:C+	1
Z1-Z4:C+	1
Z4-Z1:K+	1

Motion (Verbs)	#
Z1-Z3	1030
Z3-Z1	627
Z2-Z4	551
Z4-Z2	504
Z1-Z4	457
Z1-Z2	364
C2-C3	296
C2-C1	287
C3-C2	284
C3-C4	271
C1-C4	264
C1-C2	255
Z3-Z4	229
Z4-Z3	217
C4-C3	213
C4-C1	203
Z2-Z1	164
Z2-Z3	162
Z3-Z2	153
Z4-Z1	116

Agent Types (Nouns)	#
P	1344
C	3082
K	137
P+	729
C+	109
K+	9
E	1243

Type of Scenario	#
4W4S	989
4WUT	90
4WPT	14
4W2S	9

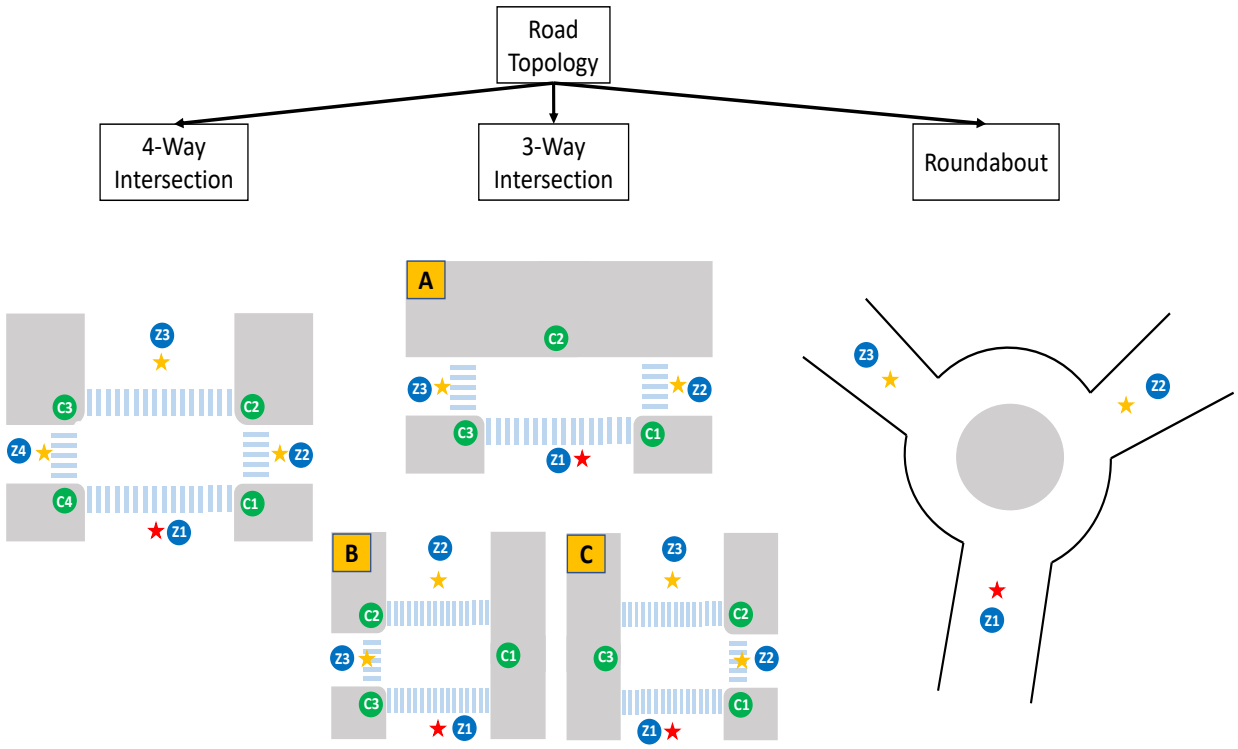

Figure 2: **Beyond 4-way intersections.** This figure depicts how our proposed traffic language can be extended to diverse road topologies such as 4-way intersections (left), 3-way intersections which can have three possible configurations, i.e. A, B, C (center), and roundabouts (right). The ego-vehicle is always in Z1 (represented by red star) and subsequent zones (shown by orange star) and corners are marked in an anti-clockwise fashion w.r.t. the ego vehicle.

