Vehicle Ground-truth Database for the Vertical-view Ft. Hood Imagery

Gang Liu and Robert M. Haralick
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500, U.S.A.
{gliu,haralick}@isl.ee.washington.edu

Abstract

This paper reports the work on building the ground-truth databases for the vehicles in the vertical-view Ft. Hood (VVFH) image data set. We briefly describe the protocols followed in manual annotation of the images, how the ground-truth information is inferred from the annotated image data, and the major entities in the ground-truth database. The vehicle detection performance of a few algorithms is evaluated using the data set. The entire data set of images and ground-truth is available on the Internet.

1. Introduction

Automatic vehicle counting in aerial images is a highly desirable feature of aerial image understanding and exploitation systems. A ground-truthed data set is very valuable in developing and evaluating the performance of such algorithms. For example, it facilitates the identification of key features to be used for vehicle detection, and makes it very easy to construct training data sets used in training a Bayesian classifier or a neural network. Also, objective and quantitative performance measures can be obtained by comparing the algorithm output with the ground-truth. This is absolutely important in measuring the improvement in performance and finding out what should be done to further improve the algorithm performance.

The second set of image data [1] taken of Ft. Hood, TX, for the RADIUS program contains 26 aerial images [9]. Of these, 7 images were taken with the vertical viewing angle. We are only concerned with these seven images in this document. There are many vehicles, mostly civilian passenger cars and utility trucks, in the parking lots and on the roads in these images. The ground-truth for these vehicles was manually created.

Human operators delineated the set of pixel locations in the image which fall on the boundary of some vehicle. Ground-truth information about the vehicle shape, orientation and other related entities was inferred from the manually delineated vehicle boundary. A software package was developed to organize and access the extracted information in database files ready for use in algorithm development and performance evaluation.

This paper describes the protocol followed in obtaining the ground-truth information about the vehicle related entities in the imagery. These include the delineated vehicle boundary and various entities derived from the that, the measurements computed and statistics gathered for these entities, and the organization of these information in database files. As an example of using the ground-truth information, we very briefly compare the performance of several vehicle detection algorithms.

2. Ground-truth creation

2.1. Manual annotation of vehicle boundaries

The ground area covered by the seven original images contains parking lots, roads, buildings and natural backgrounds. The image sizes are around 7700×7700 pixels. They are stored as 8-bit raw image files. The dimension of a pixel in the image corresponds to the length of 0.31 meters on the ground. Most vehicles in the image are around 5 to 30 pixels in length and around 3 to 10 pixels in width.

The huge size of these images imposes severe inconvenience in carrying out experiments for algorithm development and performance evaluation. Besides, for each image, there are vast areas which contain no vehicle at all. Since what we want to create here is a ground-truth description of the vehicles in the image, those parts containing no vehicles are not of interest. To facilitate the ground-truth creation and the experiments on vehicle detection algorithm development, for each image we extracted a number of non-overlapping rectangular sub-images, each containing some vehicles. The sub-images are typically of size 1024×1024. There are a total of 337 such sub-images covering a total area of about 214 million pixels. Together the sub-images

[1] `http://www.mbvlab.wpafb.af.mil/public/sdms/datasets/fthood/`

1051-4651/00 $10.00 @ 2000 IEEE

contain the vast majority of the vehicles in the original images. A total of 72584 vehicles were identified and delineated in the images.

Subsequent specifications of the ground-truth information about the vehicles are given in each of the sub-images. However, should there be the need to obtain the ground-truth description for the seven original images, it is easily done since we know the location of every sub-image in the original image. This need never rose in our research activities.

For each sub-image, human operators used a pointing device to mark the pixels falling on the outer boundary of the vehicles in the images. Since all vehicles in the images appear as compact 2D regions, every vehicle boundary forms a closed contour. Contours were allowed to touch each other and share common boundary pixels, but none of them were allowed to cross or enclose each other. In the case of overlapping vehicles, we ignore the parts being hidden, and hence will have vehicles right next to each other, where they share some common boundary points. In the case where there is shadow from the vehicles, every effort was made by the annotator to guess the locations of the true boundary points of the vehicle and mark it, not the boundary of the shadow. Two segments of such images after annotation are shown in Figure 1.

