Multi-Classifier Feature Fusion-Based Road Detection for Connected Autonomous Vehicles

Prabu Subramani 1, Khalid Nazim Abdul Sattar 2, Rocio Pérez de Prado 3*, Balasubramanian Girirajan 4 and Marcin Wozniak 5

1 Department of Electronics and Communication Engineering, Mahendra Institute of Technology, Namakkal 637503, Tamil Nadu, India; vsprabu4u@gmail.com
2 Department of CSI, College of Science Majmaah University, Majmaah 11952, Saudi Arabia; k.sattar@mu.edu.sa
3 Telecommunication Engineering Department, University of Jaén, 23700 Linares, Jaén, Spain
4 Department of ECE, SR University, Warangal 506371, Telangana, India; girirajan.b@srut.edu.in
5 Faculty of Applied Mathematics, Silesian University of Technology, 44-100 Gliwice, Poland; marcin.wozniak@polsl.pl
* Correspondence: rperez@ujaen.es

Abstract: Connected autonomous vehicles (CAVs) currently promise cooperation between vehicles, providing abundant and real-time information through wireless communication technologies. In this paper, a two-level fusion of classifiers (TLFC) approach is proposed by using deep learning classifiers to perform accurate road detection (RD). The proposed TLFC-RD approach improves the classification by considering four key strategies such as cross fold operation at input and pre-processing using superpixel generation, adequate features, multi-classifier feature fusion and a deep learning classifier. Specifically, the road is classified as drivable and non-drivable areas by designing the TLFC using the deep learning classifiers, and the detected information using the TLFC-RD is exchanged between the autonomous vehicles for the ease of driving on the road. The TLFC-RD is analyzed in terms of its accuracy, sensitivity or recall, specificity, precision, F1-measure and max F measure. The TLFC- RD method is also evaluated compared to three existing methods: U-Net with the Domain Adaptation Model (DAM), Two-Scale Fully Convolutional Network (TFCN) and a cooperative machine learning approach (i.e., TAAUWN). Experimental results show that the accuracy of the TLFC-RD method for the Karlsruhe Institute of Technology and Toyota Technological Institute (KITTI) dataset is 99.12% higher than its competitors.

Keywords: accuracy; connected autonomous vehicles; deep learning classifiers; road detection; superpixel generation; two-level fusion of classifiers

1. Introduction

Autonomous vehicles are expected to accomplish an important role in the upcoming urban transport systems, because they offer high accessibility, improved productivity, extra safety, improved road efficiency and a positive effect on the environment [1,2]. The sensors involved in autonomous vehicles are utilized for two different purposes: environment perception, to identify the objects that exist around the vehicle, and localization, to detect where the vehicle is located on the road. Vehicle odometry, inertial measurement unit and global navigation satellite system sensors are used to achieve the localization of the autonomous vehicles. To enable environment perception, radar, camera and light detection and ranging (LiDAR) sensors are used for safety navigation. The autonomous vehicles take actions based on the outputs acquired from environment perception and localization [3,4]. In real time, the road status is determined by the communication between the autonomous vehicles [5]. In autonomous driving vehicles, robust road detection plays
an important role in achieving higher automation levels [6–8]. Specifically, the precise identification of the environment is essential to achieve safe driving on highways and complex inner-city roads [9,10].

Road detection is categorized into structured and unstructured types based on the scenic environment. On a structured road, the detection of the road is achieved using lane detection due to road edges or well-painted markings. However, an unstructured road without any edges and road markings causes issues while detecting the region of the road [11,12]. The following features affect the road detection performance: poor adaptability, variations in illumination, diverse road environments, obstacles and no versatile and low-cost approaches [13]. Moreover, existing road detection research has been fully supervised, which requires a huge amount of labeled training data for identification purposes. This is considered to be the main disadvantage when detecting roads [14]. However, developed road detection methods should achieve high accuracy to satisfy the requirements of autonomous driving [15]. In this paper, a two-level fusion of classifiers is presented to accomplish the precise road detection. The merits combined in the TLFC are given as follows: LeNet-5 extracts the internal features without excessive rounds of training, Long Short-Term Memory (LSTM) avoids the issues of gradient explosion and gradient disappearance, and the residual block used in the residual network (ResNet) is used to resolve the degradation issue while extracting the features. Further, the SVM uses the fused feature maps from the LeNet-5, LSTM and ResNet to achieve the better classification of roads.

The major contributions of this research paper are summarized as follows:

• The KITTI and Cambridge-Driving Labeled Video Database (CamVid) datasets are used in this TLFC-RD method to evaluate the performances of the approaches. Here, superpixel generation-based pre-processing is used in the images to group similar pixels into superpixels and thus minimize the classification complexity.

• The feature extraction methods used to detect the roads are the spatial values of pixels, the RGB value of pixels, entropy, HSV color space, texton features, the local distance distribution and local binary pattern (LBP). These different kinds of beneficial features are used to classify the images for extreme lighting conditions as well as in the shadow regions.

• Further, a two-level fusion of classifiers is used during the classification. At the first level, the LeNet-5, LSTM and ResNet are used to extract the feature maps from the feature vectors. Here, the TLFC uses multiple classifiers for feature map extraction, because the road scene images are acquired from a different perspective and angle. Then, the cross fold is used in the TLFC to create a different input image frame for each classifier. Additionally, this input frame (i.e., vehicles) is introduced to the deep learning classifiers, where it is randomly selected based on the vehicle’s preference.

• Subsequently, the SVM is used in the second level for the precise classification of the road into drivable and non-drivable areas based on the fused feature maps. Hence, the combination of the cross fold and TLFC is used to improve the classification accuracy.

The overall organization of this paper is as follows: Section 2 provides the existing works conducted on road detection. Section 3 provides the problem statement acquired from the related work, along with the solution provided by the TLFC-RD method. A clear explanation of the TLFC-RD method is provided in Section 4. Section 5 provides the performance and comparative analysis of the TLFC-RD method. Finally, the conclusion is drawn in Section 6.

2. Related Work

The existing road detection research along with its limitations are described in this section.
Yuan et al. [16] achieved road segmentation and lane recognition by using a normal map. Initially, depth information was used to create a normal map. Subsequently, a normal map was used to obtain the road pavement without any buildings and vehicles. The markings of the lane were improved based on the adaptive threshold segmentation method and denoising procedures. Moreover, the starting point of the lane was identified by integrating the vanishing point and using a Hough transform. The high-precision depth map was used to mitigate the interference of vehicles and buildings. However, this normal map-based road detection approach was only suitable for structured roads.

