Design and Implementation of Smart Vehicular Camera for Real-Time Visual Metadata Extraction and Sharing

Sanghyun Son, Beomjun Kim, Yeonsu Jung and Yunju Baek
School of Computer Science and Engineering, Pusan National University, Busan, Republic of Korea
sonsang@eslab.re.kr, bjkim@eslab.re.kr, yeonsu@pusan.ac.kr, yunju@pusan.ac.kr

Abstract - The number of vehicles has grown, accompanied by an increase in accidents. Thus, advanced driver assistance systems providing the status of a vehicle and the surrounding environment using various sensor data are being studied. We designed and implemented a smart vehicular camera device for real-time extraction and visual metadata sharing. Moreover, we propose a technique for extracting the metadata from an input image and sensor data using an image recognition algorithm. Moreover, we propose the S-ROI and D-ROI techniques, which set the region of interest in an image frame to improve the image processing, and a pattern check technique for increasing the recognition rate. We developed the main server for information sharing and a real-time road view service. We also evaluated the performance of the two ROI methods, and confirmed that the video processing speed of S-ROI and D-ROI are 3.0- and 4.8-times better than a full-sized frame analysis, respectively.

Keywords: region of interest; metadata extraction; smart vehicular camera; image recognition; information sharing

1 Introduction

Along with urban development, the number of the vehicles on the roads has increased, with the greater complications accompanying such development [1]. The need for a technologically-based increase in vehicle safety to mitigate the risk of traffic accidents is increasing in proportion to the increasing number of vehicles [2]. An advanced driver assistance system provides a variety of information for solving traffic and safety problems [3]. Initially, driver assistance systems were studied in terms of driver convenience. Recently, this system provides safety services such as alerts of lane departure and directly vehicular control. In major advanced countries, a trend in enforcing the mandatory installation of such safety-related systems has begun.

IT companies such as Google and Apple, along with automobile manufacturers such as BMW, WV, and Mercedes Benz are researching autonomous driving and driver assistance techniques. Mobileye [4] ADAS provides more information using a computer vision algorithm and a proprietary hardware device than other types of sensor-based ADAS. In image-processing based ADAS, the front and rear images are recorded and analyzed to provide road condition information.

Although the existing driver assistance systems are different from utilizing of sensors and algorithms, the vehicle can recognize and utilize the information separately. Vehicles are able to collect various types of road information using image processing and an existing sensor technique; however, information collected by a vehicle is limited when compared to shared information. Therefore, it is necessary to allow vehicles to share information through a network.

For vehicle-to-vehicle wireless communication, research based on wireless sensor networks and ad-hoc networks [5] is underway. Moreover, IEEE 802.11p WAVE [6] is a related standard that defines the PHY and MAC layer for V2X communication. However it is difficult to apply an ad-hoc network because such a network requires a high penetration of networking devices and infrastructure. To solve this problem, a mobile cellular network can be used. Although the use of a mobile network is limited by fee it is possible to use such a network through the driver’s smart phone.

When a mobile cellular network is used, a metadata extraction technique is required to minimize the data transmission. For visual data, visual metadata analyzing an input image are extracted. For sensor data, specific information from various sensors is extracted. Therefore, extracting metadata minimizes the data size.

An image analysis is conducted using a computer vision algorithm [7]. Such algorithms find specific patterns based on learned data within a frame. An algorithm used for an image analysis searches sequentially by applying various mask filters, and thus requires a great deal of computing power. For image processing, we applied a high-performance processor and reduced ROI technique to improve the processing speed.

For this paper, we designed and implemented a computer-vision based smart vehicular camera for real-time extraction and sharing of visual metadata. The smart vehicular camera extracts various types of visual metadata for analyzing an input image and shares the data through a network. Moreover, we propose a D-ROI technique for minimizing the computational load, and a pattern check technique for increasing the recognition rate. We developed a main server for information sharing and a real-time road view service. Finally, we evaluated the implementation of the smart vehicular camera device and confirmed its image processing speed and recognition rate.
2 Related Works

This section describes the advanced driver assistance systems associated with the proposed smart vehicular camera and image recognition technologies.

ADAS provides a variety of audio-visual information to the driver to prevent accidents that may occur during operation. The system alarms to the driver using sensors to recognize risks on the road, and the driver confirms the warning and response. The main functions of ADAS consist of forward collision avoidance, lane departure warning function, blind spot monitoring and rear monitoring.