Only the boundaries of the vehicles in the image, including passenger cars, utility trucks and military vehicles such as tanks and armored personnel carriers, were marked. None of the other entities in the image was marked.

2.2. Labeling image pixels using ground-truth boundaries

All vehicles in the images are compact shaped, and the pixels belonging to one vehicle, which are precisely the pixels located inside the delineated boundary for that vehicle, form a connected component. Thus the labeling of the image pixels according to which vehicle they belong to can be done by a connected component analysis procedure [2]. This gives every image pixel a label which is unique for each vehicle in the image. The label shows to which vehicle the pixel belongs. A single label for the background is used for all pixels not falling on any vehicle.

2.3. Vehicle boundary representation

The free-hand drawn outer boundaries of the delineated vehicles are represented by lists of 2D locations of the boundary points. The list starts from the upper-leftmost boundary point and goes clockwise around the vehicle. Polygon models are fitted to the boundary. This is done by identifying corners (break points) along the boundary. The Ji and Haralick corner detector is used for this purpose [3]. The corners and the list of straight line segments connecting them are the vertices and edges of the fitted polygon. Both the delineated boundaries and the fitted boundaries are stored in the database. Certain measurements are computed for the following entities: the area within the delineated and the fitted boundaries, the centroid of the area, the lengths of the fitted straight line segments, inner angles made by the successive straight line segments at the corners, the gray scale values and gradient within the delineated boundary, on the delineated boundary, and on the fitted straight line segments.

2.4. Gap region between closely parked vehicles

In detecting vehicles parked regularly in parking lots, the space between vehicles often confuses vehicle detection algorithms. In order to improve the algorithm’s ability to distinguish true vehicles from these spaces, some characterization of such regions between closely parked vehicles is needed. Such regions are called gaps and are identified between vehicles which are parked side by side, roughly in parallel and within a short distance from each other. A total of 50225 gap regions were identified. Certain measurements are computed for each gap, including its rectangularity and some gray scale value based statistics.

2.5. Bounding rectangle and rectangularity

Although the human delineated vehicle boundaries are often not rectangular, we expect most vehicles to be roughly rectangular shaped. The bounding rectangle is used for measuring the length, width and orientation of the vehicles. The bounding rectangle of a polygon is defined as the smallest rectangle which entirely contains the polygon. It is in general different from the bounding box whose sides are aligned horizontally and vertically. The rectangularity of the polygon is defined as the ratio of the area of the polygon over the area of its bounding rectangle. It is a value between 0 and 1. Vehicles in vertical view images most likely have rectangularity close to 1. Gaps are expected to have larger variation in the rectangularity.

2.6. Regions of non-interest

Some of the pixel locations in the image are labeled as within vehicle, and some as within gap regions. These pixels as well as some other pixels that are spatially close to them bear important information about the vehicles. These are called pixels of interest. The pixels of interest form the region of interest (ROI) in this situation. All pixels not belonging to the ROI form the region of non-interest. The name region of non-interest is somewhat misleading, since

