A Multi-Agent Vision-Based System for Vehicle Detection

Saeed Samadi and Farhad Mohammad Kazemi

Electronic Engineering Dept., Research Institute for Food Science & Technology, Khorasan Science & Technology Park, Mashhad, Iran
Computer Engineering Dept., Payame Noor University, Mashhad, Iran

Abstract: In this paper a multi-agent system approach for vehicle detection in image is proposed. The goal is to be able to localize vehicles in a given image. Developed agents are capable of detecting pre-specified shapes after image processing. Cooperation involves communicating hypotheses and resolving conflicts between the interpretations of individual agents. Specifically in the proposed system, eight process agents, consisting of edge, contour, wheel, LPL (License-Plate Line), LPR (License-Plate Rectangle), PCV (Plate-Candidates Verification) and vehicle symmetry agents, were developed for vehicle detection in various outdoor scenes. In the testing data, there are 500 car blobs and 100 non-car blobs. It is shown that the system effectiveness on Detecting vehicles in various outdoor scenes is about 90%.

Key words: Image processing · Multi-agent system · Vehicle detection · Conflict Resolution · Agent Cooperation

INTRODUCTION

Beymer et al. divided the methods of vehicle detection into four classes as 3D model-based, regional-based, active contour-based and feature-based, each of which has its advantages and disadvantages as described [7]. Achler et al., Gupte et al. and Koller et al. used an estimated background to extract moving objects and updated the background for real-time luminance changes [8-10]. However, this method requires an initial background with no foreground objects and updating the background information occupies a considerable time in the whole process of detection. Haag et al. used edge information and set up an optical flow model to detect vehicles [11]. Handmann et al. performed the detection using a neural network, based on local-oriented coding and entropy analysis [12]. However, there still exist a number of open problems to be solved, one of which is congestion. In this case, heavy traffic flow makes inter-vehicle occlusion a common scenario and further makes detecting vehicles more difficult. All methods referred above do not explicitly consider occlusions and are not applicable under such condition. For segmenting vehicles under occlusions, Kamijo et al. utilized a Markov random field (MRF) model to segment and track vehicles, assuming that every vehicle should appear unoccluded at a certain starting area of the interested area first [13].
Kanhere et al. detected and tracked features based on several motion-related cues to segment vehicles from videos taken from a low-angel camera [14]. However, these methods all use information such as background and motions and should be performed on videos or at least image sequences. Despite the large amount of literature on vehicle detection using videos, the detection using static images has relatively less work done, due to lack of motion information that many current methods require. However, when facing a heavy traffic (which is common in the urban area), the traffic slows down or even halts for minutes and vehicles get occluded. Even using videos, it is difficult to extract motion information or update background for detection. Actually, in such cases, we come to the problem of detecting vehicles in static images. Moreover, we believe that static images already contain enough information for detection, which is less redundant than images and can lighten calculation pressure. Detection in static images is also a more fundamental task and can be considered as a step or a subtask whose information can be integrated during video-based detection. Some attempts have been made to detect vehicles in static images. The methods are generally model-based in order to describe what a “vehicle” in the image is and follow the general object recognition ideas originated from computer vision. Detection has been performed as in the work of Sullivan and Roller et al., both of them used vehicle models and a ground-plane constraint for orientation [15, 16]. Zhou et al. casted the vehicle detection as a pattern recognition problem and trained a PCA-and-wavelet-based classifier to detect vehicles in static images [17]. Rojas et al. and Tsai et al. used a similar method by searching for important “vehicle colors” to detect vehicles, but required relatively strict requirements to image qualities such as good luminance and little shadow [18, 19]. Methods described in Ref. 20 used 2D shape models to detect vehicles in aerial images. The object recognition-based idea can be further seen in the recent works such as [21] and [22]. All of the methods referred above mainly focus on detecting vehicles or reasoning about their categories and do not consider segmentation as the main issue. Thus they either can be applied to the unoccluded conditions only, or do not consider the solution of inter-vehicle occlusions explicitly in their work. As for occlusions, Song et al. used a Bayesian method to segment vehicle blobs based on 2D black and white images projected from 3D vehicle shape models [23]. Oberti et al. modeled the shapes of vehicles by corners [24]. Pang et al. used a decomposition method based on cubic models to segment occluded vehicles [25]. However, although these methods can be performed on static images and get promising results, they all require certain initial information, most commonly background extraction. Such initial information is hard to obtain simply by static images. Thus these methods cannot be performed on static images alone and require other prior information (usually obtained via videos) to assist the initialization of their algorithm.

