Technical description of the HRI RoadTraffic dataset

A.Gepperth

Abstract

In this report, we give detailed technical information about the HRI RoadTraffic dataset that is used in our recent publications ([1, 2]). We also compare the HRI RoadTraffic dataset to other publicly available, annotated datasets and present arguments why the HRI RoadTraffic dataset constitutes a suitable reference dataset for future benchmarking efforts.

1 Overview

The HRI RoadTraffic dataset contains five distinct video streams covering a significant range of traffic, environment and weather conditions. All videos are around 15 minutes in length and were taken during test drives along a fixed route covering mainly inner-city areas, along with short times of highway driving. Please see Tab. 1 for details and Fig. 1 for a visual impression. For the quantitative evaluation of object detection performance, we manually annotated relevant objects in the recorded video streams, please see Fig. 2 for details.

2 Comparison of the HRI RoadTraffic dataset to other vehicle benchmark datasets

There exist, by now, a number of annotated vehicle datasets which are often used for benchmarking the performance of object detection systems in a comparable way. For traffic related areas of interest, the most notable datasets are the CBCL StreetScenes Database (see, e.g., [3]) and the UIUC Image Database for

ID	weather	daytime	single images	annotated images
I	overcast,dry	afternoon	9843	957
II	low sun, dry	late afternoon	22600	949
III	heavy rain	afternoon	6725	643
IV	dry	midnight	6826	464
V	after heavy snow	afternoon	16551	867

Table 1: Details about the individual video streams in the HRI RoadTraffic dataset. Please note that streams II and V were recorded at a frame rate of 20Hz.

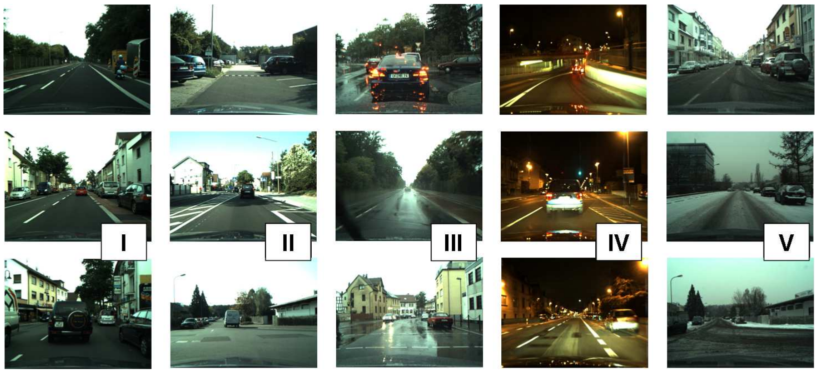

Figure 1: Selected example images from streams I-V. All videos were taken in RGB color using a MatrixVision mvBlueFox camera at a resolution of 800x600. Used frame rates were 10Hz except for video II where a setting of 20Hz was used. Aperture was always set to 4.0 except for video IV where we used a value of 2.4. A self-implemented exposure control was used on both cameras, manipulating the gain and exposure settings of each camera.

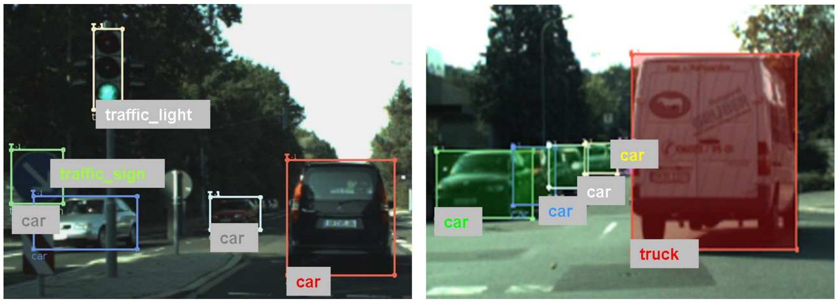

Figure 2: Examples of annotated objects(annotated road area is not shown). Each annotation consists of a polygonal area, an identity and an occlusion value (not shown). In order to reduce the annotation effort, only every tenth image in a video sequence was annotated. We provide positive examples for a number of different object classes. Annotations are exhaustive, i.e., all objects of a certain class present in a given image are covered by an annotation. As can be seen from the images, we use what we term *semantic annotations*, which means that is has been tried to mark the whole area containing an object even if it is partially occluded.