Table 2: Classwise results of our method on all three splits of the OATS dataset.

Classes	Splits s1	s2	s3
Z1-Z3:C+	2.64	6.11	4.51
C1-C2:P+	28.48	35.06	22.20
C2-C1:P	19.43	24.11	20.79
Z3-Z1:C	51.17	67.64	48.07
Z2-Z1:C	23.94	66.90	57.17
C4-C3:P	22.53	17.15	25.08
Z1-Z3:C	51.84	74.14	70.71
Z1-Z2:C	15.83	18.89	11.64
Z2-Z4:C	49.82	59.49	62.72
Z4-Z2:C	52.83	72.01	71.10
Z3-Z4:C	21.51	25.04	24.63
C2-C3:P	26.16	31.67	32.92
Z4-Z1:C	26.30	17.09	17.14
C3-C4:P	24.89	33.89	25.39
Z1-Z4:C	16.80	23.93	14.82
Z2-Z3:C	14.96	16.40	14.26
C1-C2:P	20.53	24.34	28.95
C2-C3:P+	17.08	19.27	24.23
C3-C2:P	24.45	35.87	29.56
C3-C4:P+	15.83	16.75	20.86
C4-C3:P+	19.23	19.01	16.72
C1-C4:P	44.81	30.58	38.25
C3-C2:P+	8.69	22.17	16.16
Z3-Z2:C	15.48	20.06	15.76
Z3-Z1:K	8.14	1.20	2.19
Z4-Z3:C	21.69	18.87	18.89
C1-C4:P+	51.61	55.27	41.86
Z4-Z2:C+	8.28	1.22	0.60
C4-C1:P+	30.52	37.94	27.22
C2-C1:P+	17.11	32.29	21.21
Z3-Z1:C+	51.92	17.96	25.11
C4-C1:P	29.24	29.14	34.11
Z2-Z4:K	1.10	1.99	5.93
Z2-Z4:C+	7.56	1.80	41.64
Z1-Z3:K	6.43	2.19	0.54
mAP	24.34	28.56	27.21

Table 3: Multilabel atomic activity recognition results corresponding to two different fusion methods of motion and appearance features in our network.

Method	Splits s1	s2	s3	mAP
Tracklet level	17.17	20.05	18.05	18.42
Average	24.34	28.56	27.21	26.70