Yangqing Jia et al. propose a new vehicle detection approach based on Markov chain Monte Carlo (MCMC) [26]. They mainly discuss the detection of vehicles in front-view static images with frequent occlusions. Models of roads and vehicles based on edge information are presented, the Bayesian problem's formulations are constructed and a Markov chain is designed to sample proposals to detect vehicles. Using the Monte Carlo technique, they detect vehicles sequentially based on the idea of maximizing a posterior probability (MAP), performing vehicle segmentation in the meantime. Their method does not require complex preprocessing steps such as background extraction or shadow elimination, which are required in many existing methods.

Nicholas A. Mandellos et al. present an innovative system for detecting and extracting vehicles in traffic surveillance scenes [27]. This system involves locating moving objects present in complex road scenes by implementing an advanced background subtraction methodology. The innovation concerns a histogram-based filtering procedure, which collects scatter background information carried in a series of frames, at pixel level, generating reliable instances of the actual background. The proposed algorithm reconstructs a background instance on demand under any traffic conditions.

This paper deals with on a new methodology for computer vision which tries to enhance the carrying out of three specific aspects: (i) integration of knowledge and uncertainty; (ii) cooperation between visual tasks. (iii) enhance image processing tasks. We will then present our design of a general multi-agent object detection system.

Advantages of multi-agent image interpretation are [28]:

- Possibility of separated knowledge representation from different image domains.
- Possibility of separating image processing algorithms from control strategies, accommodating a large variety of control heuristics and selecting the best processing methods for different data situations.
Ease of construction and maintenance.
Ability to benefit from parallel architectures.
Focusing ability: Not all knowledge is needed for all tasks. Both spatial and interpretive focus possible.
Heterogeneous problem solving.
Reliability: An agent may be corrected by others.

Several multi-agent image interpretation systems have been reported in the literature. In Ref. [29] a system is presented with low-level agents that produce partial data-driven edge-based segmentations of medical images which are merged into a global result. Agents have only indirect influence on the global result and have, for instance, no possibility to negotiate about segmentation methods and parameter settings with neighboring agents facing similar segmentation problems.

Rodin et al. present a parallel image processing system based on simple reactive agents in [30]. Agents act according to a perception-action model without problem solving or deliberation. The system is applied to the detection of concentric striate, like the year rings of trees, with darkening and lightening agents.

Agents in the system by Liu et al. can either “breed” or “diffuse” (search) with the aim of producing segmentation results similar to split-and-merge algorithms [31]. Agents are purely reactive with predefined behavior determined by contrast, mean and variance of pixel intensities within a certain region. The agents are claimed to be more robust and efficient than split-and-merge algorithms in case of complex shapes.

Clouard et al. describe the automatic assembly and control of a chain of image processing operators using high-level knowledge to achieve a given image processing task [32]. Their system (Borg) is comparable to the image processing task of a single agent in our system. The goal of this system is always only to detect a single type of objects and the possible interaction with other kinds of objects is beyond the scope of the system, as is semantic interpretation. Issues like occlusion and conflicting interpretations are not considered.