Dong et al. [17] developed the architecture of a deep network for detecting roads. The developed deep network was the combination of DAM and the network U-Net-prior; U-Net-prior was generally a modified segmentation network that integrated the shape and location before U-Net. Next, the DAM model was used to reduce the gap between the training and test images. The accuracy of the detection was enhanced based on the offline prior used in the detection. However, the variations in the environment created an impact on the performances of U-Net-DAM.

Yu et al. [18] presented the Two-Scale Fully Convolutional Network (TFCN) to identify the regions of roads. The deep learning model was enhanced by using the data acquired from the various scales. Here, the high-level semantic information and low-level details were fused using the fully convolutional layers and a skip-architecture in the two-scaled model. The multi-scale feature maps achieved from the various reception fields were used to identify the road areas. Moreover, the feature map was used to eliminate the redundancies of the scale information. This TFCN was analyzed only in the low-level information, and the lesser intersection over the union affected the accuracy of road detection.

Gu et al. [19] developed the hierarchical multi-feature road image segmentation framework (HMRF) to obtain road target data. A superpixel method, namely the Simple Linear Iterative Clustering (SLIC) algorithm, was used to obtain the local feature subregions from the image. The features obtained using the SLIC algorithm preserved the boundary information as it is used to minimize the incidence of over-segmentation. Next, the ensemble-learning random-forest method was used to classify the road environment. This HMRF failed to analyze larger datasets for road detection.

Alam et al. [20] presented the cooperative machine learning approach TAAWUN to achieve road detection for connected autonomous vehicles. TAAWUN was used as a binary classification, where TAAWUN was used to classify either the road or background class labels. Moreover, the developed TAAWUN has two variations: a hard majority vote (MV) and soft majority vote (MVP). In a challenging environment, the problem-specific feature sets were used to improve the accuracy of the TAAWUN method. TAAWUN was used with only a small number of features during detection as it provided the final results based on voting, which may lead to inappropriate classification.

Gu, S et al. [21] developed the LiDAR-camera fusion strategy to exploit color and range data during road detection. In this work, discrete 3D LiDAR points were converted into 2D LiDAR range images at the LiDAR. A faster road estimation was achieved by using the distance-aware height-difference based scanning method. Next, the Multi-Modal Conditional Random Field (MM-CRF) was used, fusing the binary and dense results of road detection. However, this MM-CRF was processed only in the urban road condition.

Yang, F et al. [22] presented an end-to-end road segmentation network, namely the Spatial Propagation and Spatial Transformation Fusion Network (SPSTFN), that considered the merits of the multi-modal data fusion and deep network. The coarse representation of the road was obtained by using an effective lightweight network. To detect the road, the bird view and perspective view were integrated in the deep network. However, the details of the image were constantly lost because of the continuous pooling operations.

3. Problem Statement
This section states the problems found in the related work along with the solutions provided by the proposed TLFC-RD method.

To improve performances, classification be performed for both structured and unstructured roads. However, normal map-based road detection has been analyzed only with structured roads [16]. Next, the variation in the environment leads to effects on the classification performances [17]. An inappropriate selection of features has a great impact on the performance of road detection [18,20]. Moreover, a voting-based classification may sometimes lead to inaccurate results while classifying the road as drivable or non-drivable areas.

**Solution**

In this TLFC-RD method, better classification is achieved even in the changing environments by extracting the optimal features from the images. For example, LBP is used in the feature extraction to extract the local texture of the image, even under heavy lighting conditions. Therefore, a variety of features such as spatial pixels, color features, texton and LBP features are used to extract the optimal features. Further, the deep learning classifier used in the TLFC achieves an effective classification of roads into drivable areas and non-drivable areas. Next, the classified information is transmitted between the connected vehicles to gain information about the environment.

**4. Proposed TLFC-RD Method**

In this TLFC-RD method, the two-level fusion of classifiers is used to predict the different classes of the road for CAVs. Here, the road is classified into two different classes: drivable area (i.e., road) and non-drivable area (i.e., background). These classifications are performed through the video frames obtained from the connected autonomous vehicles. There are four different classifiers used in this TLFC-RD method: LeNet-5, LSTM, ResNet and SVM. The classification of the road is improved by using the deep learning classifiers along with superpixel generation, appropriate features and multi-classifier feature fusion. Figure 1 illustrates the block diagram for the TLFC-RD method.

![Figure 1. Block diagram of the proposed TLFC-RD method.](image)

**4.1. Data Acquisition**

Generally, no public dataset is accessible or exists for CAVs in which the vehicles can share the information acquired from their respective visual sensors. Even if the datasets are available, the particular datasets are not free and require permission; furthermore, access is restricted only to certain organizations and countries. Hence, two different datasets, namely the KITTI dataset [23] and CamVid dataset [24], for road detection in a CAV are...
considered. From these datasets, images are acquired and preprocessed using superpixel generation. For example, the sample image from the KITTI dataset is shown in Figure 2.

![Sample image](image)

Figure 2. Sample image.

4.2. Preprocessing Using Superpixel Generation

In the TLFC-RD method, the superpixel method groups the identical pixels of the neighborhood into superpixel blocks. The similarity among pixels is calculated to group the identical pixels. This preprocessing method is used to minimize the redundant image information as well as preserve the boundary data of the images. Moreover, this superpixel generation reduces the difficulty in the subsequent road detection process. Here, simple linear iterative clustering is considered for the location and color of pixels in the neighborhood. The feature vector of the color image is transformed into a five-dimensional feature vector, Labxy, that contains a Lab color and two-dimensional planar space. Generally, the Lab is the color model in the CIELAB color space. The brightness is represented as $L$, two colors are represented as $ab$ and the spatial location in the flat image is denoted as $xy$. Here, the calculation of the similarity is accomplished by using the distance among the pixels. Therefore, the similarity among the pixels is computed, labels are assigned to the identical pixels and the overall algorithm is iterated until convergence.