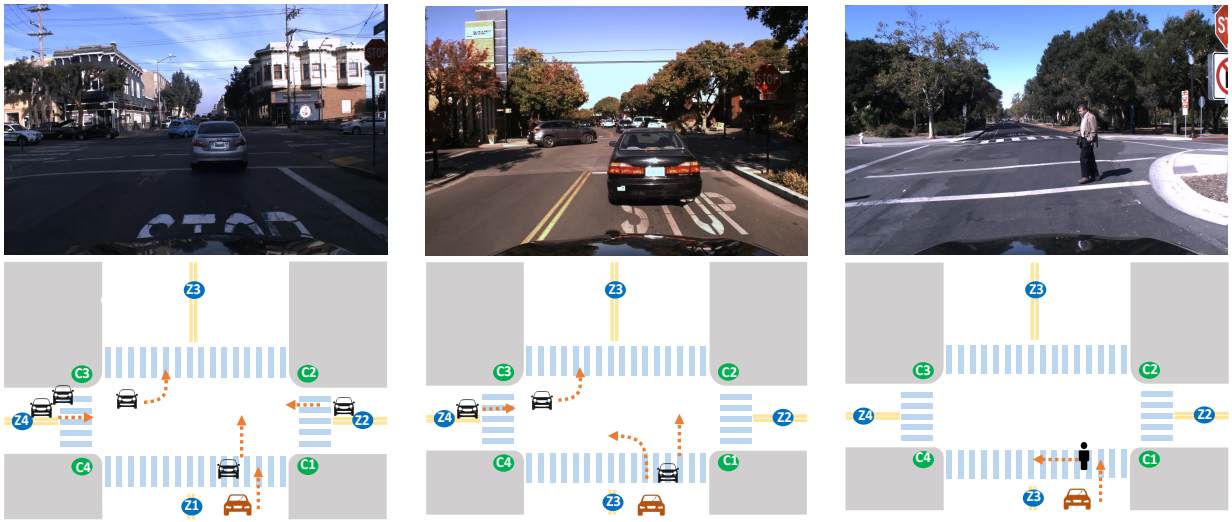

Figure 3: Qualitative results for multilabel atomic activity recognition of our method against ARG [4]. All ground truths contain ego vehicle action, i.e. activities starting with 'E' just for reference and it is not used for classification. The GT denotes ground truth, and green and red color denote true and false positives respectively.

Figure 4: Qualitative results for multilabel atomic activity recognition of our method against ARG [4]. All ground truths contain ego vehicle action, i.e. activities starting with 'E' just for reference and it is not used for classification. The GT denotes ground truth, and green and red color denote true and false positives respectively.

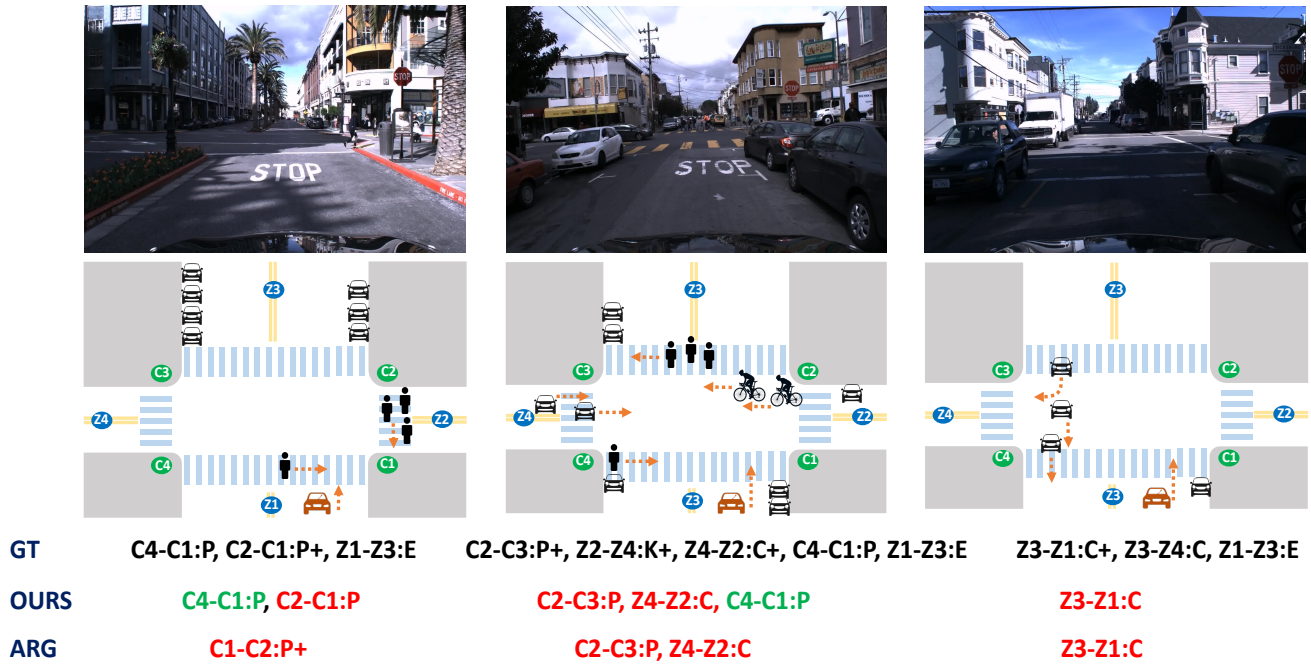

Figure 5: **Failure Cases.** Qualitative results representing failure cases for multilabel atomic activity recognition of our method and ARG [4]. All ground truths contain ego vehicle action, i.e. activities starting with 'E' just for reference and it is not used for classification. The GT denotes ground truth, and green and red color denote true and false positives respectively.