Bovenkamp et al. present a novel multi-agent image interpretation system which is markedly different from previous approaches in especially its elaborate high-level knowledge-based control over low-level image segmentation algorithms [33]. Agents dynamically adapt segmentation algorithms based on knowledge about global constraints, contextual knowledge, local image information and personal beliefs. The agent knowledge model is general and modular to support easy construction and addition of agents to any image processing task. Each agent in the system is further responsible for one type of high-level object and cooperates with other agents to come to a consistent overall image interpretation. Cooperation involves communicating hypotheses and resolving conflicts between the interpretations of individual agents. That system has been applied to IntraVascular UltraSound (IVUS) images which are segmented by five agents, specialized in lumen, vessel, calcified-plaque, shadow and side branch detection. Through the Ref. [34] the characterization of a comfort model, enriching that proposed by Fanger's theory with an adaptive approach, is carried out using a Multi Agent System (MAS).

Kamal E. Melkemi et al. propose a new distributed image segmentation algorithm structured as a multiagent system composed of a set of segmentation agents and a coordinator agent. Starting from its own initial image, each segmentation agent performs the iterated conditional modes method, known as ICM, in applications based on Markov random fields, to obtain a sub-optimal segmented image [35]. The coordinator agent diversifies the initial images using the genetic crossover and mutation operators along with the extremal optimization local search. This combination increases the efficiency of their algorithm and ensures its convergence to an optimal segmentation.

A situated approach to Markovian image segmentation is proposed based on a distributed, decentralized and cooperative strategy for model estimation [36]. According to this approach, the EM-based model estimation is performed locally to cope with spatially varying intensity distributions, as well as non-homogeneities in the appearance of objects. This distributed segmentation is performed under a collaborative and decentralized strategy, to ensure the consistency of segmentation over neighboring zones and the robustness of model estimation in front of small samples. Specific coordination mechanisms are required to guarantee the proper management of the corresponding processing, which are implemented in the framework of a reactive agent-based architecture.

J. Fleureau et al. propose a new technique for general purpose, semi-interactive and multi-object segmentation in N-dimensional images, applied to the extraction of cardiac structures in MultiSlice Computed Tomography (MSCT) imaging [36]. The proposed approach makes use of a multi-agent scheme combined with a supervised classification methodology allowing the introduction of a priori information and presenting fast computing times.
System Architecture: A multi-agent system is a good architecture for an image interpretation system. Agent platform and Global view of the multi-agent system architecture as applied to vehicle detection shown in Figure 1 and Figure 2.

System Control: In this system, we have identified the following four basic decisions that an agent can make:

Image Processing: Object detection, object adjustments.

Communication: All operators for communication between agents (send, receive, process-message, etc.).

Organization: Storage, retrieval and removal of image objects.

Conflict Resolution: Resolving interpretation conflicts between agents.

The Structure of Agents

Edge Agent: The purpose of this agent was to enrich the edge features. This will improve the successful rate of the LPL (License-Plate Line Agent) and LPR (License-Plate Rectangle Agent) Agents. The algorithms sequentially used in this agent are graying, normalizing and histogram equalization. After having obtained a grey-scale image, we use Sobel filters to extract the edging image, then thresholding the image to a binary one. The resulted images are used as inputs for the contour, LPL, LPR and wheel agents (Figure 3).

Contour Agent: In order to detect regions of plate-candidate image, we apply contour agent for detecting closed boundary objects. However, this agent has difficulties in processing bad quality images due to scratches, plug-in helixes. In these cases, the contour agent produces incomplete closed boundary lines that do

In Section 2, the system architecture is introduced while in Section 3, we present the system control module. In Section 4, we present the structure of agents. In Section 5, the Behaviors of agents is presented. Finally, we present the experimental results using the test images that consist of various outdoor scenes in Section 6 and conclusion is presented in Section 7.

The multi-agent system is organized around a communicating agent which manages a population of situated agents which segment the image through cooperative and competitive interactions. Some other useful methods are also explained in [37,38].

In Section 2, the system architecture is introduced while in Section 3, we present the system control module. In Section 4, we present the structure of agents. In Section 5, the Behaviors of agents is presented. Finally, we present the experimental results using the test images that consist of various outdoor scenes in Section 6 and conclusion is presented in Section 7.
Fig. 4: The figure illustrates all the possible values for directions.