ADAS with these features are likely to continue to grow due to the changes of the social structure, regulatory strengthening. Consumers are increasingly in demand, and since technological development and mass production are reduce the production cost. Therefore, ADAS market is expected to grow an annual average of 25% by 2017 [8].

The image recognition technologies for the vehicular environment is a part of computer vision technology. According to research of driver assistance system is underway, the image recognition technologies has attracted attention as complementary elements for vehicular sensors. Computer vision that is a branch of artificial intelligence aims to understand scene or feature on the image.

In order to perform these functions on the smart vehicular camera, many researchers use the computer vision library which is OpenCV (open computer vision). This library is possible that various recognitions such as face, pedestrian and motion tracking on a variety of platforms, and provides sample codes in various program languages. [9]-[12] The proposed smart vehicular camera uses this library to recognize pedestrian, license plate, lane and lane departure detection.

3 Smart vehicular camera

In this section, descriptions of the hardware prototype of the proposed smart vehicular camera device and the image processor, as well as the region-of-interest technique used to extract the visual metadata, are provided.

3.1 Smart camera device

A smart camera device must be capable of analyzing data from an input image and sensor data from onboard modules, and share the collected metadata over a network. Computer vision algorithms require high computational power, and thus high-performance embedded processors are required for the image processing device. We selected the high-performance Samsung Exynos 5410 core processor, which contains an ARM-based 1.6 GHz octo-core processor and a PowerVR SGX544MP3 GPU. The proposed prototype device includes an Exynos core, 2 GB of DDR3L RAM, an IEEE 802.11n Wi-Fi module, an HSDPA + WCDMA module, and various sensors. Table 1 shows the specifications of the proposed device.

The proposed device uses an HD class camera to record the road images, 3GPP Release 5 for the WCDMA module, and a Wi-Fi module for wireless networking capability. In addition, the device recognizes the current state of the vehicle using a GPS device, an acceleration sensor, and an OBD scanner. This information is forwarded to the driver over an LCD screen and speaker, which provide warning messages and alarms. Figure 1 illustrates the block diagram of the proposed smart vehicular camera device.

For operation of the smart camera device, the required software components are as follows: an embedded operating system for control of the peripheral devices, modules and libraries of the metadata extraction and wireless networking, and an application constituting all functions and the forwarding of information to the user. Figure 2 shows the software architecture of the proposed device.
We developed a prototype device based on the design of the smart vehicular camera. Figure 3 shows the implementation of a smart vehicular camera device. The device, based on a Hardkernel Odroid-XU [12] development board, includes several modules such as a camera, GPS device, Wi-Fi, a WCDMA module, an OBD scanner, an LCD panel, and a speaker. The OBD scanner was developed using a CAN transceiver and STM32F103 microcontroller.

### 3.2 Visual Metadata Extraction Technique

For image recognition, we developed visual metadata extraction module by applied OpenCV library. In the various recognition functions, we implemented image recognitions such as pedestrian, license plate, lane, lane departure. To recognize pedestrian, we applied a HOG algorithm in the library and used SVM classifier with learning data for pedestrian recognition.

For vehicular license plate recognition, we applied a rectangle detection algorithm and an optical character recognition (OCR) algorithm to our recognizing application. After detect rectangles in image frame, analyze candidate rectangles using OCR to recognize a vehicular number. For lane recognition, the application finds straight lines and selects lanes that straight lines toward the vanishing point in the region. The application uses Hough transform to find straight lines and calculates the vanishing point using random sample consensus (RANSAC) algorithm. The lane departure function is verifying the gradient of lane continuously, and the function determines lane departure condition when the gradient changes more than a predetermined value.

### 3.3 ROI Technique

The image recognition technique is conducted using many simpler comparison calculations compared to other applications, and the computing time required for the feature extraction is increased when the size of the target region of the image frame is increased. Thus, it is recommended to reduce this region by setting the ROI according to the recognized target. Therefore, the application sets the ROI according to the camera attached to the vehicle. A vehicular camera that consistently monitors the same area is able to set the ROI intuitively.

Owing to the use of a narrow surveillance area, the computing time is reduced and recognition errors occurring outside of the search region are avoided. The camera attached to the front of the vehicle monitors the center region, and thus the application is able to increase the image processing time by excluding the top and bottom regions, such as the sky and hood of the vehicle. Likewise, the side regions such as the other side of the road and the pedestrian area are excluded from the ROI.