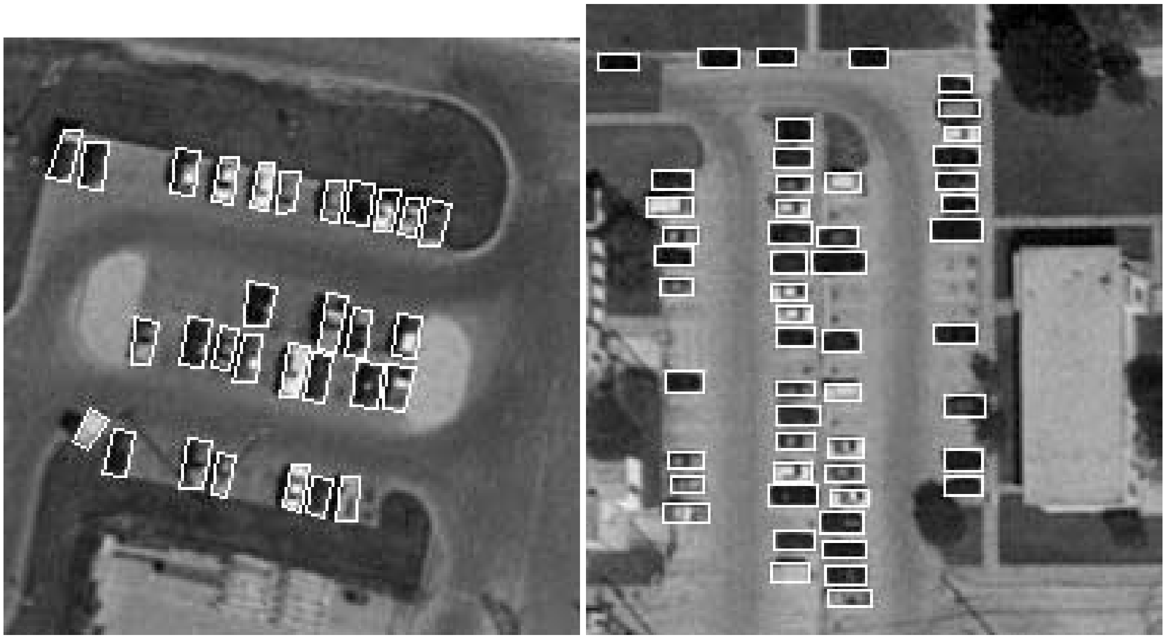

Figure 1. Segments of the annotated image with manually delineated vehicle outer boundaries.

it is actually also of very much interest to developing a vehicle detector — it provides negative training samples.

In creating the database, the core of the ROI consists of pixel locations from the vehicle and gap regions. All pixel locations within 3 pixels from the core are included in the ROI.

2.7. Organization of the database

We construct one database for each of the sub-images. Logically a database is a hierarchical structure. The database contains information about all the identified entities related to the delineated vehicles, including the delineated and fitted boundaries, gaps, area and rectangularity of vehicles and gaps, statistics on gray scale value and gradient value at various parts, etc.

3. Software and data

Software written in the `C` programming language on the Unix platform has been developed for constructing the ground-truth database from the images, and for accessing and manipulating the database files. The source code package including the detailed documentation [4, 5], the database files, and the image files are can be found on the Internet at `http://isl.ee.washington.edu/IAPR/ICPR00/vehicle_detection/vehicle_detection.html`.

4. Vehicle detection performance

The vehicle ground-truth data set is used in developing and evaluating performance of vehicle detection algorithms. The Maryland algorithm [10, 1] uses edge detection and generalized Hough transform (GHT). The performance is improved by a Bayesian classifier based algorithm [6] using eight input features. The performance is further improved recently by the introduction of the centroid uncertainty feature [7, 8] to Bayesian classifier. These algorithms are designed to detect target vehicles which meet certain size and orientation specifications.

In our experiment, not all vehicles are to be detected. The ones of interest are referred to as the target vehicles. They have lengths of 13 ± 5 pixels, widths of 5 ± 2 pixels, and orientations of 90 ± 1 degrees. Other vehicles which do not meet the specifications are the non-target vehicles and not to be detected. The distribution of these vehicles in the images are summarized in Table 1. [2]

The algorithms are applied independently on the images in the data set. The vehicles declared by the algorithms are

[2] Due to difficulty in computation, two large sub-images (`fhn717_56` and `fhn719_39`) are not included.

Table 1. Some statistics for the VVFH data set. Size is the total number of pixels in millions; the hash sign (#) stands for the number of; T.V. stands for target vehicles (meeting specifications); N.T.V. stands for non-target vehicles.

