2024 Data and Evaluation Method (Beta)

To participate, please fill out this online AI City Challenge Datasets Request Form.

Data Download Links

Track 1: Multi-Camera People Tracking

- Track1-Data-Download (by clicking on the link, you are accepting this particular data license agreement)

Track 2: Traffic Safety Description and Analysis

- Track2-Data-Download (by clicking on the link, you are accepting this particular data license agreement)

Track 3: Naturalistic Driving Action Recognition

- Track3-Data-Download (by clicking on the link, you are accepting this particular data license agreement)

Track 4: Road Object Detection in Fish-Eye Cameras

- Track4-Data-Download (by clicking on the link, you are accepting this particular data license agreement)

Track 5: Detecting Violation of Helmet Rule for Motorcyclists

- Track5-Data-Download (by clicking on the link, you are accepting this particular data license agreement)

Evaluation and Submission

Frame Extraction

Submissions for some tracks will require frame IDs for frames that contain information of interest. In order to ensure frame IDs are consistent across teams, we suggest that all teams use the FFmpeg library (https://www.ffmpeg.org/) to extract/count frames.

Track 1: Multi-Camera People Tracking

Challenge Track 1 involves synthetic animated people data in multiple indoor settings generated using the NVIDIA Omniverse Platform. All feeds are high resolution 1080p feeds at 30 frames per second.

- **Task**

Teams should detect and track targets across multiple cameras.

- **Submission Format**

One text file should be submitted containing, on each line, details of a detected and tracked person, in the following format. Values are space-delimited.

⟨camera_id⟩ ⟨obj_id⟩ ⟨frame_id⟩ ⟨xmin⟩ ⟨ymin⟩ ⟨width⟩ ⟨height⟩ ⟨xworld⟩ ⟨yworld⟩

  - ⟨camera_id⟩ is the camera numeric identifier.
  - ⟨obj_id⟩ is a numeric identifier for each object. It should be a positive integer and consistent for each object identity across multiple cameras.
  - ⟨frame_id⟩ represents the frame count for the current frame in the current video, starting with 0.
  - The axis-aligned rectangular bounding box of the detected object is denoted by its pixel-valued coordinates within the image canvas, ⟨xmin⟩ ⟨ymin⟩ ⟨width⟩ ⟨height⟩, computed from the top-left corner of the image. All values are integers. They are not currently used in the evaluation.
  - ⟨xworld⟩ ⟨yworld⟩ are the global coordinates of the projected bottom points of each object, based on the provided camera matrices for each scene. They are used for evaluation using the HOTA metric.

The text file containing all predictions should be named track1.txt and can be archived using Zip (track1.zip) or tar+gz (track1.tar.gz) to reduce upload time.

- **Evaluation**

Higher Order Tracking Accuracy (HOTA) [1] is an evaluation metric for multi-object tracking that addresses the limitations of previous metrics like MOTA [2] and IDF1 [3]. It integrates three key aspects of MOT: accurate detection, association, and localization into a unified metric. This comprehensive approach balances the importance of detecting each object (detection), correctly identifying objects across different frames (association), and accurately localizing objects in each frame (localization). Unlike some older metrics that may overemphasize one aspect over the others, HOTA provides a more balanced evaluation by considering all these aspects simultaneously. Furthermore, HOTA can be decomposed into simpler components, allowing for detailed analysis of different aspects of tracking behavior. This dual approach of a unified metric and its decomposable components caters both to end-users, who need a straightforward way to compare trackers, and to researchers, who require detailed information to refine their algorithms.

In this challenge track, teams will be ranked on the leaderboard based on their HOTA scores calculated using 3D distance measurements in a multi-camera setting. A notable feature of this year's challenge is the emphasis on online tracking methodologies. These methods rely exclusively on data from previous frames to predict outcomes in the current frame. To encourage this approach, submissions that employ online tracking will be granted a 10% bonus to their HOTA scores. However, it's important to note that this bonus will only be considered when determining the challenge winner and runner-up, and will not be reflected on the leaderboard itself. For instance, consider two teams: Team A and Team B. Team A has a leaderboard HOTA score of 69%, while Team B scores 61%. Upon reviewing their submitted papers and verifying their open-source code, it's established that Team A used an offline method, whereas Team B utilized an online method. For the purpose of deciding the winner and runner-up, Team B's score is effectively adjusted to 71% HOTA, surpassing Team A, due to their use of the online tracking approach.

References

[1] Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., & Leibe, B. (2021). Hota: A higher order metric for evaluating multi-object tracking. International journal of computer vision, 129, 548-578.