In this paper, for pedestrian and license plate recognition, we set the ROI to 33% of the frame. For the lane and lane departure recognition, we set the ROI to a narrower region than that of the S-ROI used for pedestrian recognition. For the lane recognition, we set the ROI to 12% of the frame, and for the lane departure recognition, the ROI is set to 4%. The ROIs used for recognition are defined as static ROIs (S-ROIs). An S-ROI is always the same size regardless of the recognition result, and the exact position and size of the ROI can change according to the camera conditions.

Although the S-ROI technique is simple to set up and easy to apply, it is inefficient for a road environment, such as when monitoring the same region for a predetermined time. In particular, when pedestrian and license plate recognition algorithms find a target object, the probability that the target will be found near the same region in the next frame is high. Considering these features, the application searches the S-ROI for recognition by default, and minimizes the ROI fitting the recognized target when the algorithms find their target. We defined and implemented this technique as a dynamic ROI.

If a target such as a pedestrian or license plate is recognized in a previous frame, the application temporarily reduces the ROI as follows. For the case of a pedestrian, most pedestrians appear in front of the vehicle and are crossing the road at a constant speed. Thus, the application sets a new D-
ROI by considering the pedestrian speed. Considering recognition-able human size, the size of the D-ROI for a pedestrian is 7% of the frame. For recognizing a license plate of the front vehicle, because the vehicle can move in all directions, the application sets a new D-ROI by considering the lane-to-lane width. The size of the D-ROI for a license plate is 11% of the frame.

The D-ROI condition is changed to normal when the target disappears for longer than a D-ROI timeout. The D-ROI technique reduces the image processing time; however, this technique has a problem in that it cannot recognize new targets during a narrow region search. Thus, the D-ROI technique has a timeout value, and this is designed searching S-ROI after a predetermined time. The timeout value varies according to the application or recognized target, and is set to achieve the optimal performance in terms of the processing time and recognition rate.

3.4 Pattern Recognition Techniques

For the proposed license plate recognition technique, rectangles are recognized within the frame to collect candidate license plates, after which the technique determines the existence of a license plate by checking the aspect ratio of the target rectangle. However, the error rate of this technique, i.e., determining that a rectangle is not a license plate, is high.

Table 2 shows the results of license plate recognition for various input image sizes. The results confirm that the recognition rate is low and the error rate is high. To solve this problem, we proposed a pattern check technique, which checks the center line of the candidate rectangle and counts the number of color pattern changes. The background color of a Korean license plate is white, and the characters are black, and thus the technique determines the existence of a license plate when the number of white-black pattern changes is over the threshold value. Figure 5 shows an example of pattern check technique for increasing recognition-rate.

3.5 Metadata Sharing Technique

To utilize the metadata collected from a smart vehicular camera device, information sharing is required. Research into wireless ad-hoc-based vehicle-to-vehicle communication is underway; however, it is difficult to apply to the ad-hoc network because such a network requires a high penetration of the networking device and infrastructure. Therefore, we used a mobile cellular network to communicate this information.

The number of methods using a mobile network is two. In the first method, the smart vehicular camera device uses a WCDMA module for direct communication. In the second method, the device connects to the smart phone of the driver and thus connects to its mobile network. Although the use of a WCDMA module is convenient, an additional network charge must be paid. When using a smart phone, the two devices are connected using a wireless local area network technique such as Wi-Fi or Bluetooth. Although the devices can connect through tethering, such a connection process is inconvenient.

The proposed smart vehicular camera device is able to use these two methods, including a WCDMA module and Wi-Fi module, and set the communication directly by default. If metadata are extracted, the information is stored in an internal database and the updated information is transmitted to the main server for information sharing. In addition, the server forwards this information to the nearby vehicles based on their GPS position information.

