DRIVING ENVIRONMENT ASSESSMENT USING FUSION OF IN- AND OUT-OF-VEHICLE VISION SYSTEMS

S. Y. KIM[1)*], H. C. CHOI[1)], W. J. WON[2)] and S. Y. OH[1)]

[1)]Department of Electronic and Electrical Engineering, Pohang University of Science and Technology, Gyeongbuk 790-784, Korea
[2)]Daegu Technopark Venture 2nd Plant, Hosan-dong, Dalseo-gu, Daegu 704-230, Korea

(Received 20 June 2008; Revised 7 October 2008)

ABSTRACT–Because the overall driving environment consists of a complex combination of the traffic Environment, Vehicle, and Driver (EVD), Advanced Driver Assistance Systems (ADAS) must consider not only events from each component of the EVD but also the interactions between them. Although previous researchers focused on the fusion of the states from the EVD (EVD states), they estimated and fused the simple EVD states for a single function system such as the lane change intent analysis. To overcome the current limitations, first, this paper defines the EVD states as driver's gazing region, time to lane crossing, and time to collision. These states are estimated by enhanced detection and tracking methods from in- and out-of-vehicle vision systems. Second, it proposes a long-term prediction method of the EVD states using a time delayed neural network to fuse these states and a fuzzy inference system to assess the driving situation. When tested with real driving data, our system reduced false environment assessments and provided accurate lane departure, vehicle collision, and visual inattention warning signals.

KEY WORDS : Advanced driver assistance systems, Active appearance model, Lane and vehicle detection, Neural networks, Fuzzy inference systems

1. INTRODUCTION

Advanced Driver Assistance Systems (ADAS) support driver decision making to increase safety and comfort by issuing warning signals or by exerting active control in dangerous conditions (Kim *et al.*, 2008). Recently, the computer vision research community has nearly commercialized some vision based ADAS, such as Lane Departure Warning Systems (LDWS), Collision Warning Systems (CWS) and Drowsy Driver Warning Systems (DDWS), due to improvements in detection and recognition performance.

However, the overall driving environment consists of a complex combination of the traffic Environment (Wu *et al.*, 2007), Vehicle (Chung *et al.*, 2007), and Driver (Kim *et al.*, 2007) (EVD) (McCall *et al*., 2007). ADAS must consider not only events from each component but also interactions between them. For example, most LDWS use a specific threshold of the Time to Lane Crossing (TLC: the time until the host-vehicle gets to the left or right lane). Although the algorithm uses a well-defined TLC based on statistical analysis, the LDWS gives too many false alarms regardless of the driver's intention. Similarly, if a CWS using Time To Collision (TTC: the time until the host-vehicle reaches another vehicle) can consider the driver's attention, it could suppress the false alarms by adjusting the TTC threshold. Also driver fatigue and inattention warning systems using a driver's gazing angle can also be improved by useful contextual information on the surrounding traffic environment and the vehicle state. Therefore, except in some critical situations such as an abrupt variation of the TLC or TTC, the EVD states should be fused for an interactive ADAS.

Some related papers support the possibility of an interactive ADAS using fused EVD states. McCall *et al*. (2007) increased the time margin of TLC and improved the estimation probability of the lane change-intent by fusing the driver's head direction, the vehicle state, and the lane states, rather than just the vehicle and lane states. Fletcher *et al*. (2005) proposed an ADAS which could provide the road sign information that the driver missed by correlating the driver's gazing direction with the position of the road sign. Fletcher's ADAS could effectively reduce the driver's burden by suppressing useless information. Apostoloff and Zelinsky (2004) integrated the lane tracker with the gaze direction and they showed the relationship between the yaw of the driver's gaze and the yaw motion of vehicle. Cheng *et al.* (2007) and Stiller *et al.* (2007) considered the ADAS as an interactive closed-loop system of EVD states and proposed an interactive road situation analysis framework and a cooperative cognitive automobile framework, respectively.

Although these papers overcame some problems of the

*Corresponding author: e-mail: tripledg@postech.ac.kr

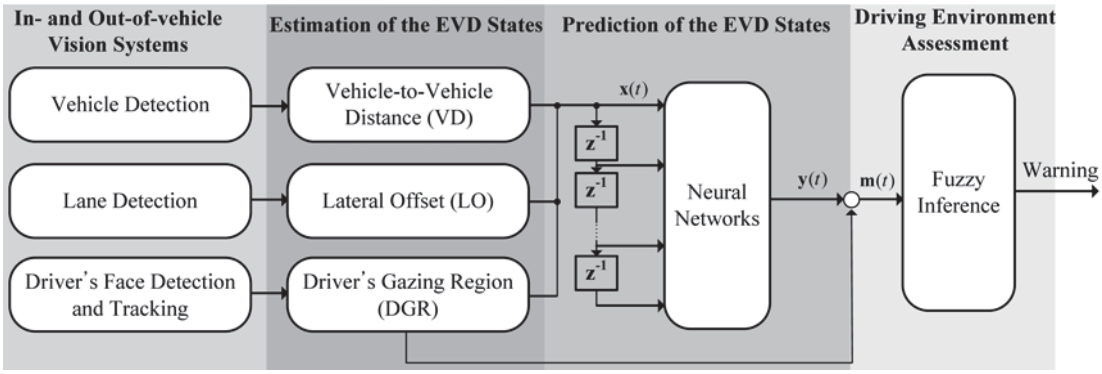

Figure 1. Overall system ($\mathbf{x}(t)$=[DGR(t) LO(t) VD(t)]^T, $\mathbf{y}(t)$=[TLC(t) TTC(t)]^T, $\mathbf{m}(t)$=[TLC(t) TTC(t) DGR(t)]^T).

previous ADAS, they used simple estimation metrics for EVD states such as the driver's head angle or the vehicle lateral offset and they developed single function systems such as the lane change intent analysis.

To improve the current limitation, this paper first defines three major EVD states from in- and out-of-vehicle vision systems:

(1) Traffic environment state: TTC estimated from the position and relative velocity of the front vehicle
(2) Vehicle state: TLC estimated from the lane position and lateral offset
(3) Driver state: Driver's gazing region estimated from the driver's gazing direction.

These states are estimated by enhanced detection and tracking methods from our in- and out-of-vehicle vision systems. Second, this paper proposes a long-term prediction method of the EVD states using a time delayed neural network to fuse these states, and a fuzzy inference system to assess the driving situation as shown in Figure 1.

2. IN-VEHICLE VISION SYSTEMS FOR ESTIMATING THE DRIVER'S GAZING REGION

Estimating the driver's gazing region consists of the two processes shown in Figure 2. First, the facial feature points are tracked and the 3D head pose (yaw, pitch, and roll angle) estimated by using the 2D+3D Active Appearance Model (AAM). Second, the driver's gazing region is found by projecting the gazing direction onto the defined frontal plane.