	size	sub-images	# T.V.	# N.T.V.
fhn75	23.8	30	1464	4594
fhn78	32.8	84	1745	5673
fhn711	23.0	24	2875	6192
fhn713	28.5	31	3638	7548
fhn715	45.0	60	5641	9264
fhn717	22.2	55	5438	6368
fhn719	24.1	51	4323	4885
total	199.4	335	25124	44524

compared to the ground-truth description of the vehicles. A ground-truth target vehicle is considered as being correctly detected if the centroid of a declared vehicle is within a 5-pixel radius of its own centroid. A declared vehicle is considered as a false alarm if there is no ground-truth target vehicle within a 5-pixel radius of its centroid. The detection rate is the percentage of ground-truth vehicles that are correctly detected. The false alarm rate is the average number of false alarms per million-pixel image area. Each algorithm is run multiple times with different values for the tuning parameters. The receiver operating characteristic (ROC) curves are shown in Figure 2. The reason for the low performance by the GHT is due to the unreliable edge detection and the medium to high amount of clutter in the imagery.

5. Summary

We have announced the availability of an image data set with manually generated ground-truth. The data set is suitable for use in development and performance evaluation of vehicle detection algorithms in aerial imagery. We briefly described the protocol for the ground-truth generation procedure and the major components of the information stored in the database. The ground-truth data and the software package for its manipulation are available on the Internet. As an example of using the data set, the performance of a few algorithms are compared by showing their ROC curves.

Acknowledgment

The manual annotation of the images was done by the ISL students Lixin Gong, Gang Liu, Ming Liu, Mingzhou Song, Yalin Wang, Feng Zhuge.

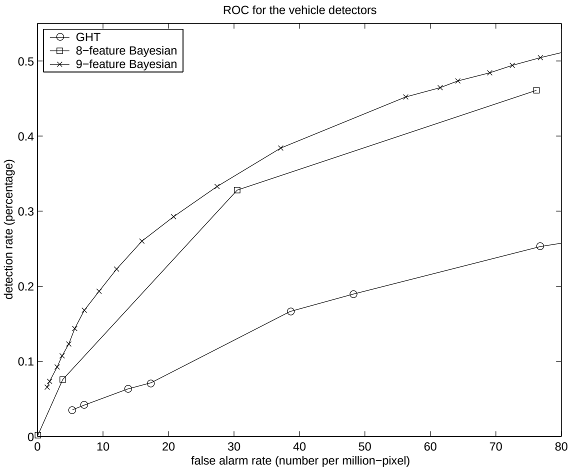

Figure 2. ROC curves of three vehicle detection algorithms on the VVFH data set. Upper – 9-feature Bayesian, middle – 8-feature Bayesian, lower – GHT.

References

[1] P. Burlina, V. Parameswaran, and R. Chellappa. Sensitivity analysis and learning strategies for context-based vehicle detection algorithms. In *1997 Image Understanding Workshop Proceedings*, pages 577–83, 1997.

[2] R. M. Haralick and L. G. Shapiro. *Computer and Robot Vision*. Addison-Wesley Publishing Company, Inc., 1992.

[3] Q. Ji and R. M. Haralick. Breakpoint detection using covariance propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):845–51, Aug 1998.

[4] G. Liu et al. Ground-truthing the vehicles in the vertical-view Ft. Hood images. Technical report, Intelligent Systems Laboratory, Dept. of EE, Univ. of Washington, Seattle, WA 98195, Dec 1997.

[5] G. Liu et al. Documentation of the software package for the vehicle database for the vertical-view Ft. Hood images. Technical report, Intelligent Systems Laboratory, Dept. of EE, Univ. of Washington, Seattle, WA 98195, Dec 1997.

[6] G. Liu, L. Gong, and R. M. Haralick. Vehicle detection in aerial imagery and performance evaluation. Submitted to the journal *CVIU*, Sep 1999.

[7] G. Liu and R. M. Haralick. Using centroid covariance in target recognition. In *ICPR'98 Proceedings*, pages 1343–46, Brisbane, Australia, August 1998.

[8] G. Liu and R. M. Haralick. FLIR ATR using location uncertainty. *Journal of Electronic Imaging*, 9(2), April 2000.

[9] C. McGlone. Fort Hood image distribution, second data set. README file with data distribution, Mar 1995. Version 1.0.

[10] V. Parameswaran, P. Burlina, and R. Chellappa. Performance analysis and learning approaches for vehicle detection and counting in aerial images. In *ICASSP'97*, pages 2753–6, 1997.