Equations (1) and (2) shows the color difference and spatial distance among the pixels.

\[
d_{lab} = \sqrt{(l_k - l_i)^2 + (a_k - a_i)^2 + (b_k - b_i)^2}
\]

\[
d_{xy} = \sqrt{(x_k - x_i)^2 + (y_k - y_i)^2}
\]

where $d_{lab}$ and $d_{xy}$ represent the color difference and spatial distance among the pixels, respectively. The distance $D_s$ is calculated by using Equation (3).

\[
D_s = d_{lab}^2 + \frac{m^2}{s^2} d_{xy}^2
\]

where the step size among the pixels is represented as $s$ and the compactness parameter is denoted as $m$. This compactness parameter is utilized to define the relative proportion of various color and location distances. Each pixel in the image is allocated with a certain amount of superpixels. A processing alteration from pixels to superpixels using pre-processing is used to minimize the complexity during the classification. After pre-processing, the image is processed under feature extraction to extract the optimal features. The example for the pre-processed image is shown in Figure 3.
4.3. Feature Extraction from the Pre-Processed Image

The classification between the drivable and non-drivable areas is obtained by extracting appropriate features from the pre-processed image (PI). In this TLFC-RD method, seven appropriate feature extraction methods are used to extract the features. The utilized feature extraction methods are the spatial values of pixels, the RGB value of pixels, entropy, HSV color space, texton features, local distance distribution and LBP. Here, the features of the spatial value, RGB value and HSV color space are used in the TLFC to categorize the background from the shadows over the road. Specifically, the spatial value of the pixels is considered to identify the road from the background mainly for the sunlight region images. Next, the HSV value accomplishes an important role when the TLFC-RD is used for dark regions. The texton feature is used to define the local structure of the images and provides the features of intersections, corners and line terminators. However, the uncontrolled lighting conditions outdoors and unpredictable weather have a great impact on region detection. The local texture of images extracted using the LBP is used to overcome the aforementioned limitation. The feature extraction process is as follows:

(a) Spatial and RGB value of pixels

Initially, the spatial value of the pixels—i.e., x and y coordinates of pixels—are taken from the input image. Subsequently, the RGB values of the pixels are taken as feature sets from the image.

(b) Entropy

The information source that exists in the image is defined by using information or Shannon entropy. This entropy value also defines the global features of the source in an average manner. Similarly, the image entropy is calculated using the histogram of the image, because it effectively provides the complex degrees of grey value distribution. Equation (4) shows the entropy value of the input image \((I)\) by the size of \(M \times N\).

\[
E = - \sum_{i=0}^{l-1} p_i \log_2 p_i, p_i = \frac{n_i}{M \times N}
\] (4)

where the probability density function and number of pixels for the gray level \(i\) are \(p_i\) and \(n_i\) respectively.

(c) HSV color space

The main advantage of using HSV in road detection is that it is identical to the human conceptual understanding of colors. Moreover, it can divide achromatic and chromatic components. Here, the color is differentiated by using the hue \((H)\), the percentage of the white light included in the pure color is denoted as the saturation \((S)\) and the perceived light intensity is denoted as \((V)\). Equations (5)–(7) express \(H, S\) and \(V\), which are the components of the HSV color space.
In general, the texton features are the output acquired from the filter bank. Here, the filter bank has Gaussians in different scales such as $k$, $2k$ and $4k$. After converting the input image ($I$) into the Lab color space, the Gaussian filters are applied in the $R,G$ and $B$ channels. Subsequently, the 18-dimension vector of the texton feature ($TF$) is acquired from each pixel of the image.

The neighborhood space $\eta$ of a pixel is divided into the grid of $M \times N \times K$, which is a three-dimensional vector. Next, the distribution histogram ($DH_i$) for a point $i$ at $cell_t$ is expressed according to Equation (8):

$$DH_i = \sum_{k \in cells}D_{ik}$$

where $t = \{1,2,\ldots,M \times N \times K\}$ and $K$ is a third-dimensional vector. Hence, the resultant feature of the $M \times N \times K$ vector is $DH = \{DH_1, DH_2, \ldots, DH_{M \times N \times K}\}$.

In LBP, a neighborhood around each pixel is considered to generate a binary number for each pixel. A value of 1 is allocated for the pixels whose neighboring pixel’s intensity value is greater than or equal to 1 according to the center pixel. Otherwise, a value of zero is fixed for the respective pixels. Furthermore, the label values are rotationally placed together, and an eight-bit number is generated by using Equations (9) and (10).

$$LBP_{R,Q} = \sum_{q=0}^{Q} f(g_q - g_c) \times 2^q$$

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & x < 0 \end{cases}$$

where the neighbor radius and number of adjacent pixels for the center pixel are represented as $R$ and $Q$ respectively; the brightness intensities of the center and neighborhood pixel are denoted as $g_c$ and $g_q$ respectively.

4.4. Assumptions

Consider that $Z$ vehicles randomly exist in the road, and these vehicles are connected to share the data about the drivable and non-drivable area information. Thus, $Z$ feature vectors are generated during road detection. For example, the feature vector of a single vehicle extracted during detection is shown in Equation (11).

$$FV = \{XY, RGB, HSV, TF, DH, LBP\}$$

Each vehicle has its own preferred set of feature extraction methods. So, the remaining vehicles randomly select their feature vectors as $\{XY, RGB, HSV, TF, LBP\}$, $\{XY, RGB, HSV\}$ and $\{XY, RGB, HSV, DH, LBP\}$. The extracted feature vectors are processed under the two-level fusion of the classifiers to classify the precise class of the input image.
4.5. Two-Level Fusion of Classifiers for Road Detection

In this proposed method, the fusion of the classifiers is proposed to obtain the precise identification between the drivable area and the non-drivable area. There are four different classifiers used in this two-level fusion: LeNet-5, LSTM, ResNet and SVM. The input image frame acquired from the dataset is considered as a vehicle, and then a cross fold is applied to present this input frame to any of the chosen classifiers. Hence, there is no possibility that the particular input frame could be processed by only one classifier. Subsequently, each vehicle in the CAVs uses its preferred artificial intelligence to extract the feature maps from the feature vector ($FV$). Specifically, the vehicle in the CAV randomly selects any of the deep learning classifiers of LeNet-5, LSTM and ResNet. Next, the extracted feature maps are again given as an input to the SVM for the precise detection of the road. Therefore, the combination of the cross fold and TLFC is used to optimize the performances of road detection. The process of the two-level fusion of the classifier is explained in the following section.