[2] Bernardin, K., & Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: the clear mot metrics. EURASIP Journal on Image and Video Processing, 2008, 1-10.

[3] Ristani, E., Solera, F., Zou, R., Cucchiara, R., & Tomasi, C. (2016, October). Performance measures and a data set for multi-target, multi-camera tracking. In European conference on computer vision (pp. 17-35). Cham: Springer International Publishing.

Track 2: Traffic Safety Description and Analysis

This task revolves around the long fine-grained video captioning of traffic safety scenarios, especially those involving pedestrian accidents. Leveraging multiple cameras and viewpoints, participants will be challenged to describe the continuous moment before the incidents, as well as the normal scene, captioning all pertinent details regarding the surrounding context, attention, location, and behavior of the pedestrian and vehicle. This task provides a new dataset WTS, featuring staged accidents with stunt drivers and pedestrians in a controlled environment, and offers a unique opportunity for detailed analysis in traffic safety scenarios. The analysis result could be valuable for wide usage across industry and society, e.g., it could lead to the streamlining of the inspection process in insurance cases and contribute to the prevention of pedestrian accidents. More features of the dataset can be referred to the dataset homepage.

- Data

The train and validation dataset contains 810 videos and 155 scenarios, each scenario has ~5 segments annotated for capturing detailed behavior changes from pre-recognition, recognition, judgment, action, and avoidance. Each segment has 2 long detailed captions generated from a manual checklist with 170+ items regarding traffic scenarios about pedestrians and vehicles respectively with an average caption length of about 58.7 words. Target pedestrian and vehicle Bounding Box are provided as instance information that the caption related. All videos are provided in a high 1080p resolution at 30 fps. Moreover, this task dataset also provides long fine-grained caption annotations obtained using the same annotation manner for around 3.4K pedestrian-related traffic videos selected from BDD100K, which are offered to be used as part of the external train and validation dataset for generalized performance check.

The description focuses on [location][attention][behavior][context] information about the pedestrian and vehicle respectively, especially about the moments as short segments before the staged accidents along the time directions, as well as normal cases.

The ground truth file contains caption and target instance BBox information. BBox is annotated manually for the first frame of each segment and uses the video object tracking method for generating the left frames BBox.

Caption annotation format is defined as:

```
{
    "id": 722, ## UUID
    "overhead_videos": [  ## caption related videos
        "20230707_8_SN46_T1_Camera1_0.mp4",
        "20230707_8_SN46_T1_Camera2_1.mp4",
        "20230707_8_SN46_T1_Camera2_2.mp4",
        "20230707_8_SN46_T1_Camera3_3.mp4"
    ],
    "event_phase": [
        {
            "labels": [
                "4"  ## segment number
            ],
            "caption_pedestrian": "The pedestrian stands still on the left,
            "caption_vehicle": "The vehicle was positioned diagonally to ..
            "start_time": "39.395",  ## start time of the segment in second
            "end_time": "44.663"     ## end time of the segment in seconds
        },
...
```

BBox format follows COCO format, we provided a frame extraction script here (https://github.com/woven-visionai/wts-dataset/tree/main?tab=readme-ov-file#data-preparation) to reproduce the frame ID for associating with our annotation .

```
{
    "annotations": [
        {
            "image_id": 904, ## frame ID
            "bbox": [
                1004.4933333333333, ## x_min
                163.28666666666666, ## y_min
                12.946666666666667, ## width
                11.713333333333333  ## height
            ],
            "auto-generated": false,  ## human annotated frame
            "phase_number": "0"  ## segment index
        },
        {
            "image_id": 905,
```

```
            "bbox": [
                1007.1933333333333,
                162.20666666666668,
                12.946666666666667,
                11.713333333333333
            ],
            "auto-generated": true,  ##generated bbox annotation for the fr
            "phase_number": "0"
        },
...
```

- **Task**

Teams in this challenge will provide two captions about pedestrians and vehicles for each segment in the traffic events in the video involving the accidents and normal scenes. The performance will be evaluated across multiple metrics that measure the fidelity of the predicted description against the ground truth.