2.1. Tracking Facial Feature Points and Estimating 3D Head Pose Using 2D+3D AAM

AAM is a stochastic model which can be taught from large sample data. This can represent general facial shape and texture if the sample data represent faces. AAM includes both shape and texture models (Matthews and Baker, 2004). A 2D facial shape model consists of the mean shape and its principal components, including many shape feature points. A 2D facial texture model consists of the mean texture and its principal components. Here, the basic theory

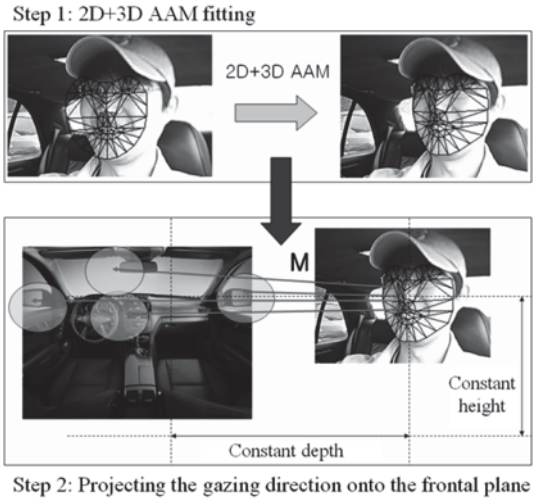

Figure 2. The proposed driver's gazing region estimation (**M**: 3D pose matrix).

of 2D+3D AAM is briefly introduced.

To make a 2D AAM, a database of facials shapes and textures is required. After obtaining the sample facial images under various poses and illumination conditions, 66 predefined facial shape feature points are marked manually on the images. Equations (1) and (2) represent the marked shape points as one dimensional vectors. The mean shape and principal components of the shape (shape bases) can be obtained through principal component analysis of the marked shape points (Equations (3)~(6)).

$$\mathbf{A}=[\mathbf{s}_1\ \mathbf{s}_2\ \cdots\ \mathbf{s}_N], \tag{1}$$

$$\mathbf{s}_i=\lfloor x_i^1\ y_i^1\ x_i^2\ y_i^2\ \cdots\ x_i^{66}\ y_i^{66}\rfloor, \tag{2}$$

where $\mathbf{A}$ is the shape data matrix, $\mathbf{s}_i$ is the i^{th} shape data vector, and N is the number of samples.

$$\mathbf{m}_s=\frac{1}{N}\sum_1^N \mathbf{s}_i, \tag{3}$$

$$\hat{\mathbf{A}}=[\mathbf{s}_1-\mathbf{m}_S\ \mathbf{s}_2-\mathbf{m}_s\ \cdots\ \mathbf{s}_N-\mathbf{m}_s], \tag{4}$$

$$\frac{1}{N}\hat{\mathbf{A}}\hat{\mathbf{A}}^T=\mathbf{U}_s\mathbf{D}_s\mathbf{U}_s^T, \text{ and} \tag{5}$$

$$\mathbf{U}_s=[\hat{\mathbf{s}}_1\ \hat{\mathbf{s}}_2\ \cdots\ \hat{\mathbf{s}}_m]\,, \tag{6}$$

where $\mathbf{m}_s$ is the mean shape vector, $\mathbf{U}_s$ is the matrix of eigen-vectors, and $\mathbf{D}_s$ is the matrix whose diagonal elements are eigen-values of the covariance matrix of shape data.

To make a 2D texture model, the image inside the shape feature points should be warped by an affine transformation from the original shape to the mean shape, as shown in Equations (7)~(12).

$$\mathbf{B}=[\mathbf{t}_1\ \mathbf{t}_2\ \cdots\ \mathbf{t}_N], \tag{7}$$

$$\mathbf{t}_i=\lfloor R_i^1\ G_i^1\ B_i^1\ R_i^2\ \cdots\ G_i^n\ B_i^n\rfloor\,, \tag{8}$$

$$\mathbf{m}_t=\frac{1}{N}\sum_1^N \mathbf{t}_i\,, \tag{9}$$

$$\hat{\mathbf{B}}=[\mathbf{t}_1-\mathbf{m}_t\ \mathbf{t}_2-\mathbf{m}_t\ \cdots\ \mathbf{t}_N-\mathbf{m}_t]\,, \tag{10}$$

$$\frac{1}{N}\hat{\mathbf{B}}\hat{\mathbf{B}}^T=\mathbf{U}_t\mathbf{D}_t\mathbf{U}_t^T\,,\ \text{and} \tag{11}$$

$$\mathbf{U}_t=[\hat{\mathbf{t}}_1\ \hat{\mathbf{t}}_2\ \cdots\ \hat{\mathbf{t}}_l]\,, \tag{12}$$

where $\mathbf{B}$ is the texture data matrix, $\mathbf{t}_i$ is the i^{th} texture data vector, n is the number of pixels, $\mathbf{m}_t$ is the mean texture vector, $\mathbf{U}_t$ is the matrix of eigen-vectors, and $\mathbf{D}_t$ is the matrix whose diagonal elements are eigen-values of the covariance matrix of texture data.

The overall procedure for making the 2D shape and texture model is depicted in Figure 3.

Once a 2D AAM is constructed, it is assumed that any facial shape ($\mathbf{s}$) and texture ($\mathbf{t}$) can be represented as the linear summation of mean ($\mathbf{m}_s$, $\mathbf{m}_t$) and principal components ($\hat{\mathbf{s}}_i$, $\hat{\mathbf{t}}_i$) (Equations (13) and (14)).

$$\mathbf{s}=\mathbf{m}_s+\sum_i \alpha_i\hat{\mathbf{s}}_i\,, \tag{13}$$

$$\mathbf{t}=\mathbf{m}_t+\sum_i \beta_i\hat{\mathbf{t}}_i\,, \tag{14}$$

where the coefficients α_i and β_i are the shape and texture parameters, respectively.

The 3D head pose cannot be obtained only with 2D AAM. Therefore, an additional 3D shape model must be generated by a 3D reconstruction method (Xiao *et al*., 2004). This method can recover 3D shape data from the 2D shape data. Then, an 3D shape model, which consists of 3D mean shape and principal components (3D shape bases), is constructed by principal component analysis. Like the 2D shape case, any 3D facial shape ($\mathbf{s}_{3D}$) can be represented as a linear summation of a 3D mean shape ($\mathbf{m}_{3D}$) and bases ($\mathbf{s}_i$) as in Equation (15).

$$\mathbf{s}_{3D}=\mathbf{m}_{3D}+\sum_i \gamma_i\hat{\mathbf{s}}_i\,, \tag{15}$$

where the coefficients γ_i are the 3D shape parameters.