Fig. 5: Image

not contain correctly the plate-images. But, edge and contour agents in cooperation with together jump to the conclusion for solving this problem (Figure 5).

Trace Contour: First, we trace the outline of an object in binary image. Nonzero pixels belong to an object and zero pixels constitute the background. Suppose A is a two-element vector specifying the row and column coordinates of the point on the object boundary where we want the tracing to begin. We specify the initial search direction for the next object pixel connected to A. We use strings such as '0' for east, '1' for northeast, to specify the direction. The following figure (Figure 4) illustrates all the possible values for directions. We get certain number of boundary pixels for each region (normalize).

LPL Agent (License-Plate Line Agent): This agent detects lines from images. The Hough Transform (HT) is applied to the image to extract lines from object-images [39,40]. The parameterized version of the Hough Transform due to Duda and Hart, states that if a line whose normal makes an angle $\theta$ with the x axis and has a distance $\rho$ from the origin is considered, the equation of a line corresponding to any point $(x,y)$ on this line is given by $\rho = x \cos \theta + y \sin \theta$ (1).

The coordinates of each line along with its $\rho$ and $\theta$ values are examined and a chain of connected line segments is formed. Any chain formed by the extracted line segments is a candidate license plate if it defines a closed loop and the number of its constituent line segments is 4. Ideally, closed loops consisting of 4 line segments are declared as license plates if they meet the following criteria:

- Alternate line segments have the same peak value, which indicates that they are of the same size.
- Alternate line segments have the same value of $\theta$, which indicates that they are parallel to each other. (we looked for two parallel lines, whose the contained region is considered plate-Candidates)
- Adjacent line segments have a $\theta$ separation of 90°, to form right angled corners of the license plate.
- Ratio of the non-parallel edges is a constant.

Each candidate is verified by using a Plate-Candidates Verification Agent. However, the main limitation of this agent is the time required since the Hough transform is applied to a usually great number of pixels. Especially, the larger image is, the slower the agent is. The speed of the algorithm may be improved by thinning image before applying the Hough transform. However, the thinning algorithm is also slow. For vehicle tracking purposes, on successful detection of a license plate, a new window of interest is defined around it and all further processing is limited to this window. If no license plate is detected in the present frame, the previously defined window is used for processing in the next iteration. If no vehicle is found within the window of interest for 3 successive iterations, the whole image is defined as the window of interest and the system then tries to find the vehicle from the entire image (Fig. 6).

LPR Agent (License-Plate Rectangle Agent): We aim to detect patterns that can characterize rectangles. To do so, we defined LPR Agent (License-Plate Rectangle Agent). In this section, we explore geometric characteristics of a rectangle in the domain of the Hough Transform and such characteristics are used for rectangle detection directly in the Hough space. The proposed agent works for rectangle with unknown dimensions and orientations and does not require the extraction and/or grouping of linear segments (i.e., it is applied directly to the edge map).

Some works have been done about shape description in the Hough Space. Rosenfeld and Weiss proved that a convex polygon is uniquely determined by the peaks of its Hough transform (in fact, these peaks form the convex hull of the polygon). However, we face a different problem: detecting rectangles in images containing several objects. We want to detect patterns in the Hough Space that can characterize rectangles.
For that purpose, we note that a rectangle has specific geometric relations that can be detected directly in the Hough Space. The proposed algorithm for this agent is explained in Ref. [41].

**PCV (Plate-Candidates Verification) Agent**: One of the agents for candidate evaluation is applied on images obtained from the LPL and LPR agents to separate plate object (Plate-Candidates Verification Agent).

Our evaluating plate-candidates agent based on one main module that evaluates ratio between height and width of candidate (Figure 7).