The main server used for information sharing collects the data from the vehicles and provides road condition information. This server stores the vehicular data such as the time, position, speed, road conditions, snapshot images, and video sequences. Moreover, the server transmits the information to the requesting vehicles and provides road images and condition information to the user by utilizing a Web-based viewer. Figure 6 shows the viewer screens such as the road condition and real-time street view. In figure 6 (a) and (b), the arrows on the Google map show the road conditions in accordance with the direction based on color, and the road conditions are confirmed through the uploaded

| TABLE II | RECOGNITION-RATE AND ERROR-RATE OF OBJECT RECOGNITION |
|-----------|-----------------|-----------------|-----------------|
|           | VGA  | WXGA | FHD  |
| Recognition-rate | 45.5% | 49.4% | 49.0% |
| Error-rate     | 30.6% | 68.3% | 75.5% |

<table>
<thead>
<tr>
<th>TABLE III</th>
<th>SPECIFICATION OF THE METADATA MEASURING PERIOD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Device</td>
<td>Technique</td>
</tr>
<tr>
<td>Camera</td>
<td>Record Image</td>
</tr>
<tr>
<td></td>
<td>Computer Vision (with ROI)</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>GPS</td>
<td>Satellite positioning</td>
</tr>
<tr>
<td>Accel.</td>
<td>Sensor measuring</td>
</tr>
<tr>
<td>OBD</td>
<td>CAN comm.</td>
</tr>
</tbody>
</table>
road images. Figure 6 (c) shows the speed change of the

target vehicle in the viewer.

4 Performance Evaluation

In this section, an evaluation based on the hardware

performance, image processing speed, and recognition-rate is

conducted.

The smart camera device uses various modules such as a
camera, GPS device, an accelerator sensor, and an OBD

scanner to extract the metadata. A still image is a type of
visual metadata and is analyzed as data using computer vision.
The still image is stored and forwarded when a request from
the main server is received. If image processing has been
conducted, the measurement time is changed based on the
computational load.

Image processing of a WVGA (800 x 480) input image
using the D-ROI technique takes 0.33 s for the proposed
device. When sensor metadata are extracted, the extraction
period is changed according to each sensor module. The GPS
module is able to change the measurement period based on
the setup value, and we set the GPS measurement period to 1
s. The acceleration sensor used for shock recognition is able
to measure approximately 15,343 raw datum per second. We
set the measurement period to 0.1 s using an average of ten
values, and determine a sudden change of acceleration using
150 average values. The OBD scanner used for collecting the
sensor data of a vehicle is able to take approximately 22
measurements per second. We set the measurement period to
0.5 s using the average of ten values. Table 3 shows the
specifications of the metadata measurement period.

We developed the proposed pattern-check technique
to increase the license plate recognition rate. For a VGA
sized image, two cases, i.e., rectangle detection and a check
of the aspect ratio of the candidate rectangles, and a color-
pattern check after determining the aspect ratio, are compared.
The results of the image recognition show that the recognition
rate of the proposed technique is increased by approximately
39% and that the error-rate is decreased by approximately
15%. We applied this pattern-check technique and conducted
a performance evaluation for the image processing speed.
Table 4 shows the result of license plate recognition using the
pattern check technique in the image of VGA size.

We evaluated the proposed S-ROI and D-ROI
techniques, and confirmed the processing speed and
recognition rate based on an evaluation of each algorithm.
For the evaluation of the ROI techniques, we experimented
with pedestrian and license plate recognition algorithms. We
confirmed the result of the S-ROI using only lane and lane
departure recognition algorithms. We evaluated the
processing speed of each algorithm, and measured the
average processing time of the algorithms overall. For an
evaluation of the hardware-based image processing

<table>
<thead>
<tr>
<th>Recognition-rate</th>
<th>Error-rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-check</td>
<td>Pattern check</td>
</tr>
<tr>
<td>Recognition-rate</td>
<td>45.5%</td>
</tr>
<tr>
<td>Error-rate</td>
<td>30.6%</td>
</tr>
</tbody>
</table>

Table IV

Result of Plate Recognition Using the Pattern Check Technique

We developed the proposed pattern-check technique
to increase the license plate recognition rate. For a VGA
sized image, two cases, i.e., rectangle detection and a check
of the aspect ratio of the candidate rectangles, and a color-
pattern check after determining the aspect ratio, are compared.
The results of the image recognition show that the recognition
rate of the proposed technique is increased by approximately
39% and that the error-rate is decreased by approximately
15%. We applied this pattern-check technique and conducted
a performance evaluation for the image processing speed.
Table 4 shows the result of license plate recognition using the
pattern check technique in the image of VGA size.

