Ice Detection on Edge Device Based on Most Significant Digit First SVM

Mohammadreza Najafi
Department of Computer Engineering, Chosun University, Gwangju, Republic of Korea
najafi@chosun.kr

Saeid Gorgin
Department of Computer Engineering, Chosun University, Gwangju, Republic of Korea and Department of Electrical Engineering and Information Technology, IROST, Tehran, Iran
gorgin@chosun.ac.kr

Danial Javaheri
Department of Computer Engineering, Chosun University, Gwangju, Republic of Korea
javaheri@chosun.ac.kr

Jeong-A Lee
Department of Computer Engineering, Chosun University, Gwangju, Republic of Korea
jalee@chosun.ac.kr

ABSTRACT

Abstract- The present notion of smart cities inspires urban planners and academics to provide citizens with modern, secure, and sustainable infrastructure and a decent standard of living. To meet this demand, video surveillance cameras have been installed to improve the safety and well-being of the populace. Despite technological advances in modern science, detecting anomalous events in surveillance video systems is difficult and takes extensive human effort. Due to the context-dependent nature of the anomalous concept, we identify several objects of interest and freely accessible datasets for anomaly detection. Iced surface in highways is one of the serious anomaly which causes a large number of accidents on the highway every year. CCTVs installed on highways can be a good tool to detect iced surface. Detection is a time-sensitive application of computer vision; hence, this paper has concentrated on investigating iced surface detection using edge devices and methods. We have leveraged Most-Significant Digit First arithmetic to improve the performance and resource utilization of the Support Vector Machine. We have applied our proposed method to address the problem of ice detection on highways, where the experimental results indicated a significant improvement in terms of accuracy, speedup, and energy consumption. These metrics are essential for edge computing and real-time intelligent surveillance applications.

CCS CONCEPTS

• **Computing methodologies**; • **Artificial intelligence**; • **Computer vision**; • **Object detection**;

KEYWORDS

FPGA, Machine learning, Computer arithmetic, Computer vision

ICVIP 2022, December 23–26, 2022, Shanghai, China

ACM ISBN 978-1-4503-9756-8/22/12…$15.00
https://doi.org/10.1145/3579109.3579120

ACM Reference Format:
Mohammadreza Najafi, Saeid Gorgin, Danial Javaheri, and Jeong-A Lee. 2022. Ice Detection on Edge Device Based on Most Significant Digit First SVM. In *2022 The 6th International Conference on Video and Image Processing (ICVIP 2022), December 23–26, 2022, Shanghai, China.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3579109.3579120

1 INTRODUCTION

In recent years, machine learning (ML) and computer vision (CV) have developed and indicated a significant role in addressing many severe problems in various domains and applications [1]. With the aid of processing and analysis, CV is able to comprehend things at a deep level and gather data. Additionally, these systems are made to automate some activities that the human visual system does. Automatic Inspection, Modeling Objects, Controlling Processes, Navigation, Video Surveillance, and many more multidisciplinary domains employ CV. A major use of CV is video surveillance, which is utilized in most public and private settings for observation and monitoring. Intelligent video surveillance systems are employed to identify, monitor, and analyze things at a high level without human intervention. Depending on the user's preferences, these intelligent video surveillance systems are employed in homes, workplaces, hospitals, malls, and parking lots. In order to improve public safety, surveillance cameras are being utilized more often in public spaces, including roadways, junctions, banks, and shopping centers. Law enforcement organizations' monitoring capacities, however, have not kept up. As a result, there is a conspicuous gap in the use of surveillance cameras, and the ratio of cameras to human monitors is untenable. Identifying abnormal occurrences, such as traffic accidents, crimes, or unlawful activity, is a crucial duty in video surveillance. As opposed to usual activity, anomalous events often happen far less frequently. Therefore, there is an urgent need to create clever computer vision algorithms for automatic video anomaly identification in order to reduce labor and time loss. One of the anomaly events in the highway is the icing surface which causes the accident. The decrease in friction caused by ice accumulation on highways creates dangerous circumstances. Forecasting, spotting, and avoiding ice development are of utmost significance for those responsible for winter road maintenance. There are sophisticated sensors for detecting road ice, but more affordable methods are needed. The methodology appears to promise a low-cost approach appropriate for a number of cold-climate applications where ice formation detection is crucial.