**Ratio Between Width and Height of Candidate**: In this module, we check and only select out candidates that have height and width ratio satisfied pre-defined constraint:

\[
\min \text{WHRatio} < \frac{H}{W} < \max \text{WHRatio}
\]

Since there are two main types of Iran plate: 1-row and 2-rows (Figure 7), we have two appropriate constraints for two types.

- \(3.2 < \frac{W}{H} < 4.2\) with 1-row plate-candidates
- \(0.7 < \frac{W}{H} < 1.3\) with 2-rows plate-candidates

The candidates that satisfied one of two above constraints are selected and passed to the next module.

**Vehicle Agent**: Vehicle Agent (vehicle detection) consists of 3 modules, that is, the preprocessing module working on the input raw image, the vehicle candidate extraction module by a shadow region and a template and the validation module by a prior knowledge [42].

**Preprocessing Module**: We apply histogram equalization below the vanishing line in the raw image. This process can clear the gap between the dark road and other objects on the road can easily extract the shadow region used as the first feature of the vehicle in the day time. After this process, we create a binary image by a low threshold that can eliminate the bright region.

**Vehicle Candidate Extraction Module**: As the shadow region between the vehicle and the road appears in the day time regardless of weather conditions or low light conditions and we use the shadow region as the first feature of a vehicle. The shadow region is defined as where the intensity value drops below a certain threshold. As in Figure 8, the region with a certain width and height is scanned from bottom to top until it satisfies the constraint.
At First, the histogram profile of $x$ direction is calculated in the input image $I(x, y)$ using a horizontal scanning and then the vertical position of shadow region $Vs$ is extracted using variable shadow height by a vertical scanning like Figure 8(a). The final shadow region is determined by the “AND” operation of the $Vs$ that represents indices with a high intensity change and the dark area in the preprocessed binary image. After transforming extracted shadow regions to the world coordinate by the inverse perspective transform (IPT), these regions are split among different lane while some regions in non-region of interest are filtered out like Fig. 8 (b). If the lane information does not exist like the failure of the lane detection or due to an intersection, etc., three virtual lanes are generated instead of the proper lane width estimated from the ego-vehicle position used by the splitting and filtering of shadow regions. Although one vehicle on the lane change may be clustered to two vehicles, this problem is resolved by the symmetry scanning process.

**Vehicle Candidate Validation Module:** The intensity change by the road repair, road sign, guardrail, oil spill and shadow, by non-vehicle objects in the road side may be mistaken for shadows. Thus the validation of the vehicle candidate is required and there are many false-positive errors by shadow pattern at the road boundary and the bottom of the guardrail in real experiments. Because, this shadow pattern has a regular histogram in some continuous section, we eliminate this shadow pattern before validation step. At First, the symmetry scanning is used for extracting the exact left and right boundaries. For symmetry scanning, we construct an edge angle map using Sobel operators and scan the left and right regions with the center line of the current vehicle position. The window size for symmetry scanning is the pixel width of the current vehicle. Then, we find the symmetry axis with the maximum symmetry rate by as follows:

$$Symmetry = \frac{s^2}{n}$$

Where $s$ is the number of the absolute same edge angle and $n$ is total number of edges. By checking the two vertical edges at both sides of the vehicle, this system finally validates the candidates using the vertical edge histogram. If vertical histograms at both sides exceed certain ratio of the pixel width of a vehicle, this candidate is valid.

**Wheel Agent:** This section presents an agent for circle or wheel detection according to the next side of vehicle’s images (Figure 9). The proposed agent also belongs to the multipoint transform category. Theoretically, to evaluate circle parameters for point triplets in an edge image containing $n$ points, $C3$ enumerations of the edge points have to be examined. Anyway, if specific relations of the circle points are sought, the required number of enumerations can be reduced enormously.
The idea of the proposed agent comes from a property of a circle that any three points lying on it and with two of these points being the diameter, the third point forms a right angle triangle with the remaining two points. Using such a criterion to form specific sets of a feature point triplet, the number of enumerations can be reduced to C2 only. The performance of the proposed agent is promising in that it is both fast and memory saving as compared with conventional Hough transform methods [43]. Some other useful information can be found in [44] to [46].