- **Submission Format**

The test results are required to be provided per scenario. Users could use the multi-view videos in the same scenario folders for validation purposes, as well as multi-view videos in train for training purposes. For the normal scenarios in “normal_trimmed” folder and “BDD_PC_5K” part, every single video in the test set is required to provide the caption results.

```
{
    "20230707_12_SN17_T1": [  ##scenario index
        {
            "labels": [  ## segment number, this is known information will
                "4"
            ],
            "caption_pedestrian": "",  ## caption regarding pedestrian
            "caption_vehicle": ""      ## caption regarding vehicle
        },
        {
            "labels": [
                "3"
            ],
            "caption_pedestrian": "",
```

```
            "caption_vehicle": ""
        },
        {
            "labels": [
                "2"
            ],
            "caption_pedestrian": "",
            "caption_vehicle": ""
        },
        {
            "labels": [
                "1"
            ],
            "caption_pedestrian": "",
            "caption_vehicle": ""
        },
        {
            "labels": [
                "0"
            ],
            "caption_pedestrian": "",
            "caption_vehicle:  ""
        }
    ]
}
```

Notice that, unlike the training data, the segment label and its timestamp are not required for the test data submission.

- **Evaluation**

Data from “BDD_PC_5K” are available to be used as a train part, the user could use it directly into the training or utilize it as pre-train as well. Test includes the “BDD_PC_5K” test part as well as the generalization performance test. The metric used to rank the performance of each team will be an averaged accuracy to compare the predicted descriptions against the ground truth with multiple metrics across all scenarios including the video in “normal_trimmed” folder in the staged WTS dataset and “BDD_PC_5K” part. 4 metrics for being averaged are BLUE-4, METERO, ROUGE-L, and CIDEr.

Please pay attention that this is not a retrieval task and we are seeking for generative solutions. Teams may submit results to evaluation system and rank on the leaderboard with any method. But we will be manually evaluating award contenders and teams only using retrieval method will be disqualified from winning the awards. For example, considering a method which uses features extracted from the test set videos and retrieves the "closest-meaning" caption from the training set for submission, this will not be qualified for winning the track since it is not a generative solution.

More details about the dataset explanation can be referred to: https://github.com/woven-visionai/wts-dataset

Track 3: Naturalistic Driving Action Recognition

Distracted driving is highly dangerous and is reported to kill about 8 people every day in the United States. Today, naturalistic driving studies and computer vision techniques provide the much-needed solution to identify and eliminate distracted driving behavior on the road. In this challenge track, users will be presented with synthetic naturalistic data of the driver collected from three camera locations inside the vehicle (while the driver is pretending to be driving). The objective is to identify the start time, end time and type of distracted behavior activities executed by the driver in each video. Participating teams will have the option to use any one or two or all three camera views for the classification of driver tasks. Teams will be provided with training data with labels to develop algorithms and should submit functional code that can be executed on a reserved testing data set. The final winner will be determined by the performance on this reserved testing data set.

- **Data**

The data set contains 594 video clips (about 90 hours in total) captured from 99 drivers. Drivers do every one of the 16 different tasks (such as talking on the phone, eating, and reaching back) once, in random order. There are three cameras mounted in the car, recording from different angles in synchronization. Each driver performs the data collection tasks twice: in one go with no appearance block and in another go with some appearance block (e.g., sunglasses, hat). Thus, there are 6 videos collected for each driver, 3 videos in sync with no appearance block and 3 videos in sync with some appearance block, resulting in 594 videos in total.

The 90 hours of videos (99 drivers total) in this track are split into three data sets A1, A2 and B, each containing 69, 15, and 15 drivers respectively. Teams will be provided dataset A1 with the ground truth labels of start time, end time and type of distracted behaviors (manually annotated), and dataset A2 with no labels. Teams can use both A1 and A2 to develop their algorithms and

submit results for data set A2 to our online evaluation server to be represented on the public leader board for performance tracking. The public leader board only provides a way for a team to evaluate and improve their systems and the ranking will NOT determine the winners of this track.

Dataset B is reserved for later testing. Top performers on the public ranking board will be invited to submit functional code for both training and inference for the problem. Organizers will test the submitted code against dataset B and the final winner will be determined on the performance against dataset B. Teams wishing to be considered for evaluation on dataset B must also make their training and inference codes publicly available.

- **Submission Format**

To be ranked on the public leader board of data set A2, one text file should be submitted to the online evaluation system containing, on each line, details of one identified activity, in the following format (values are space-delimited):

⟨video_id⟩ ⟨activity_id⟩ ⟨start_time⟩ ⟨end_time⟩

Where:

  - ⟨video_id⟩ is the video numeric identifier, starting with 1. Video IDs have been provided in the video_ids.csv file in the data download link.
  - ⟨activity_id⟩ is the activity numeric identifier, starting with 1 (Class 0- Normal Forward Driving, is not considered for evaluation).
  - ⟨start_time⟩ is the time the identified activity starts, in seconds. The start_time is an integer value, e.g., 127 represents the 127th second of the video, 02:07 into the video.
  - ⟨end_time⟩ is the time the identified activity ends, in seconds. The end_time is an integer value, e.g., 263 represents the 263rd second of the video, 04:23 into the video.