Car Detection[4]. Another popular benchmark are the datasets of the yearly PASCAL object detection challenges, which also contain traffic objects but are not restricted to them.

In contrast to the HRI RoadTraffic dataset described here, these datasets consist of monocular still images instead of continuous stereo video. Apart from the usefulness of stereo information, we believe that the possibilities of processing continuous video streams are manifold, since object detection could be supported by, e.g., tracking algorithms. In addition, the number of annotated images is significantly larger in the HRI RoadTrafic dataset; furthermore, our annotations include information about object occlusion as well as the precise geometric layout of the obstacle-free/drivable road areas, neither of which is contained in the other described datasets. Finally, the HRI road traffic dataset contains (for each image) additional information, such as the results of the free-area computation described in [5], as well as speed/yaw rate information. The reason for including the free-area computation results is, on the one hand, because we want the experiments conducted in [1] to be reproducible, and on the other hand, because it can make sense for processing algorithms to work with less-than-perfect data always encountered in real-world processing (as opposed to using the free-area from the annotations).

Going beyond the area of vehicle detection, there exist several publicly available datasets containing pedestrian annotations which are comparable to HRI RoadTraffic: one of these is the Daimler pedestrian detection benchmark[6]. It contains a large number of cropped pedestrian images for classifier training, and a continuously (i.e., every frame may contain annotations) annotated sequence of approximately 27 minutes of inner-city driving for evaluation purposes. The image resolution is 640x480 recorded with a monocular grayscale camera. In contrast to HRI RoadTraffic, additional data such as free-area and speed/yaw rate information are not included; however the total number of annotated objects is larger than in the HRI RoadTraffic dataset. Furthermore, annotations contain occlusion as well as track information, the latter meaning that it is possible to identify the same annotated pedestrians in consecutive images. In contrast to HRI RoadTraffic, the annotations are not exhaustive, meaning that not all of the pedestrians visible in any given image are annotated.

A further large-scale pedestrian dataset is the CalTech Pedestrian Dataset [7] containing a very large amount (order of magnitude: 100000) of exhaustively annotated pedestrians in approximately 50 video sequences recorded with identical hardware setup in a variety of inner-city settings. Annotations are continuous and include occlusion as well as track information, however in contrast to HRI RoadTraffic, free-area and speed/yaw rate of the ego-vehicle are not available. Another unfavorable point is the fact that only a part of the annotations was performed manually, whereas the remaining annotations were generated by tracking algorithms, resulting in a significant number of corrupted or imprecise annotations. Images are monocular, have a resolution of 640x480 and are available in RGB color.

A very recent addition to the publicly available benchmark pool is the CVC pedestrian dataset [8]. It contains a large amount (order of magnitude: 10000) of (almost) exhaustively annotated pedestrians in several (approx. 15) video sequences of inner-city driving, recorded using identical hardware setup. Annotations are continuous and do not include occlusion, tracking information, free-area or speed/yaw rate. In exchange, pre-computed stereo information is available for every image. Available images are nevertheless monocular and RGB color, having a resolution of 640x480 pixels.

3 Technical description of the HRI RoadTraffic dataset

3.1 High-level description of annotated content

The main feature of the HRI RoadTraffic dataset is the availability of stereo and free-area information as well as high-quality annotations. These annotations do not only contain the positions and identities of cars and vehicles but also of the pedestrians/cyclists (although there are only a few), traffic signs (also few) and, most notably, the obstacle-free area. All annotations contain polygons with an arbitrary number of edges stored in the well-known LabelMe XML file format. Especially for the free area, great care was taken to model this quantity as precisely as possible. Stereo information is pre-computed to save the user the tedious steps of calibration, rectification, matching etc. Right images are available on request in case anyone wants to do this task by hand, please send email requests to hri-road-traffic@honda-ri.de.

Technically, there is a *main directory* containing supplementary information, convenience python code for working with the data, and the streams themselves. In this main directory, there are several additional entries:

cameraCalibration This subdirectory contains a single text file with the camera calibration parameters used for all recordings as well as explanations about the used coordinate system. They are stored human-readable format using the common conventions for camera parameters.

pythonTools This subdirectory contains python code for reading and writing annotation xml files in a very simple fashion. This code depends on no additional packages except xml.dom.minidom which is included by default in standard python distributions.