4.5.1. LeNet-5

Generally, LeNet-5 is a gradient-based learning CNN that is applied to extract feature maps from a feature vector ($FV$). Except for the input and output layer, the LeNet-5 has six layers: three convolutional layers, two polling layers and one fully connected layer. The training of the parameters is reduced by minimizing the number of neurons in the fully connected layer. The process of LeNet-5 is as follows:

1. The convolutional layer is generally used to accomplish the feature extraction. In this layer, the input matrix is convolved with the convolution kernel. Let the feature vector be $FV = \{FV_{i,j} | i = 1, 2, ..., I, j = 1, 2, ..., J\}$, where $I$ represents the number of input images and $J$ represents the amount of data in the respective $FV$. Moreover, the convolutional kernel is represented as $W = \{w_{u,v} | u = 0,1,2, ... CKS, v = 0,1,2, ... CKS\}$, where the convolutional kernel’s size is denoted as $CKS$. Equation (12) expresses the convolutional layer output ($cl_{i,j}$),

$$cl_{i,j} = \left\{f \left( \sum_{u=0}^{CSK-1} \sum_{v=0}^{CSK-1} w_{u,v} FV_{i+u,j+v} + ot \right) \right\}$$

where the offset term considered in each convolution is $ot$ and the activation function is denoted as $f(.)$.

2. There are five different activation functions that are widely used: Gaussian, Rectified Linear Unit (ReLU), Softplus, Tanh and Sigmoid. In these activation functions, the ReLU does not have a gradient saturation issue as it is faster than the saturating non-linear functions. Therefore, the ReLU is considered in the CNN.

3. Next, the pooling layer is used to accomplish the feature selection to minimize the dimensions of the data, whereas the main features of the data are preserved at the same time. In the local accepted domain, the mean, random and larger values are extracted using the mean, random and maximum pooling in the pooling layer. The output of the pooling layer ($pl$) is expressed in Equation (13):

$$pl_{ln} = pool(pl_{ln-1})$$

where $ln$ represents the layer number and $pl_{ln-1}$ represents the former layer’s result.

4. In general, the fully connected layer is considered as the final layer of the CNN. The ReLU function is used in each neuron that is linked with the previous layer’s neuron. The local data are integrated into this layer and can be used to differentiate the classes. The output of the fully connected layer is expressed in Equation (14):

$$fc_{ln} = f(w_{ln} \cdot fc_{ln-1} + ot_{ln})$$
5. Further, multiple classifications are accomplished by using the output layer or softmax layer. Here, the softmax layer maps the output of the previous layer to (0,1). Each result is related to the classification probability and its sum is 1. Next, the output is chosen based on the classification of higher probability values. Equation (15) provides the output of the output layer \( o_t^{ln} \).

\[
o_t^{ln} = \text{softmax}(w^{ln} a_t^{ln-1} + o_t^{ln})
\]  

(15)

4.5.2. LSTM

LSTM [25] includes two activation functions and three gates, which are used to extract the feature maps from the feature vector (FV). The gates included in the LSTM are forgetting gates, input gates and output gates. Next, the long-term memory is included to create the black box of input and output. This leads to improve the training process of LSTM, therefore it helps to utilize the full historical sequence information. The result of the input gate \( (i_t) \), cell output \( (h_t) \) and output gate \( (o_t) \) for LSTM are expressed in Equations (16)–(18):

\[
i_t = \sigma(WM_i [h_{t-1}, FV] + ov_i)
\]  

(16)

\[
h_t = o_t \odot \text{tanh}(c_t)
\]  

(17)

\[
o_t = \sigma(WM_o [h_{t-1}, FV] + ov_o)
\]  

(18)

where the sigmoid activation function is represented as \( \sigma \), the weight matrix for the input and output gate is \( WM_i \) and \( WM_o \), the offset vectors/bias for the input and output gate are \( ov_i \) and \( ov_o \), memory is denoted as \( c_t \), and element-wise multiplication is denoted using \( \odot \). The output from the cell is used as a feature map for road detection in CAVs.

4.5.3. ResNet

ResNet [26], used in the TLFC, utilizes the residual block to solve the issues of gradient disappearance and degradation that exist in the convolutional neural network (CNN). The residual block used in the ResNet does not depend on the depth of the network, which also improves network performances. The integration of the input and output of the residual block is used to design the residual block in the ResNet.

The first layer’s activation is \( \text{ReLU} \), where \( \text{ReLU} \) is the residual, which is obtained after processing the linear transformation. In the second layer of ResNet, the FV is added to the residual value. The concentration of parameterized layers over the residual learning is obtained by using a direct connection channel between the input and output. The feature maps from the nonlinear function of ResNet are represented in Equation (19):

\[
\text{ReLU} = RW \cdot \delta(RW, FV)
\]  

(19)

where the residual block weight is represented as \( RW \) and the nonlinear function is denoted as \( \delta \).

4.5.4. Decision Fusion Using Multiple Classifiers

The proposed TLFC-RD method uses the two-level fusion of diverse classifiers, which is performed based on a different set of feature vectors. At first, multiple classifier training is used to create the feature sets. Here, the feature maps are generated from the feature vector obtained for each vehicle in the CAVs. In the first level, feature maps are extracted from the fully connected layers of LeNet-5, the cell gate of LSTM and a residual output from ResNet. The extracted feature maps are fused together and given as an input to the next-level classifier.

A multi-classifier feature fusion model [27] is used in this TLFC-RD, where the feature maps from the Lenet-5, LSTM and ResNet are used as inputs. This helps to improve
the classification accuracy during the road detection. The reason for using multi-classifier feature fusion is that the pooling layer used in the deep learning classifier eliminates some information while performing the classification. However, the multiple features obtained from the Lenet-5, LSTM and ResNet have the supervision information about road scenes, which is utilized to differentiate the classes as drivable and non-drivable data. As the road scenes obtained from the vehicles are captured from different perspectives and angles, the proposed multi-classifier feature fusion model is used to extract appropriate feature maps from the road scenes. Therefore, the feature maps from Lenet-5, LSTM and ResNet are concatenated to fuse the features as shown in Equation (20), and the fused information is given as an input to the second-level classifier (i.e., SVM).

$$FFM = \{fcl^{tn}, hl, Re\}$$

where $FFM$ represents the fused feature map vector.