- **Evaluation**

Evaluation for track 3 will be based on model activity identification performance, measured by the average activity overlap score, which is defined as follows. Given a ground-truth activity *g* with start time *gs* and end time *ge*, we will find its closest predicted activity match as that predicted activity *p* of the same class as *g* and highest overlap score *os*, with the added condition that start time *ps* and end time *pe* are in the range [*gs* – 10s, *gs* + 10s] and [ge – 10s, ge + 10s], respectively. The overlap between *g* and *p* is defined as the ratio between the time intersection and the time union of the two activities, i.e.,

$$os(p,g) = \frac{\max\left(\min(ge,pe) - \max(gs,ps), 0\right)}{\max(ge,pe) - \min\left(gs,ps\right)}$$

After matching each ground truth activity in order of their start times, all unmatched ground truth activities and all unmatched predicted activities will receive an overlap score of 0. The final score is the average overlap score among all matched and unmatched activities.

Track 4: Road Object Detection in Fish-Eye Cameras

Fisheye lenses have gained popularity owing to their natural, wide, and omnidirectional coverage, which traditional cameras with narrow fields of view (FoV) cannot achieve. In traffic monitoring systems, fisheye cameras are advantageous as they effectively reduce the number of cameras required to cover broader views of streets and intersections. Despite these benefits, fisheye cameras present distorted views that necessitate a non-trivial design for image undistortion and unwarping or a dedicated design for handling distortions during processing. It is worth noting that, to the best of our knowledge, there is no open dataset available for fisheye road object detection for traffic surveillance applications. The datasets (FishEye8K and FishEye1Keval) comprises different traffic patterns and conditions, including urban highways, road intersections, various illumination, and viewing angles of the five road object classes in various scales.

- **Data**

We use the FishEye8K benchmark dataset, which is published in CVPRW23, as a training set (the train set of the FishEye8K) and a validation set (the test set of the FishEye8K). The training set has 5288 images, and the validation set has 2712 images with resolutions of 1080×1080 and 1280×1280. The two sets have a total of 157K annotated bounding boxes of 5 road object classes (Bus, Bike, Car, Pedestrian, Truck). Dataset labels are available in three different formats: XML (PASCAL VOC), JSON (COCO), and TXT (YOLO).

The test dataset FishEye1Keval will contain 1000 images similar to the training dataset images; however, extracted from 11 camera videos which were not utilized in the making of FishEye8K dataset. The test dataset will be released later.

- **Submission Format**

One JSON file should be submitted, containing each dictionary, details of a detected object, and corresponding class ID and bounding box. The submission format schema to be followed is as follows:

```
[
  {
    "image_id": ,
    "category_id": ,
    "bbox": [x1, y1, width, height],
    "score":
  },
  // Add more detections as needed
]
```

How to create an image ID?

```
def get_image_Id(img_name):
  img_name = img_name.split('.png')[0]
  sceneList = ['M', 'A', 'E', 'N']
  cameraIndx = int(img_name.split('_')[0].split('camera')[1])
  sceneIndx = sceneList.index(img_name.split('_')[1])
  frameIndx = int(img_name.split('_')[2])
  imageId = int(str(cameraIndx)+str(sceneIndx)+str(frameIndx))
  return imageId
```

Above shows basic code that converts image name to image ID, which is useful when you create the submission file.

For example:

- If image name is "camera1_A_12.png" then cameraIndx is 1, sceneIndx is 1, frameIndx is 12 and imageId is 1112.
- If image name is "camera5_N_431.png" then cameraIndx is 5, sceneIndx is 3, frameIndx is 431 and imageId is 53431.

Category IDs:

The numbering of category IDs are as follows:

Class names	Bus	Bike	Car	Pedestrian	Truck
Category ID	0	1	2	3	4

- **Evaluation Metric**

To ensure a fair and comprehensive evaluation process, we will use the F1-score (harmonic mean of total precision and recall (not cumulative, that is used in plotting the PR curve) as the main metric for Track 4. For more details please visit track4 data download page.

- **Additional Datasets**

Teams aiming for the public leaderboard and challenge awards must not use non-public datasets in any training, validation, or test sets. Winners and runners-up must submit their training and testing codes for verification after the deadline to confirm no non-public data was used, ensuring tasks were algorithm-driven, not human-performed.