3.2 Stream details

The actual data, i.e. the streams I-V which are used in [1, 2], comes in 5 subdirectories of the main directory. These subdirectories are, for internal reasons, labelled differently than in [1], namely 017 (stream I), 018 (stream II), 020 (stream IV), 023 (stream III), 033 (stream V). In each stream subdirectory, there are the following entries:

leftImages This subdirectory contains numbered color images from the left camera in PNG format.

timesteps.txt A text file containing one row per image. The first column gives the image index, the second index gives the timestep of the image having that index. This information is important for linking image and CAN information based on timesteps since CAN information was recorded at a different frequency.

rightImages (on request) This subdirectory contains numbered images from the right camera. They are identical in nature to the left camera images described in the previous paragraph.

stereo This subdirectory contains numbered PGM files with pre-computed stereo information. There are 4 different files per image number with the suffixes ”_x”,”_y”, ”_z”, ”_l”. Each of these file indicates carries x/y/z coordinate information as well as validity information (file suffix ”_l”) for each of its points. The coordinate system is defined as described in the previous section. Important: each PGM grayscale image is scaled between 0 and 255. These values must be transformed to actual metric world coordinates by extracting the original minimal and maximal values (which are stored as comments in the PGM header) and subsequent rescaling of the pixel values.

annotations This subdirectory contains XML files that can be read with the Matlab toolbox[1] provided by the LabelMe project[9], or with the python code distributed in the main directory of the HRI RoadTraffic dataset (see below). Both toolboxes allow to access the occlusion value defined for each annotation. XML files can be linked to camera images via their numbers. **Caveat:** label identifiers are not totally consistent, e.g., for the ”vehicle” class, identifiers may be ”vehicle”, ”lorry”, ”car” or similar. The same applies for the free area/drivable area where there are several different label identifiers. Please just do a grep on the XML files or use the provided python code to generate a list of relevant identifiers.

freeArea This subdirectory contains numbered binary PNG images (either having pixel values of 0 or 255) indicating the free area computed from the corresponding left camera image according to the method described in [5]. Each binary image can be linked to images via its number.

proprioception This subdirectory contains a single text file, each line of which contains a timestep, current yaw rate (in degrees/s), current steering wheel angle (in degrees) and current speed in km/h. The correspondence with camera images has to be established via the timestep information.

[1] labelme.csail.mit.edu/LabelMeToolbox/index.html

3.3 General comments

Files in the subdirectories leftImages, rightImages, annotations, stereo and freeArea have a unique consecutive number which is used to link images to each other through different subdirectories. Indices can be mapped to physical timesteps by evaluation of the timesteps text file in each directory. Proprioceptive data were recorded independently at different frequency. They can be matched to image data with the knowledge that 1 second corresponds to 32000 timesteps.

References

[1] A Gepperth, S Hasler, S Rebhan, and J Fritsch. Biased competition in visual processing hierarchies: a learning approach using multiple cues. *Cognitive Computation*, 2010.

[2] T Kühnl, F Kummert, and J Fritsch. Monocular road segmentation using slow feature analysis. In *IEEE Symposium on Intelligent Vehicles*, 2011. to appear.

[3] L Wolf and SM Bileschi. A critical view of context. *IJCV*, 69(2):251–261, August 2006.

[4] S Agarwal, A Awan, and D Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(11), 2004.

[5] T Michalke, R Kastner, J Fritsch, and C Goerick. A generic temporal integration approach for enhancing feature-based road-detection systems. In *Intelligent Transportation Systems Conference*, Peking, 2008. IEEE Press.

[6] M Enzweiler and DM Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2008.

[7] P Dollar, C Wojek, B Schiele, and P Perona. Pedestrian detection: A benchmark. In *CVPR*, June 2009.

[8] D Geronimo, AM Lopez, AD Sappa, and T Graf. Survey of pedestrian detection for advanced driver assistance. *IEEE Transaction on Pattern Recognition and Machine Intelligence*, 32(7), 2010.

[9] BC Russell, A Torralba, KP Murphy, and WT Freeman. Labelme: a database and web-based tool for image annotation. *International Journal of Computer Vision*, 77(1-3), 2008.