The 3D head pose is found by minimizing the summation of the square errors between (i) the image produced by the 2D texture model and the 2D input image and (ii) the 2D input image and the projected 2D shape of the 3D shape (Xiao *et al*., 2004). The least squares solution is found via the second order AAM fitting method (Choi and Oh, 2006). This fitting process initializes the first face position using the Adaboost face detection technique (Viola and Jones, 2004).

2.2. Estimating the Driver's Gazing Region in the Frontal Plane

To estimate the driver's gazing region in the frontal plane, one must consider the 3D pose matrix ($\mathbf{M}$), the distance between the driver and frontal plane (d), and the height of driver's face (h) as shown in Figure 4. The problem was

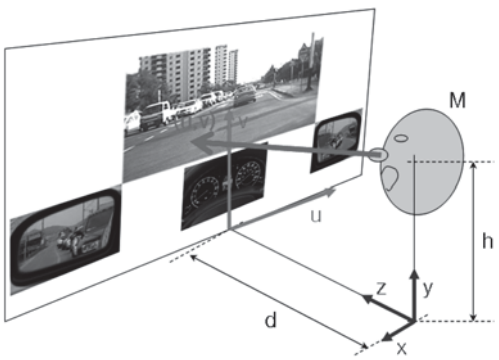

Figure 4. Projection of the driver's gaze onto the frontal plane.

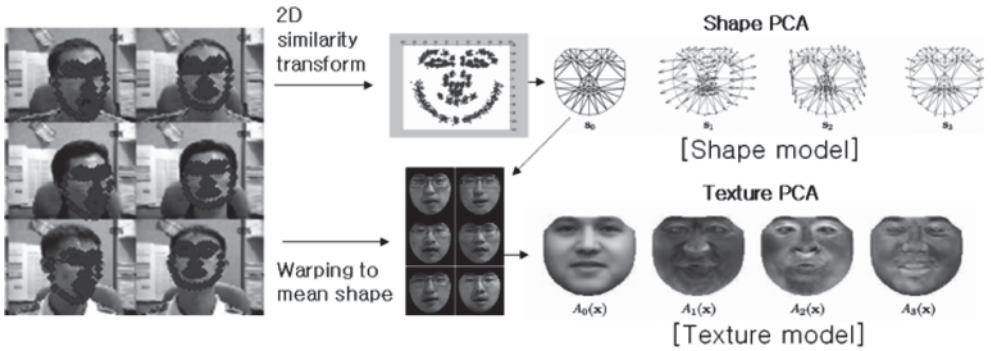

Figure 3. Construction of 2D AAM: shape and texture models (the images representing shape Principal Components Analysis (PCA) and texture PCA are adopted from Matthews and Baker (2004)).

simplified by assuming that the frontal plane is orthogonal to the Z-axis and that there is no translation of the driver's head (rotation only). These assumptions are valid because the driver's body and head remain on the seat and headrest most of the time.

In Figure 4, the frontal plane is represented by Equation (16) because the plane is orthogonal to the Z-axis and expanded through the point (0, 0, d).

$$(0,0,1)\cdot(x-0,\ y-0,\ z-d)=z-d=0. \tag{16}$$

The line of the driver's gaze is parallel to the directional cosine vector of **M** (direction of the driver's frontal face) and goes through the point (0, h, 0). Therefore, the gazing line is modeled as Equation (17). Here, the directional cosine vector of **M** is calculated by multiplying **M** to the unit vector along to the Z-axis as Equation (18).

$$\frac{x}{\cos\theta_x}=\frac{y-h}{\cos\theta_y}=\frac{z}{\cos\theta_z} \tag{17}$$

$$\begin{bmatrix}\cos\theta_x\\ \cos\theta_y\\ \cos\theta_z\end{bmatrix}=\mathbf{M}\times\begin{bmatrix}0\\ 0\\ 1\end{bmatrix} \tag{18}$$

Solving Equations (16) and (17) simultaneously gives the crossing point (u, v), which is both on the frontal plane and on the gazing line.

$$x=\frac{d\cos\theta_x}{\cos\theta_z},\ y=\frac{d\cos\theta_y}{\cos\theta_z}+h,\ z=d \tag{19}$$

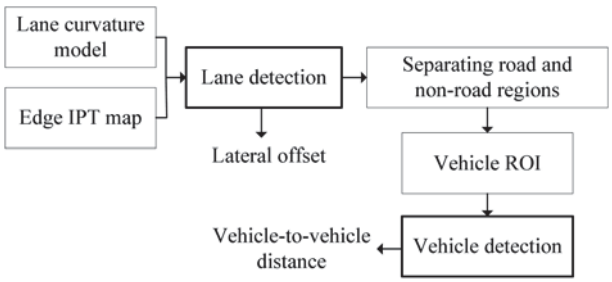

Figure 5. Lane and vehicle detection systems.

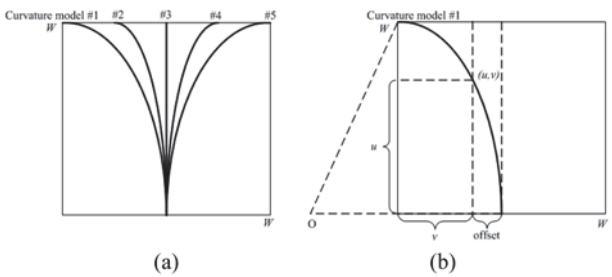

Figure 6. Estimation of the curvature model. (a) 5 lane curvature models. (b) Road coordinates and offset for transforming a curve model to a line (W: predefined width for a road ROI) (Above image for curvature models is adopted from He *et al*. (2004)).

$$(u,v)=(-x,y)=\left(-\frac{d\cos\theta_x}{\cos\theta_z},\ \frac{d\cos\theta_y}{\cos\theta_z}+h\right) \tag{20}$$

The frontal plane was divided into 7 different regions: frontal window (front, left, and right view), audio, cluster, and left and right side mirrors. Thus, the driver's gazing position in (u, v) coordinate can be correlated to the 7 regions on the frontal plane.

3. OUT-OF-VEHICLE VISION SYSTEMS FOR ANALYSING TRAFFIC ENVIRONMENT AND VEHICLE STATE

Out-of-vehicle vision systems mainly consist of lane and vehicle detection components, as shown in Figure 5. Lanes are detected by the difference between the predefined lane curvature models and an Inverse Perspective Transform (IPT) map. Vehicles are detected by first generating Regions of Interest (ROIs) for vehicles by separating road and non-road regions. Then, these ROIs are verified using the classifier designed for this purpose. Finally, lateral offset and vehicle-to-vehicle distance are estimated.