**Symmetry Agent:** Symmetry agent is one of the basic features characterizing natural shapes and objects. It is believed to enhance recognition and reconstruction and is likely to be employed in pre-attentive vision. This section presents an agent for extending the normalized cut (n-cut) segmentation algorithm to find symmetric regions present in natural images. We use an existing algorithm to detect possible symmetries present in an image, quickly. The detected symmetries are then individually verified using the modified n-cut algorithm to eliminate spurious detections. The weights of the n-cut algorithm are modified so as to include both symmetric and spatial affinities. A global parameter is defined to model the tradeoff between spatial coherence and symmetry. The reader is referred to [47] for a description of the n-cuts algorithm. This agent was based on a modified version of n-cuts algorithm to include spatial and symmetry affinities. The blue axis is the principal axis of symmetry. The symmetric structure of the wheels causes them to fall into the symmetric segment (Figure 10).

**Behavior of Agents:** The internal control system is implemented by a rule system:

- Features extraction: each agent stores different types of information in a local database.
- Exploration: agents are exploring environment (image).
- Cooperation: agents cooperate with each other to enhance the quality of their merging plan.

**Cooperation:** This section presents cooperative behaviors between agents. From the extracted edging image, we use the contour agent to detect closed boundaries of objects. These contour lines are transformed to Hough coordinate to find two interacted parallel lines (one of two parallel lines hold back the other 2-parallel lines and establishes an parallelogram form object) that are considered as a plate-candidate. Since there are quite a few (black) pixels in the contour lines, transforming these points to Hough coordinate required much less computation. Hence, the speed of the algorithm is improved significantly without a noticeable loss of accuracy (Edge agent + contour agent + LPL agent). However, some images may exist that include other objects such as glasses, head lights and decorated goods. These objects may also have the shape of two interacted 2-parallel lines, and therefore, are also falsely detected as plate-candidates. To reject such incorrect candidates, we implement an agent for evaluating whether a candidate is a plate or not (PCV Agent). LPR and LPL agents will be validated by an agent that is called PCV Agent (Figure 1). Then, PCV and vehicle agents in cooperation with together jump to the conclusion for detecting any vehicle. The obtained consequences by PCV and vehicle agents may be in contradictions that all these contradiction are solvable with some (If-then-else) rules. For detecting the wheels, symmetry and wheel agents must cooperate together and precisely at this time contour and wheel agents have the same act, too. If the gained outcomes by wheel and contour agents be in contradictions, these contradictions are solvable with some (If-then-else) rules, too.

**Experimental Results:** The proposed system was tested on images captured in various outdoor scenes for vehicle detection. This data set includes 600 images that are divided to 250 images from front side or back side of vehicle, 250 images from next side of vehicle and 100 images without any scence of vehicle. Table 1 shows the system performance on the various images. Figure 11 shows the vehicles detected by our system on four images. In Figure 11.d two vehicles not detected.
We showed through experiments that our system is effective in detecting vehicles in various outdoor scenes. These results showed that agents with different image processing behaviors can extract complementary information in the image.

**CONCLUSION**

The main contribution of this work is a new architecture for vehicle detection using a vision-based Multi-agent System. The efficiency of the system's different agents is shown. With the proposed system, it is easy to improve and modify individual agents and to add new agents. In general, control over the image processing task is also very satisfactory. Evaluation results showed the ability of the system to detect various vehicles in various scenes.

**REFERENCES**


42. SamYong Kim, Se-Young Oh and JeongKwan Kang, 2005. YoungWoo Ryu; Kwangsoo Kim; Sang-Cheol Park; KyongHa Park; Front and rear vehicle detection and tracking in the day and night times using vision and sonar sensor fusion, Intelligent Robots and Systems, In the proceedings of the 2005 IEEE/RSJ International Conference, pp: 2173-2178.