4.5.5. SVM

In general, SVM [28] depends on the statistical learning theory and is used to perform road detection by using the $FFM$. The optimization issue of SVM is converted into a convex problem by using the radial basis kernel function. This is used to avoid the local minimum and obtains the classification of global optimization during the road detection of CAVs. Equation (21) shows the kernel function of the SVM:

$$K(FFM_i, FFM_j) = e^{-\frac{[(FFM_i - FFM_j)^2]}{2r^2}}$$

where $r$ is the radius, and $K(FFM_i, FFM_j)$ is additionally used in the next-level classification to detect the road. Here, the SVM is also used as a second-level classifier to predict a road by using the feature maps from the first-level classifiers. Figure 4 shows the architecture of the two-level fusion of the classifiers.

![Figure 4. Architecture of TLFC.](image-url)
Hence, the TLFC allows the better detection of roads with the KITTI and CamVid datasets. The performance of the TLFC-RD is mainly improved by the following four strategies: (i) the cross fold process at input and pre-processing using superpixel generation, which is used to minimize the complexity during the classification; (ii) an optimal feature extraction from the images is used to provide precise classification; (iii) multi-classifier feature fusion; and (iv) the TLFC-based classification of images provides better classification based on the generated feature maps.

5. Results and Discussion

The results and a discussion of the proposed TLFC-RD method are presented in this section. The MATLAB R2018a software was used to implement and simulate the TLFC-RD method to detect the road for CAVs. The simulation as carried out with the Windows 10 operating system with an i9 processor, 128 GB RAM and a 22 GB GPU. The TLFC-RD method was used to classify the visual information acquired from autonomous vehicles. More specifically, the TLFC-RD method was used to classify a road into drivable areas and non-drivable areas.

5.1. Dataset Description

To analyze the performance of the TLFC-RD method, the performance was analyzed in two different datasets: KITTI and CamVid datasets. The details of the datasets are as follows:

i. KITTI dataset

The KITTI dataset comprises synchronized images and LiDAR data with the ground-truth images, parameters of calibration and scripts for evaluation. This KITTI dataset has four types of road images: urban marked (UM), urban multiple marked lanes (UMM), urban unmarked (UU) and the combination of the three above (URBAN). There are two different types of images from the KITTI dataset that were considered for the analysis of the TLFC-RD method: extreme sunlight and shadow region.

ii. CamVid dataset

The CamVid dataset includes the collection of videos along with the object class semantic labels, and it has 32 different classes of videos such as sky, tunnel, archway, road, lane markings and so on.

5.2. Performance Metrics

The classification between the drivable area and non-drivable area using the TLFC-RD method was analyzed by using different metrics: accuracy, sensitivity or recall, specificity, precision, F1-measure and max F measure. The performance metrics are described as follows.

a. Accuracy

Accuracy represents the classification capacity between the different classes, and this accuracy (ACC) is expressed in Equation (22).

$$\text{ACC} = \frac{TP + TN}{TP + TN + FP + FN}$$  \hspace{1cm} (22)

b. Sensitivity or recall

Sensitivity or recall was used to calculate the capacity to search for the positive class, and the recall is expressed in Equation (23).

$$\text{Recall} = \frac{TP}{TP + FN}$$  \hspace{1cm} (23)

c. Specificity
Specificity represents the proportion of the negative class discovered during road detection, and this specificity is expressed in Equation (24).

\[
\text{Specificity} = \frac{TN}{TN + FP}
\]  

(24)

d. Precision

Precision (PRE) was used to calculate the capacity of the model for categorizing the positive class, and this precision is expressed in Equation (25).

\[
\text{PRE} = \frac{TP}{TP + FP}
\]  

(25)

e. F1-measure and max F value

The harmonic mean between the recall and precision is defined as the F1-measure, which is expressed in Equation (26). The selection of the classification threshold (β) was used to compute the max F, which is expressed in Equation (27).

\[
F1\text{-measure} = \frac{2 \times \text{PRE} \times \text{Recall}}{\text{PRE} + \text{Recall}}
\]  

(26)

\[
\text{Max} F = \arg \max_F F1\text{-measure}
\]  

(27)

5.3. Performance Analysis

Initially, the images from the dataset were acquired and considered as the video frames acquired from the vehicles between the CAVs. The images were processed under superpixel generation to minimize the complexity during the classification. Subsequently, the features from the images were extracted and a feature vector was generated for each vehicle of the CAVs. Finally, the classification of the drivable and non-drivable areas was accomplished by using the TLFC. For example, a preprocessed sample and detected image using the KITTI dataset are presented in Figures 5 and 6. Figures 5 and 6 also show a shadow region image and extreme sunlight image, respectively.

Figure 5. Road detection in shadow region scenario: (a) sample image, (b) pre-processed image, (c) detected image.
As the proposed TLFC-RD method uses deep learning classifiers for road detection, the performances of the proposed TLFC-RD method were compared with some deep learning classifiers: Lenet-5 and ResNet. On the other hand, the TLFC-RD method was also compared with three different conventional classifiers: the Artificial Neural Network (ANN), SVM and Random Forest Classifier (RFC). Therefore, the TLFC-RD method was evaluated with five different classifiers. Lenet-5, ResNet and the deep learning classifiers used in the TLFC-RD method use the same activation function, namely ReLU, whereas the conventional classifiers are operated with a default setup. Here, the performance metrics considered for the evaluation were accuracy, sensitivity or recall, specificity, precision, F-score and max F measure. The performance evaluation of the TLFC-RD method is described as follows.

The performance evaluation of the TLFC-RD method with ANN, SVM and RFC for KITTI and CamVid datasets is shown in Tables 1–3. Tables 1 and 2 show the performances of the KITTI dataset for extreme sunlight and shadow region images, respectively. Table 3 provides the performance evaluation of the TLFC-RD method with the CamVid dataset. From the evaluation, it could be concluded that the TLFC-RD method achieved better performance than the three existing classifiers (ANN, SVM and RFC). The fusion of feature maps from the LeNet 5, LSTM and SVM was used to improve the classification between the road and background for the autonomous vehicles. Additionally, the beneficial features from the images were used to improve the classification accuracy of the images. After performing the classification, we considered the detected information to be transmitted between the CAVs for ease of driving.