3.1. Lane Detection and Vehicle ROI Setting

Lane detection starts with the estimation of the lane curvature, as is done in He *et al*. (2004). Pre-defined curvature models (Figure 6(a)) and offsets are useful to transform a curve model to a line (Figure 6(b) and Equation (21)).

$$\text{offset}=a_i\left(\sqrt{(v+b_i)^2+u^2}-(v+b_i)\right)\ \text{ and } i\text{: curvature model number} \in [1,\ \ldots\ ,\ 5], \tag{21}$$

where a_i and b_i are transformation parameters for each curvature model (for example, a_1=1, b_1=W), and u and v are a point in the road coordinates. Here only 5 curvature models are used because 5 curvature models is all that is required to cover most lane curvatures.

IPT images for each curvature model (Figure 7(c)) are transformed from the edge image (Figure 7(b)), and the lane is the maximum of the vertical histogram for each IPT image (Figure 7(c)). Road and non-road regions are classified from initial road training samples acquired from the near region between both lanes (Figure 7(d)). The initial non-road training samples are acquired from above the vanishing point. After detecting or tracking vehicles, the road and non-road training samples are expanded to other regions using the known vehicles. A Classification and Regression Tree (CART) is used for on-line learning (Davis and Lienhart, 2006). The input feature of the CART is the RGB (Red, Green, Blue) color components. Although some vehicle colors may be similar to the road color, color in the shadows underneath the vehicle or rear lamps can distinguish the vehicle from the road as shown in Figure 7(e).

The vehicle ROI with the proper vehicle size (R) is determined by the probability of the non-road ($P_{nr}(R)$).

$w=k(r-hz)$, where

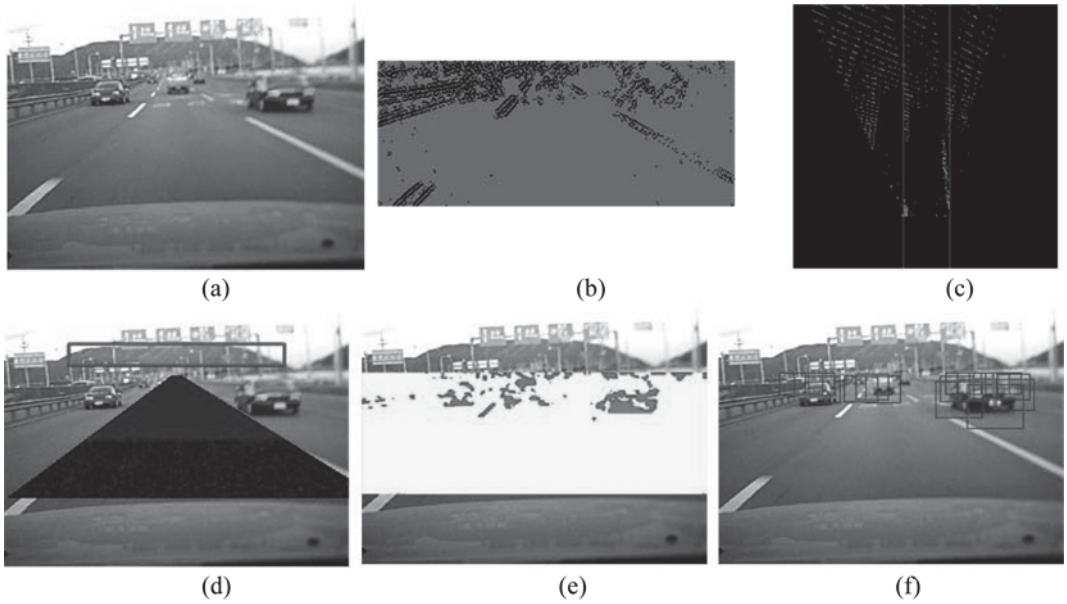

Figure 7. Extraction of vehicle ROIs: (a) Input image; (b) Edge image; (c) IPT image (transformed by curvature model #3); (d) Generation of road (blue dot) and non-road (red dot) training samples; (e) Road classification result (yellow: road region, magenta: non-road region); (f) Generation of vehicle ROIs.

$$k=\frac{\lambda^2 D}{H(\lambda^2+hz^2)}, \tag{22}$$

H is the height of the camera, hz is the vertical coordinate of the vanishing point in the image, λ is the focal length, r is the vertical coordinate of the image, and D is the predefined vehicle width.

$P_{nr}(R)$ is derived from the integrated $ii_n(x, y)$ for the non-road region from Equations (23) and (24) (Viola and Jones, 2004) as follows:

$$ii_n(x,y)=\sum_{x'\le x,y'\le y} i_n(x',y'), \tag{23}$$

where if $i_n(x',y')\in$ non-road region, $i_n(x',y')=1$, and otherwise, $i_n(x',y')=0$.

$$P_{nr}(R)=\frac{S_n(R)}{S(R)} \tag{24}$$

where $S(R)$: area of region R, and $S_n(R)=\sum_{x,y\in R} i_n(x,y)$.

If $P_{nr}(R)\ge 0.5$, region R may contain a non-road object as shown in Figure 7(f). This attention mechanism can effectively reduce the number of search windows and reduce false positive errors.

3.2. Vehicle Detection

Vehicles are detected via Support Vector Machine (SVM) and Scale Invariant Feature Transform (SIFT) (Lowe, 1999). First, SIFT features (keypoints) are extracted from ROI. The rich and distinct texture information of the vehicle is expressed by dividing a ROI into overlapping sub-regions as shown in Figure 8(a). The top region generally includes the rear-window, tail-lamps, etc., and the bottom region mainly includes the bumper, license plate, tires, etc. An input feature vector is composed of the integrated and normalized descriptor as shown in Figure 8(b).

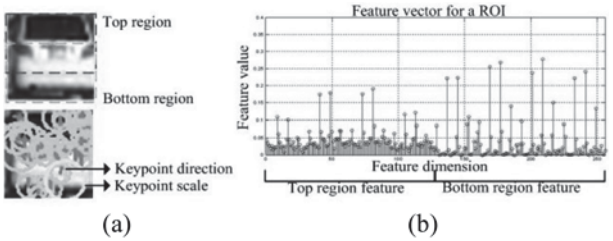

Figure 8. Generation of the input feature vector: (a) Keypoint extraction; (b) Input feature vector.

Figure 9. Examples of vehicle rear-view images used for training.

The training Database (DB) includes rear-view images of vehicles (Figure 9) and non-vehicles. Vehicle images include various vehicle types. Non-vehicle images include many false positive samples due to road repairs, road signs, guardrails, oil spills and shadows.

3.3. Parameter Estimation for Analyzing Traffic Environment and Vehicle State

Table 1 lists representative parameters which can analyze the traffic environment and host-vehicle state (EV).

The TLC and TTC parameters were used to consider the host-vehicle and traffic environment simultaneously.