Table V

Recognition-rate and Error-rate of Recognizing Objects

 Pace
performance of the proposed device, we conducted experiments on various development boards such as an Odroid-X2 (Exynos 4412) [13] and Arndale Octa (Exynos 5420) [14].

We evaluated the S-ROI and D-ROI techniques for improving the image processing speed for pedestrian and license plate recognition on each device. While applying the ROI techniques, confirmed the fps and recognition rate. The recognition rate was the same for all devices because they all use the same algorithm. Figure 7 shows a graph of the results for fps when pedestrian recognition was applied. When applying the S-ROI technique, the recognition algorithms search the same reduced region, and thus the fps value is always similar. When applying the D-ROI technique, the recognition algorithms search the same sized S-ROI. If a target object is found, the size of the D-ROI is reduced, and thus the image processing speed is increased. Therefore, the two ROI techniques show the same FPS values when there is no target in an image. If a target object is found, the fps value of the D-ROI is increased.

The results of the license plate recognition in Figure 8 show that a plate is continuously present within the image. Thus, we confirmed that the fps value of the D-ROI is always high. A comparison of the results for each device shows that the proposed device has the highest fps value. Figure 9 shows the results of the recognition rate, which is determined as follows (1). In (1), \( TP \) denotes the true positive and \( FN \) represents the false negative. \( TN \) denotes the true negative and \( FP \) represents the false positive.

\[
\text{Recognition rate} = \frac{TP + FN}{TP + FN + TN + FP} \tag{1}
\]

We confirmed the results of the image recognition processing for each frame. The parameters for each recognition algorithm were set to minimize false positives. For the pedestrian recognition, we confirmed that the recognition rate using S-ROI is 70.7%, and using the D-ROI is 67.4%. For the license plate recognition, we confirmed that the recognition-rate using the S-ROI is 63.3%, and using the D-ROI is 45.7%. As a result, the recognition-rate when using the S-ROI is higher than that when using the D-ROI for the following reason. The D-ROI technique reduces the ROI size when the algorithm recognizes the target in an image. If the recognition is incorrect, the next frame recognition will fail with a high probability. The experimental results are given in Table 5.

We determined the average image processing time required to run all recognition algorithms for each device. In the experiments, the recognition processing times, including for pedestrians, license plates, lanes, and lane departures, were measured. We used three test devices and applied a full-sized frame, S-ROI, and D-ROI. Figure 10 shows the experiment results. The three devices received a WVGA image and applied four recognition algorithms. In the full-sized search, each device showed a rate of approximately 1 FPS. In contrast, the proposed device showed a processing time of 0.19 s. In the experiments, we confirmed an improved image processing speed of 3.0- and 4.8-times the original when using the S-ROI and D-ROI, respectively.

5 Conclusions

For this paper, we designed and implemented a computer-vision based smart vehicular camera for real-time extraction and visual metadata sharing. The proposed device analyzes a recorded image frame from the camera module and extracts the visual metadata, and then extracts the sensor metadata over the sensor modules. We developed the device and main server such that the device transmits metadata to the main server using a mobile cellular network to share information with other vehicles on the road. In addition, we proposed a dynamic ROI technique to minimize the computational load for image processing, and evaluated the performance.

We designed the smart vehicular camera device to include a high-performance embedded processor for real-time image processing, and evaluated the proposed device experimentally. The experiment results confirm that the image processing speed is increased 3.0-fold when the recognition algorithms apply the S-ROI, and 4.8-fold when the D-ROI is applied. However, when the algorithms applied the D-ROI, we confirmed that the recognition rate was
reduced. Therefore, the application should use the appropriate ROI technique based on the processing-time and recognition-rate requirements.

The proposed device is executable for an image processing speed of 5.3 fps. However, applications require a higher real-time processing speed for full input frames. Thus, we must improve the image processing speed by using advanced processors and libraries. For an object recognition technique, we have to research how to conduct an analysis under a variety of situations. Finally, we have to solve the problem of a lack in the number of libraries for image recognition processing when using the GPU core in an embedded processor. We have a plan to conduct research to improve the above problems.

6 Acknowledgement

This work was supported by the Center for Integrated Smart Sensors funded by the Ministry of Science, ICT & Future Planning as Global Frontier Project (CISS-2011-0031863). Corresponding author: Yunju Baek (yunju@pusan.ac.kr)

7 References


Fig. 10. Processing time of all recognizing operation