Track 5: Detecting Violation of Helmet Rule for Motorcyclists

Motorcycles are one of the most popular modes of transportation, particularly in developing countries such as India. Due to lesser protection compared to cars and other standard vehicles, motorcycle riders are exposed to a greater risk of crashes. Therefore, wearing helmets for motorcycle riders is mandatory as per traffic rules and automatic detection of motorcyclists without helmets is one of the critical tasks to enforce strict regulatory traffic safety measures.

- **Data**

The training dataset contains 100 videos and groundtruth bounding boxes of motorcycle and motorcycle rider(s) with or without helmets. Each video is 20 seconds duration, recorded at 10 fps. The video resolution is 1920×1080.

Each motorcycle in the annotated frame has bounding box annotation of each rider with or without helmet information, for upto a maximum of 4 riders in a motorcycle. The class id (labels) of the object classes in this dataset is as follows:

1, motorbike: bounding box of motorcycle
2, DHelmet: bounding box of the motorcycle driver, if he/she is wearing a helmet
3, DNoHelmet: bounding box of the motorcycle driver, if he/she is not wearing a helmet
4, P1Helmet: bounding box of the passenger 1 of the motorcycle, if he/she is wearing a helmet
5, P1NoHelmet: bounding box of the passenger 1 of the motorcycle, if he/she is not wearing a helmet
6, P2Helmet: bounding box of the passenger 2 of the motorcycle, if he/she is wearing a helmet

7, P2NoHelmet: bounding box of the passenger 2 of the motorcycle, if he/she is not wearing a helmet
8, P0Helmet: bounding box of the child sitting in front of the Driver of the motorcycle, if he/she is wearing a helmet
9, P0NoHelmet: bounding box of the child sitting in front of the Driver of the motorcycle, if he/she is wearing not a helmet

The groundtruth file contain bounding box information (one object instance per line) for each video. The schema is as follows (values are comma-separated):

⟨video_id⟩, ⟨frame⟩, ⟨bb_left⟩, ⟨bb_top⟩, ⟨bb_width⟩, ⟨bb_height⟩, ⟨class⟩

- ⟨video_id⟩ is the video numeric identifier, starting with 1. It represents the position of the video in the list of all videos, sorted in alphanumeric order.
- ⟨frame⟩ represents the frame count for the current frame in the current video, starting with 1.
- ⟨bb_left⟩ is the x-coordinate of the top left point of the bounding box.
- ⟨bb_top⟩ is the y-coordinate of the top left point of the bounding box.
- ⟨bb_width⟩ is the width of the bounding box.
- ⟨bb_height⟩ is the height of the bounding box.
- ⟨class⟩ is the class id of the object as given in the labels information above.

The test dataset will contain 100 videos of 20 seconds each, recorded at 10 fps, similar to the training dataset videos. The test dataset will be released later.

- **Task**

Teams should identify motorcycle and motorcycle rider(s) with or without helmet. Similar to the training dataset, each rider in a motorcycle (i.e., driver, passenger 1, passenger 2) is to be separately identified if they have a helmet or not.

- **Submission Format**

One text file should be submitted, containing on each line, details of a detected object and the corresponding class id (as per the labels information). The submission format schema to be followed is as follows (values are comma separated).

⟨video_id⟩, ⟨frame⟩, ⟨bb_left⟩, ⟨bb_top⟩, ⟨bb_width⟩, ⟨bb_height⟩, ⟨class⟩, ⟨confidence⟩

- ⟨video_id⟩ is the video numeric identifier, starting with 1. It represents the position of the video in the list of all videos, sorted in alphanumeric order.
- ⟨frame⟩ represents the frame count for the current frame in the current video, starting with 1.
- ⟨bb_left⟩ is the x-coordinate of the top left point of the bounding box.
- ⟨bb_top⟩ is the y-coordinate of the top left point of the bounding box.
- ⟨bb_width⟩ is the width of the bounding box.
- ⟨bb_height⟩ is the height of the bounding box.
- ⟨class⟩ is the class id of the object as given in the labels information above.
- ⟨confidence⟩ is the confidence score of the bounding box (values between 0 to 1).

- **Evaluation**

The metric used to rank the performance of each team will be the mean Average Precision (mAP) across all frames in the test videos. mAP measures the mean of average precision (the area under the Precision-Recall curve) over all the object classes, as defined in PASCAL VOC 2012 competition.

Additional Datasets

External dataset or pre-trained models are allowed only if they are public. Teams that wish to be listed in the public leader board and win the challenge awards are NOT allowed to use any private data or private pre-trained model for either training or validation. The winning teams and runners-up are required to submit their training and testing codes for verification after the challenge submission deadline in order to ensure that no private data or private pre-trained model was used for training and the tasks were performed by algorithms and not humans.