TLC is estimated by considering the linear lateral velocity without the vehicle's yaw angle. The state vector $\mathbf{x}_{lat}(t)$ and measurement vector $\mathbf{z}_{lat}(t)$ for Kalman filter are

$$\mathbf{x}_{lat}(t)=[L_{lat}(t)\ V_{lat}(t)\ A_{lat}(t)]^{\mathrm{T}} \text{ and} \quad \mathbf{z}_{lat}(t)=L_{lat}(t), \tag{25}$$

where $L_{lat}(t)$, $V_{lat}(t)$, and $A_{lat}(t)$ are the lateral offset, vehicle lateral velocity, and its acceleration, respectively.

The transition matrix $\mathbf{M}_{lat}$ and measurement matrix $\mathbf{H}_{lat}$ are

$$\mathbf{M}_{lat}=\begin{bmatrix} 1 & T & 0.5T^2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \text{ and} \quad \mathbf{H}_{lat}=[1\ T\ 0.5T^2], \tag{26}$$

where T is the time step.

The covariance matrix of the process noise will be $Q_{lat}=Gq_{lat}G^T$, where the matrix G and q_{lat} are

$$G=\lfloor 0.5T^2\ T\ 1 \rfloor^T \text{ and} \quad q_{lat}=\sigma_A^2 \tag{27}$$

assuming a zero mean and white Gaussian process noise with variance σ_A^2 for the acceleration.

Similarly, the covariance matrix of the measurement noise is

$$R_{lat}=\sigma_L^2 \tag{28}$$

where σ_L^2 is the variance for the lateral offset.

For the short-term TTC estimation, the state vector and measurement vector are

$$\mathbf{x}_{long}(t)=[L_{long}(t)\ V_{long}(t)\ A_{long}(t)]^{\mathrm{T}} \text{ and} \quad \mathbf{z}_{TTC}(t)=L_{long}(t), \tag{29}$$

where $L_{long}(t)$, $V_{long}(t)$, and $A_{long}(t)$ are the vehicle-to-vehicle distance, the relative velocity, and the relative acceleration, respectively.

Other vectors and matrices for TTC estimation are similar to those for TLC.

Short-term TTC can be estimated as follows:

$$TTC(t)=\frac{L_{long}(t)}{V_{long}(t)} \tag{30}$$

Table 1. Representative parameters for EV.

Vehicle	Vehicle velocity Heading angle Yaw rate Steering angle	Lateral offset Lateral velocity Relative vehicle-to-vehicle velocity TLC TTC
Traffic environment	Lane position Lane curvature Lane type Vehicle-to-vehicle distance	

4. EVD ESTIMATION AND SITUATION ASSESSMENT USING NEURAL NETWORKS AND FUZZY INFERENCE

4.1. EVD States Estimation Using Neural Networks

Because EVD states are continuously coupled to each other in a non-linear way, it is difficult to estimate the long-term EVD states with conventional estimation methods such as KFs, Bayesian filters, HMMs, etc. Neural networks have the advantage of model-free learning, adaptation, and complex nonlinear mapping. Specifically, Time-Delayed Neural Networks (TDNN) (Mandic and Chambers, 2001) are

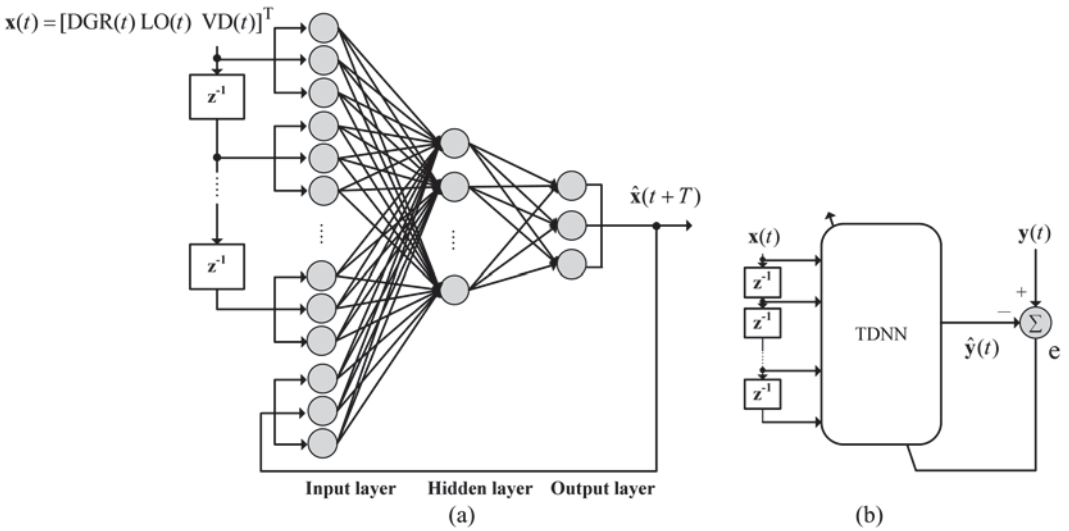

Figure 10. Applied TDNN architecture (a) and training configuration (b).

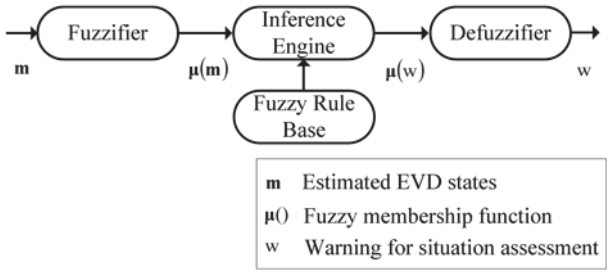

Figure 11. Used fuzzy inference systems.

powerful enough to capture the dynamics of the underlying nonlinear dynamic systems in a number of practical applications including speech recognition and time series prediction. Our system uses a TDNN with a multilayer perceptron architecture with time series inputs of lateral offset (LO), and with a vehicle-to-vehicle distance (VD) and driver's gazing region (DGR). The output is composed of the long-term EV states.

A training set for long-term TLC can be easily constructed from the real lane-crossing situation, but the ground truth for long-term TTC is not available. As the short-term TTC acquired from Section 3.3 is reliable, the training set for the long-term TTC is constructed by integrating the short-term TTC. Our training and test database included various real driving situations such as crossing lanes, approaching vehicles, and driver's visual inattention, as well as normal driving.

4.2. Fuzzy Inference for Situation Assessment

The current driving situation can be assessed from the estimated EVD states with a fuzzy inference system, because it can incorporate human knowledge. We use the Mamdani fuzzy inference systems (Mamdani and Assilian, 1975) as shown in Figure 11.