Figure 6. Road detection in extreme sunlight region scenario: (a) sample image, (b) pre-processed image, (c) detected image.
Table 1. Performance analysis of the TLFC-RD method for extreme sunlight images from the KITTI dataset.

| Method   | ACC (%) | Sensitivity/Recall (%) | Specificity (%) | PRE (%) | F1-Measure (%) | MaxF (%) |
|----------|---------|------------------------|-----------------|---------|----------------|----------|
| ANN      | 88.35   | 90.13                  | 92.12           | 90.43   | 91.35          | 93.21    |
| SVM      | 91.35   | 92.38                  | 93.04           | 91.76   | 92.98          | 93.04    |
| RFC      | 92.41   | 91.89                  | 92.04           | 92.43   | 91.08          | 94.21    |
| Lenet-5  | 96.75   | 95.48                  | 92.39           | 95.44   | 94.11          | 97.05    |
| ResNet   | 97.45   | 96.17                  | 95.76           | 95.83   | 96.79          | 97.58    |
| TLFC-RD  | 99.25   | 98.63                  | 96.37           | 97.45   | 98.33          | 97.27    |

Table 2. Performance analysis of the TLFC-RD method for shadow region images from the KITTI dataset.

| Method   | ACC (%) | Sensitivity/Recall (%) | Specificity (%) | PRE (%) | F1-Measure (%) | MaxF (%) |
|----------|---------|------------------------|-----------------|---------|----------------|----------|
| ANN      | 89.35   | 91.24                  | 91.53           | 89.25   | 92.14          | 92.75    |
| SVM      | 90.15   | 93.42                  | 92.56           | 90.15   | 93.25          | 94.03    |
| RFC      | 93.27   | 90.35                  | 93.92           | 93.52   | 92.64          | 93.28    |
| Lenet-5  | 95.18   | 94.27                  | 95.44           | 95.72   | 96.42          | 95.49    |
| ResNet   | 97.43   | 96.71                  | 96.33           | 97.04   | 96.94          | 97.03    |
| TLFC-RD  | 99.00   | 97.85                  | 97.24           | 98.76   | 99.00          | 96.19    |

Table 3. Performance analysis of the TLFC-RD method for CamVid dataset.

| Method   | ACC (%) | Sensitivity/Recall (%) | Specificity (%) | PRE (%) | F1-Measure (%) | MaxF (%) |
|----------|---------|------------------------|-----------------|---------|----------------|----------|
| ANN      | 90.12   | 91.43                  | 90.32           | 93.21   | 90.82          | 91.25    |
| SVM      | 91.24   | 92.15                  | 94.06           | 93.22   | 95.01          | 92.08    |
| RFC      | 92.96   | 94.02                  | 93.64           | 92.73   | 93.42          | 92.75    |
| Lenet-5  | 96.13   | 95.11                  | 95.88           | 96.04   | 95.61          | 96.25    |
| ResNet   | 97.81   | 96.25                  | 96.33           | 96.92   | 97.24          | 96.03    |
| TLFC-RD  | 99.24   | 98.72                  | 97.01           | 98.93   | 98.75          | 96.31    |

5.4. Comparative Analysis

The comparative analysis of the TLFC-RD method with existing research is presented in this section. The TLFC-RD method was compared with four different existing methods: U-Net-DAM [17], TFCN [18], TAAUWN-MV [20] and TAAUWN-MVP [20]. The aforementioned existing methods were chosen for comparison because all of these four methods use deep learning architectures for road detection. Similarly, the proposed TLFC-RD method also uses deep learning classifiers during road detection. Additionally, these existing methods process the road scenes with different light intensities in the same manner as the TLFC-RD method. The comparison was made by using two different datasets: KITTI and CamVid datasets. Here, the efficiency of the TLFC-RD method was evaluated by using k-fold cross-validation, where k = 10. Additionally, a unified activation function, ReLU, was considered to perform a fair comparison between the TLFC-RD method and existing methods. In that case, the existing methods—U-Net-DAM [17], TFCN [18], TAAUWN-MV [20] and TAAUWN-MVP [20]—were implemented and simulated using the same activation function.

Tables 4 and 5 show the performance comparison of the TLFC-RD method for the KITTI dataset and CamVid dataset, respectively, where NA represents the parameters that were not available for the existing methods. Moreover, the comparison for the KITTI dataset and CamVid dataset with a unified activation function is shown in Tables 6 and...
respectively. The accuracy comparison of the TLFC-RD method for the KITTI dataset with same activation function is shown in Figure 7. From the analysis, we can see that the TLFC-RD method achieves improved performance compared to U-Net-DAM [17], TFCN [18], TAAUWN-MV [20] and TAAUWN-MVP [20]. The reasons for the existing methods exhibiting poorer performance are as follows: the variations in the environment affected the performances of U-Net-DAM [17], while TFCN [18] and TAAUWN [20] achieved poorer performances because they used fewer features during the road detection. However, the performance of the TLFC-RD method was improved by using a two-level fusion of the classifiers, where deep learning methods (LeNet-5, LSTM and ResNet) were used to classify the images. Additionally, the TLFC-RD method was not affected by changes in the environment because it used appropriate features from the multi-classifier feature fusion model to improve the classification. The HSV accomplished an important role in dark regions where the image had large grey regions (shadows); furthermore, LBP was used to handle unpredictable weather and lighting conditions. The TLFC-RD method also provided better performance even when it presented with a huge amount of data. For example, the accuracy of TLFC-RD with augmentation was 99.64% for the KITTI dataset, which was higher than the existing methods. Meanwhile, the runtime of the TLFC-RD method was less compared to all the existing methods, because the deep learning classifiers used in the proposed method are independent of each other. Therefore, precise classification between the drivable and non-drivable areas was achieved by using the TLFC-RD. The detected information was transferred between the vehicles to share information about the road scenarios.