Fuzzy input and output Membership Functions (MFs) are empirically designed as shown in Figure 12. DGR is the singleton function that represents the specific gazing region and TDGR represents the gazing time duration for the specific region.

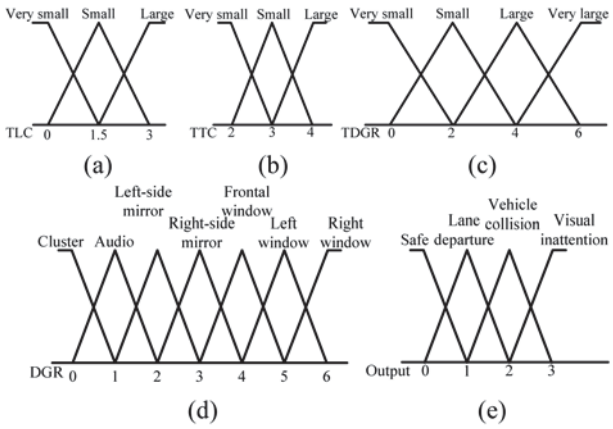

Figure 12. Fuzzy Membership Functions (MFs): (a) TLC MF; (b) TTC MF; (c) TDGR MF; (d) DGR MF; (e) Output MF.

Output MF for situation assessment consists of inattentive lane departure, inattentive vehicle collision, driver's visual inattention, and safe driving as shown in Figure 12(e).

Representative fuzzy if-then rules for driving situation assessment are as follows:

(1) *Rule for the inattentive lane departure warning:* If the TLC is very small and the DGR is not in the side mirror, then the inattentive lane departure warning should be given.
(2) *Rule for the inattentive collision warning:* If the TTC is very small and the DGR is not in the frontal window, then the inattentive vehicle collision warning should be given.
(3) *Rule for the visual inattention warning:* If the DGR maintains at the defined inattention regions (cluster, audio, left and right side mirrors) with a large TDGR, then the visual inattention warning should be given.
(4) *Rule for the safe driving:* If the TLC and TTC are large and the DGR remains in the appropriate attention regions, then the current driving situation is safe.

5. EXPERIMENTAL RESULTS

The proposed driving environment assessment system was tested in two main parts using real-driving data: EVD states estimation and situation assessment.

5.1. EVD States Estimation

The AAM model for driver state estimation was trained from 54 images of 3 people. The images included various head poses and illumination conditions.

The AAM model for driver state estimation was trained from 54 images of 3 people. The images included various head poses and illumination conditions. We marked 66 shape feature points on the images manually. Figure 13 shows sample images with the shape feature points marked. The

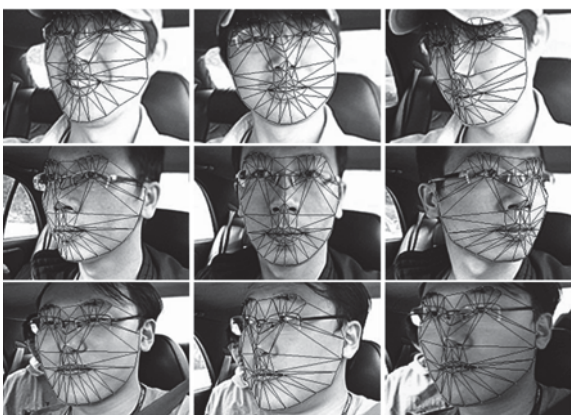

Figure 13. Some sample images for constructing 2D+3D AAM.

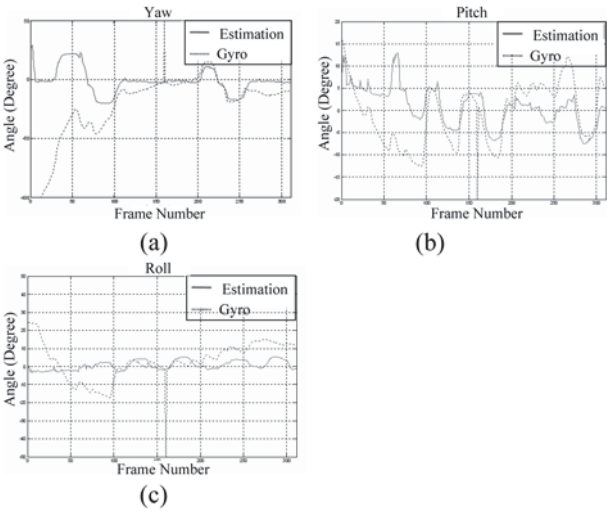

Figure 14. Pose tracking performance: (a) Yaw angle; (b) Pitch angle; (c) Roll angle.

constructed AAM consists of the mean shape, 8 shape bases, the mean texture, and 10 texture bases.

Test image sequences included various drivers' head positions while the height of the head and the distance between the driver and frontal plane were constant. Figure 14 shows the result of the driver's head pose estimation compared with the gyro sensor. After the gyro sensor stabilized (which occurred at about the 100th frame), the average error was below 3 degrees. Figure 15 shows the result of the driver's head pose estimation (left column) and the estimated gazing region on the frontal plane (right column). This result shows that our proposed method can not only correctly estimate the driver's gazing region on the frontal plane but also track the gazing direction in real time (above 15 *Hz*).

The proposed TDNN used 30 inputs (the current state + 9 time-delayed states), 2 hidden layers (30-20), and 2 outputs (TLC, TTC) to estimate the long-term EV states. It was trained with 2300 training samples and tested with 1800 other samples as shown in Figure 16.

Figures 17(a)~(c) show the driver's gazing region, lateral offset, and vehicle-to-vehicle distance, respectively. And Figures 17(d)~(e) show the estimation results of EV states using the proposed TDNN.

The performance is compared to a Kalman filter. Because the Kalman filter can't consider the driver state and

Figure 15. Experimental results of the driver's gazing region estimation; (a) Gazing at cluster: (b) Gazing at frontal window: (c) Gazing at left mirror: (d) Gazing at right mirror.

Figure 16. Database samples; (a) Normal safe driving case: (b) Intentional lane departure warning case: (c) Safe vehicle-to-vehicle distance maintenance: (d) Inattentive vehicle collision warning case.

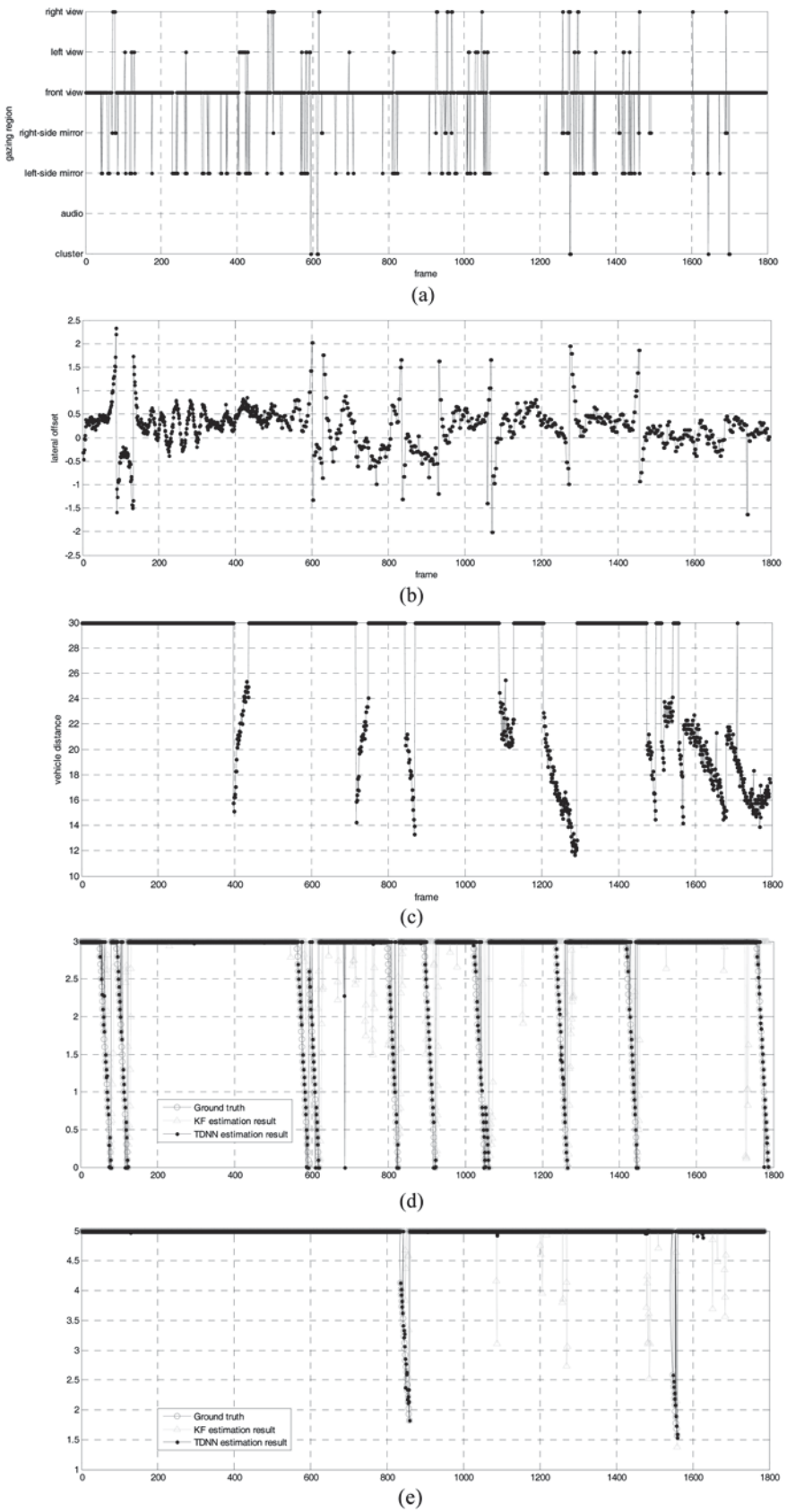

Figure 17. Long-term EV states estimation using TDNN: (a) Driver's gazing region; (b) Lateral offset; (c) Vehicle-to-vehicle distance; (d) Long-term TLC estimation result; (e) Long-term TTC estimation result.

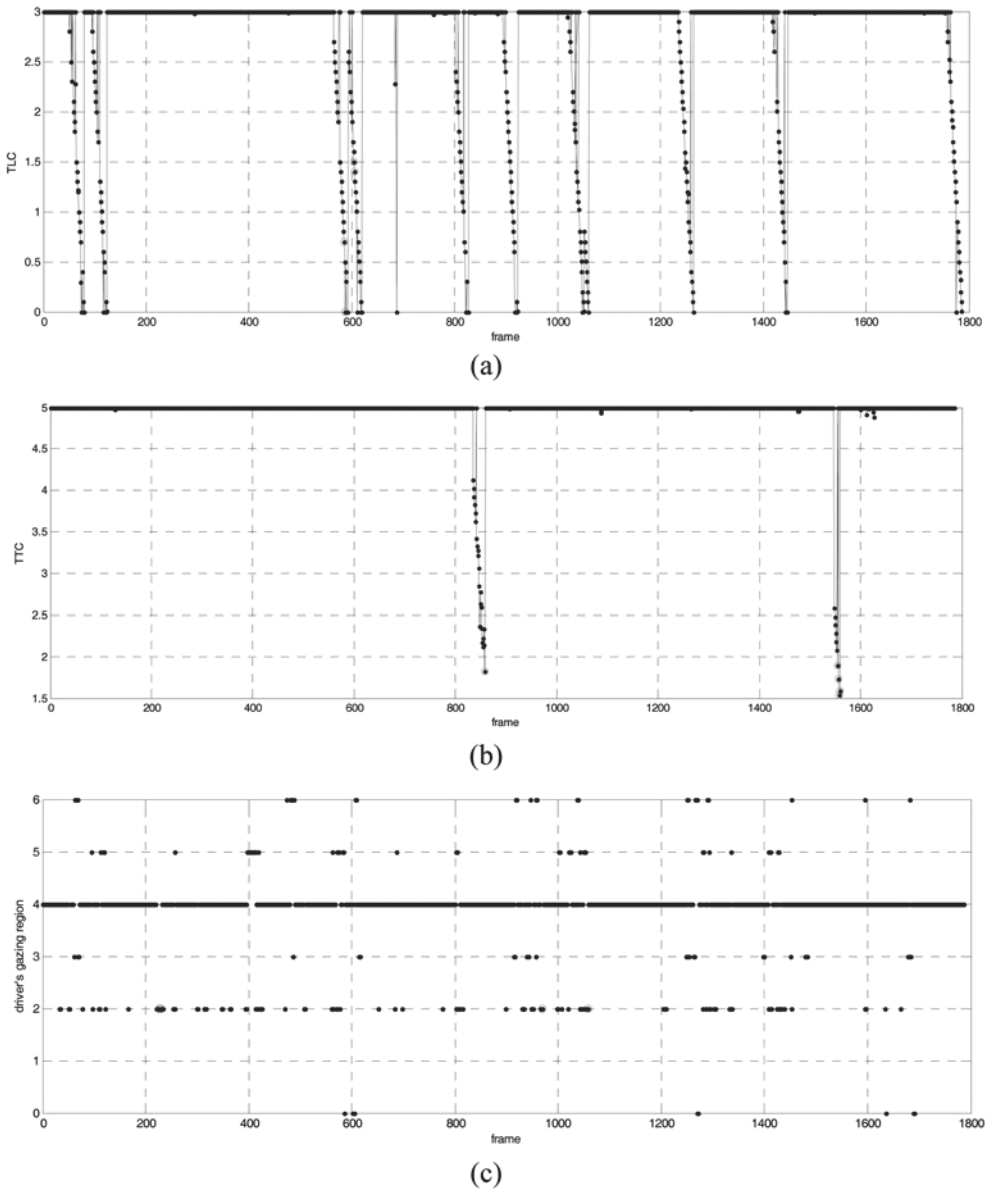

Figure 18. Situation assessment by fuzzy inference systems: (a) Inattentive lane departure warning case; (b) Inattentive vehicle collision warning case; (c) Driver's visual inattention warning case.