Table 4. Performance comparison of TLFC-RD method for the KITTI dataset.

| Performances     | U-Net-DAM [17] | TFCN [18] | TAAUWN-MV [20] | TAAUWN-MVP [20] | TLFC-RD (K-Fold) | TLFC-RD (with augmentation) |
|------------------|----------------|-----------|----------------|----------------|-----------------|-----------------------------|
| PRE (%)          | 94.86          | 97.71     | NA             | NA             | 98.10           | 99.01                       |
| Sensitivity/Recall (%) | 96.28      | 96.66     | 97.20          | 96.92          | 98.24           | 98.23                       |
| Specificity (%)  | NA             | NA        | 93.70          | 94.27          | 96.80           | 97.74                       |
| Max F (%)        | 95.57          | NA        | NA             | 96.25          | 99.12           | 99.64                       |
| Accuracy (%)     | NA             | 99.00     | 96.32          | 97.92          | 97.92           | 98.37                       |

Table 5. Performance comparison of TLFC-RD method for the CamVid dataset.

| Performances     | TFCN [18] | TLFC-RD (K-Fold) | TLFC-RD (with augmentation) |
|------------------|-----------|------------------|-----------------------------|
| PRE (%)          | 98.55     | 98.93            | 99.25                       |
| Sensitivity/Recall (%) | 98.44   | 98.72            | 99.04                       |
| Accuracy (%)     | 99.11     | 99.24            | 99.71                       |
| F1 (%)           | 98.50     | 98.75            | 98.93                       |
Table 6. Performance comparison of TLFC-RD method for the KITTI dataset with the same activation function.

| Performances      | U-Net-DAM [17] | TFCN [18] | TAAUWN-MV [20] | TAAUWN-MVP [20] | TLFC-RD (K-Fold) | TLFC-RD (with augmentation) |
|-------------------|----------------|-----------|----------------|-----------------|-----------------|-----------------------------|
| PRE (%)           | 93.42          | 97.21     | 96.07          | 96.42           | 98.10           | 99.01                       |
| Sensitivity/Recall (%) | 96.17        | 95.13     | 97.73          | 97.04           | 98.24           | 98.23                       |
| Specificity (%)   | 94.22          | 93.91     | 94.26          | 95.38           | 96.80           | 97.74                       |
| Max F (%)         | 95.01          | 95.08     | 95.11          | 94.91           | 96.23           | 98.11                       |
| Accuracy (%)      | 94.52          | 98.92     | 97.25          | 95.83           | 99.12           | 99.64                       |
| F1 (%)            | 93.55          | 96.20     | 96.14          | 96.11           | 97.92           | 98.37                       |
| Run time (s)      | 0.26           | 0.31      | 1.01           | 1.18            | 0.13            | 0.14                        |

Table 7. Performance comparison of TLFC-RD method for CamVid dataset with same activation function.

| Performances      | TFCN [18] | TLFC-RD (K-Fold) | TLFC-RD (with augmentation) |
|-------------------|-----------|------------------|-----------------------------|
| PRE (%)           | 98.21     | 98.93            | 99.25                       |
| Sensitivity/Recall (%) | 97.03        | 98.72            | 99.04                       |
| Specificity (%)   | 96.49     | 97.01            | 98.23                       |
| Max F (%)         | 96.87     | 96.31            | 99.01                       |
| Accuracy (%)      | 98.27     | 99.24            | 99.71                       |
| F1 (%)            | 97.92     | 98.75            | 98.93                       |
| Run time (s)      | 0.15      | 0.10             | 0.12                        |

Figure 7. Accuracy comparison for TLFC-RD with U-Net-DAM [17], TFCN [18], TAAUWN-MV [20] and TAAUWN-MVP [20].

6. Conclusions

CAVs are supposed to be a key aspect of future mobility, and the information passed by the CAVs is used to enhance the response, efficiency and comfort of the drivers. In this paper, the TLFC-RD method and the multi-classifier feature fusion-based classification of a road into drivable and non-drivable areas is achieved to determine the information about the road environment. The cross fold process at the input and the process of alterations from pixels to superpixels using pre-processing are used to minimize the difficulty while classifying the roads. Here, the optimal features from the images are obtained by using different feature extraction methods such as the spatial values of pixels, the RGB value of pixels, entropy, HSV color space, texton features, local distance distribution and LBP. Subsequently, the LeNet-5, LSTM and ResNet used in the TLFC are used to extract
the feature maps, and SVM is used to classify the road and background in extreme sunlight and shadow image regions. Accordingly, the information about the environment is detected and is shared between the CAVs. From the performance analysis, we can conclude that the performance of the TLFC-RD method provides better performance than the ANN, SVM, RFC, U-Net-DAM, TFCN, TAAUWN-MV and TAAUWN-MVP. The accuracy of the TLFC-RD method for the KITTI dataset is 99.12%, which is high when compared to the TFCN and TAAUWN methods. In the future, the objects that exist in the drivable area can be identified by using deep learning classifiers.

**Author Contributions:** The paper investigation, resources, data curation, writing—original draft preparation, writing—review and editing and visualization were performed by P.S. and K.N.A.S. The paper conceptualization, software, validation and formal analysis were performed by B.G. Methodology, supervision, project administration and final approval of the version to be published were conducted by R.P.d.P. and M.W. All authors have read and agreed to the published version of the manuscript.

**Funding:** The authors acknowledge contribution to this project from the Rector of the Silesian University of Technology under a proquality grant no. 09/020/RGJ21/0007.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data presented in this study are openly available in KITTI dataset at doi:10.1177/0278364913491297, reference number [23] and CamVid dataset at doi:10.1016/j.patrec.2008.04.005, reference number [24].

**Conflicts of Interest:** The authors declare no conflict of interest.

**References**

1. Pendleton, S.D.; Andersen, H.; Du, X.; Shen, X.; Meghjani, M.; Eng, Y.H.; Rus, D.; Ang, M. Perception, Planning, Control, and Coordination for Autonomous Vehicles. *Machines* 2017, 5, 6, doi:10.3390/machines5010006.

2. Li, L.; Ota, K.; Dong, M. Humanlike Driving: Empirical Decision-Making System for Autonomous Vehicles. *IEEE Trans. Veh. Technol.* 2018, 67, 6814–6823, doi:10.1109/tvt.2018.2822762.

3. Jahromi, B.S.; Tulabandhula, T.; Cetin, S. Real-Time Hybrid Multi-Sensor Fusion Framework for Perception in Autonomous Vehicles. *Sensors* 2019, 19, 4357, doi:10.3390/s19204357.

4. Xu, F.; Chen, L.; Lou, J.; Ren, M. A real-time road detection method based on reorganized lidar data. *PLoS ONE* 2019, 14, e0215159, doi:10.1371/journal.pone.0215159.

5. Balado, J.; Martínez-Sánchez, J.; Arias, P.; Novo, A. Road Environment Semantic Segmentation with Deep Learning from MLS Point Cloud Data. *Sensors* 2019, 19, 3466, doi:10.3390/s19163466.

6. Wang, K.; Yan, F.; Zou, B.; Tang, L.; Yuan, Q.; Lv, C. Occlusion-Free Road Segmentation Leveraging Semantics for Autonomous Vehicles. *Sensors* 2019, 19, 4711, doi:10.3390/s19124711.

7. Caltagirone, L.; Bellone, M.; Svensson, L.; Wahde, M. LIDAR–camera fusion for road detection using fully convolutional neural networks. *Robot. Auton. Syst.* 2019, 111, 125–131, doi:10.1016/j.robot.2018.11.002.

8. Xu, F.; Hu, B.; Chen, L.; Wang, H.; Xia, Q.; Sehdev, P.; Ren, M. An illumination robust road detection method based on color names and geometric information. *Cogn. Syst. Res.* 2018, 52, 240–250, doi:10.1016/j.cogsys.2018.06.019.

9. Byun, J.; Seo, B.; Lee, J. Toward Accurate Road Detection in Challenging Environments Using 3D Point Clouds. *ETRI J.* 2015, 37, 606–616, doi:10.4218/etrij.15.0113.1131.

10. Liu, H.; Han, X.; Li, X.; Yao, Y.; Huang, P.; Tang, Z. Deep representation learning for road detection using Siamese network. *Multimed. Tools Appl.* 2019, 78, 24269–24283, doi:10.1007/s11042-018-6986-1.

11. Li, Y.; Ding, W.; Zhang, X.; Ju, Z. Road detection algorithm for Autonomous Navigation Systems based on dark channel prior and vanishing point in complex road scenes. *Robot. Auton. Syst.* 2018, 85, 1–11, doi:10.1016/j.robot.2016.08.003.

12. Xiao, L.; Wang, R.; Dai, B.; Fang, Y.; Liu, D.; Wu, T. Hybrid conditional random field based camera-LIDAR fusion for road detection. *Inf. Sci.* 2018, 432, 543–558, doi:10.1016/j.ins.2017.04.048.

13. Li, J.; Shi, X.; Wang, J.; Yan, M. Adaptive road detection method combining lane line and obstacle boundary. *IET Image Process.* 2020, 14, 2216–2226, doi:10.1049/iet-ipr.2018.6433.

14. Han, X.; Lu, J.; Zhao, C.; You, S.; Li, H. Semisupervised and Weakly Supervised Road Detection Based on Generative Adversarial Networks. *IEEE Signal Process. Lett.* 2018, 25, 551–555, doi:10.1109/lsp.2018.2809685.

15. Zhang, X.; Yang, W.; Tang, X.; Liu, J. A Fast Learning Method for Accurate and Robust Lane Detection Using Two-Stage Feature Extraction with YOLO v3. *Sensors* 2018, 18, 4308, doi:10.3390/s18124308.
16. Yuan, C.; Chen, H.; Liu, J.; Zhu, D.; Xu, Y. Robust Lane Detection for Complicated Road Environment Based on Normal Map. *IEEE Access* 2018, 6, 49679–49689, doi:10.1109/access.2018.2868976.

17. Dong, M.; Zhao, X.; Fan, X.; Shen, C.; Liu, Z. Combination of modified U-Net and domain adaptation for road detection. *IET Image Process.* 2019, 13, 2735–2743, doi:10.1049/iet-ipr.2018.6696.

18. Yu, D.; Hu, X.; Liang, K. A two-scaled fully convolutional learning network for road detection. *IET Image Process.* 2021, doi:10.1049/iet-pr.2021.12157.

19. Gu, Y.; Si, B.; Liu, B. A Novel Hierarchical Model in Ensemble Environment for Road Detection Application. *Remote Sens.* 2021, 13, 1213, doi:10.3390/rs13061213.

20. Alam, F.; Mehmood, R.; Katib, I.; Altowaijri, S.M.; Albeshri, A. TAAWUN: A Decision Fusion and Feature Specific Road Detection Approach for Connected Autonomous Vehicles. *Mob. Netw. Appl.* 2019, 1–17, doi:10.1007/s11036-019-01319-2.

21. Gu, S.; Zhang, Y.; Tang, J.; Yang, J.; Alvarez, J.M.; Kong, H. Integrating Dense LiDAR-Camera Road Detection Maps by a Multi-Modal CRF Model. *IEEE Trans. Veh. Technol.* 2019, 68, 11635–11645.

22. Yang, F.; Wang, H.; Jin, Z. A fusion network for road detection via spatial propagation and spatial transformation. *Pattern Recognit.* 2020, 100, 107141, doi:10.1016/j.patcog.2019.107141.

23. Geiger, A.; Lenz, P.; Stiller, C.; Urtasun, R. Vision meets robotics: The KITTI dataset. *Int. J. Robot. Res.* 2013, 32, 1231–1237, doi:10.1177/0278364913491297.

24. Brostow, G.J.; Fauqueur, J.; Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognit. Lett.* 2009, 30, 88–97, doi:10.1016/j.patrec.2008.04.005.

25. Khalil, K.; Eldash, O.; Kumar, A.; Bayoumi, M. Economic LSTM Approach for Recurrent Neural Networks. *IEEE Trans. Circuits Syst. II Express Briefs* 2019, 66, 1885–1889, doi:10.1109/tcissii.2019.2924663.

26. Wang, A.; Wang, M.; Wu, H.; Jiang, K.; Iwahori, Y. A Novel LiDAR Data Classification Algorithm Combined CapsNet with ResNet. *Sensors* 2020, 20, 1151, doi:10.3390/s20041151.

27. Zhang, J.; Yan, X.; Cheng, Z.; Shen, X. A face recognition algorithm based on feature fusion. *Concurr. Comput. Pract. Exp.* 2020, p.e5748, e5748, doi:10.1002/cpe.5748.

28. Ebrahimi, M.; Khoshtaghaza, M.; Minaei, S.; Jamshidi, B. Vision-based pest detection based on SVM classification method. *Comput. Electron. Agric.* 2017, 137, 52–58, doi:10.1016/j.compag.2017.03.016.