assumes linear relationships between EV states, the TLC and TTC estimation results have many false matches. But the TDNN estimation results follow the ground truth well. In the TLC estimation case, although the pattern of EV states is similar to the lane departure case (for example, between the 600th frame and the 800th frame), TDNN can estimate the TLC with small errors. In the TTC estimation case, the Kalman filter result is very sensitive to the vehicle-to-vehicle distance, while the proposed TDNN overcomes the problem by fusing the EVD states.

5.2. Situation Assessment by Fuzzy Inference Systems

Figure 18 shows the situation assessment by the fuzzy inference systems designed in this paper, where the red circles of each figure represent the predefined warning in Section 4.2. Although there are many lane departure cases in the test DB as shown in Figure 18(a), only 4 inattentive lane departure warnings are generated. Figure 18(b) shows two inattentive vehicle collision warnings, where the host-vehicle approached the front-vehicle and the driver had not gazed at the frontal window for a while.

Figure 18(c) shows that visual inattention warnings were generated when the driver gazed at the defined inattention regions for a large TDGR.

Because the host-vehicle approaches the front of the vehicle and the driver doesn't gaze at the front window for a while, inattentive vehicle collision warning cases are generated as shown in Figure 18(b). And because the driver

gazes the defined inattention regions with a large TDGR, visual inattention warning is generated as shown in Figure 18(c).

6. CONCLUSIONS

Compared to the previous systems which used limited states and gave many false warnings, the proposed driving situation assessment system can generate proper warnings and reduce the false warnings by using the following metrics:

(1) Improved EVD detection methods
   – Driver's gazing region estimation using 2D+3D AAM fitting and projection of the driver's gaze onto the frontal plane
   – Vehicle detection using SVM and on-line learning and the probability of non-road region
(2) Long-term state prediction: TDNN fuses EVD states and predicts long-term EV states from various real driving training DB.
(3) Multiple driving situation assessment: Fuzzy inference generates lane departure, vehicle collision, and visual inattention warning signals by using heuristic EVD MFs and fuzzy if-then rules.

This research focused on the EVD states from the forward area and assumed that the height of driver's face and the distance between the driver and frontal plan are constant. Future work should cluster general driver's gaze motion and analyze EVD states from all around the vehicle to extend this research to more general driver assistance systems.

ACKNOWLEDGEMENT–This work was supported by the Basic Research Program of MOST in Daegu Gyeongbuk Institute of Science and Technology (DGIST), Korea.

REFERENCES

Apostoloff, N. and Zelinsky, A. (2004). Vision in and out of vehicles: Integrated driver and road scene monitoring. *Int. J. Robot. Res.* **23**, **4–5**, 513–538.

Cheng, H., Zheng, N., Zhang, X., Qin, J. and Wetering, H. V. E. (2007). Interactive road situation analysis for driver assistance and safety warning systems: Framework and algorithms. *IEEE Trans. Intell. Transp.* **8**, **1**, 157–167.

Choi, H. C. and Oh, S. Y. (2006). Real-time recognition of facial expression using active appearance model with second order minimization and neural network. *Proc. IEEE Conf. Systems, Man, and Cybernetics,* 1559–1564.

Chung, T., Yi, S. and Yi, K. (2007). Estimation of vehicle state and road bank angle for driver assistance systems. *Int. J. Automotive Technology* **8**, **1**, 111–117.

Davies, B. and Lienhart, R. (2006). Using CART to segment road images. *Proc. SPIE Multimedia Content Analysis, Management, and Retrieval*, 60730U, 1–12.

Fletcher, L., Loy, G., Barnes, N. and Zelinsky, A. (2005). A correlating driver gaze with the road scene for driver assistance systems. *Robot. Auton. Syst.,* **52**, 71–84.

He, Y., Wang, H. and Zhang, B. (2004). Color-based road detection in urban traffic scenes. *IEEE Trans. Intell. Trans.* **5**, **4**, 309–318.

Kim, J. H., Kim, Y. W. and Sim, K. Y. (2007). Quantitative study on the fearfulness of human driver using vector quantization. *Int. J. Automotive Technology* **8**, **4**, 505–512.

Kim, S. Y., Kang, J. K., Oh, S. Y., Ryu, Y. W., Kim, K. S., Park, S. C. and Kim, J. W. (2008). An intelligent and integrated driver assistance system for increased safety and convenience based on all around sensing. *J. Intell. Robot. Syst.,* **51**, 261–287.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. *Proc. Int. Conf. Comput. Vision* 1150–1157.

Mamdani, E. H. and Assilian, S. (1975). An experiment in linguistic synthesis with a fuzzy logic controller. *Int. J. Man-Machine Studies* **7**, **1**, 1–13.

Mandic, D. P. and Chambers, J. A. (2001). *Recurrent Neural Networks for Prediction*. John Wiley. Chichester. New York.

Matthews, I. and Baker, S. (2004). Active appearance models revisited. *Int. J. Comput. Vision* **60**, **2**, 135–164.

McCall, J. C., Wipf, D. P., Trivedi, M. M. and Rao, B. D. (2007). Lane change intent analysis using robust operators and sparse Bayesian. *IEEE Trans. Intell. Transp.* **8, 3**, 431–440.

Stiller, C., Färber, G. and Kammel, S. (2007). Cooperative cognitive automobiles. *Proc. IEEE Intell. Veh. Symp.,* 215–220.

Viola, P. and Jones, M. J. (2004). Robust real-time face detection. *Int. J. Comput. Vision* **57**, **2**, 137–154.

Wu, Y.-J., Lian, F.-L, Huang, C.-P. and Chang, T.-H. (2007). Image processing techniques for lane-related information extraction and multi-vehicle detection in intelligent highway vehicles. *Int. J. Automotive Technology* **8**, **4**, 513–520.

Xiao, J., Baker, S., Matthews, I. and Kanade, T. (2004). Real-time combined 2D+3D active appearance models. *Proc. IEEE Conf. Comp. Vis. and Pattern Recog.* 535–542.