2 PRELIMINARIES

We have considered describing our design for ICED or NOT-ICED surface classification. Since iced surface detection based on neural networks [2] has a large feature size analysis model, it can be computationally intensive for running on embedded systems and edge devices, which have limited resources in terms of power supply and processing elements. Whereas the SVM has offered many advantages, including being effective in high dimensional spaces and cases where the number of dimensions is higher than samples, it severely suffers from a computationally intensive nature with a time-complexity of O(n^3). SVM is one of the well-established algorithms in machine learning in various domains, such as face detection [3], text and hypertext categorization to classify documents into different groups [4], and image classification. SVM provides better search accuracy for image classification compared to traditional query-based searching techniques and even artificial neural networks in some cases [5]. One of the main pros of the support vector machine method, which makes it very popular, is the ability to solve various problems without losing accuracy by enhancing the size and dimension of the problem.

2.1 Support Vector Machine Internals

The SVM is a supervised classifier that finds hyperplanes in n-dimensions space to perform non-overlapping segmentation or regression utilizing all features. It allows a global classification model based on linear discrimination with maximum margin, not considering dependencies between attributes. Intuitively, the hyperplane that best separates data points is the case where the distance to the nearest training data is the largest.To determine the label (i.e., Y_{X_t}) of test data, Equation 1, or its expanded form in Equation 2, is used. In this equation, if the value of $f(X_t)$ is negative, the Sign function returns the value of -1 as the label; otherwise, if the value of $f(X_t)$ is positive, the label would be +1.

$$Y_{X_t} = \text{Sign}\left(f\left(X_t\right)\right) \tag{1}$$

$$f\left(X_t\right) = \left(\sum_{i=1}^{n}\left(y_i \alpha_i K\left(X_t, x_i\right)\right) + b\right) \tag{2}$$

$$K\left(X_t, x_i\right) = X_t \cdot x_i \tag{3}$$

In Equation 2, x_i is the train data, X_t is the test data, and y_i indicates the label of each train data point. The values of α_i and b are calculated from the training phase and $K(X_t, x_i)$ is the kernel function, which is shown by Equation 3.

2.2 Most Significant Digit First Arithmetic

Online arithmetic is a type of serial computation where the most significant digits are processed first. Indeed, contrary to conventional arithmetic, the result is generated from the right (the least significant positions) to the left (the most significant positions); hence, it is also called Most Significant Digit First (MSDF) arithmetic. In Online arithmetic, the result of the most significant position is made available first. In this computational manner, the result digits are produced after receiving a limited number of digits (i.e., online delay δ), while the remaining parts of inputs are coming, as shown in Figure 1 (Left chart). Therefore, as illustrated in Figure 1 (Right chart), the execution of even dependent operations can be overlapped. It paves the way for digit-level massive parallelism [6].

This capability is due to redundant number representation, allowing several representations of a given value. Due to the serial manner of Online arithmetic, it minimizes the area of the arithmetic unit along with a low memory footprint and negligible interconnects. Additionally, Online arithmetic is well-suited for variable precision computations; once the desired precision is attained or a specific condition is satisfied, the current operation can be terminated, and the next operation starts [7]. It has a significant impact on power consumption and performance. This computational model has successfully been applied and tested to design and implement hardware accelerators for k-NN and K-means machine learning algorithms to solve classification and clustering problems in various domains and applications [8].

2.3 Road Segmentation with Fully Convolutional Networks

Advanced driver assistance systems (ADAS) and automated driving systems (ADS) have attracted considerable scientific attention in recent years (ADAS). Road segmentation, one of the crucial modules, assesses the environment, finds the drivable area, and creates an occupancy map. A linked section of road surface that is free of automobiles, people, or other obstructions is known as a drivable region. Road segmentation supports other perception modules in the ADS workflow and produces an occupancy map for planning modules. As a result, precise and effective road segmentation is required. Since cameras commonly provide high-resolution frames and are affordable, camera-based road segmentation has been studied for decades. For road segmentation, conventional computer vision algorithms used manually set characteristics like edges and histograms. However, those capabilities were difficult to adapt to new contexts and only operated in a few specific circumstances. In recent years, research has been more interested in algorithms based on convolutional neural networks (CNNs). CNNs may handle a variety of driving circumstances by incorporating large convolutional kernels into a deep neural network. The precise drivable zone was produced using existing CNN-based road segmentation algorithms, including FCN, SegNet, StixelNet, Up-Conv-Poly, and MAP, but they were computationally expensive.

3 PROPOSED DESIGN

3.1 The First Phase

FCN is a modified CNN-based model that has demonstrated great performance in image classification, which consists of three sections: a pre-trained model, a 1-by-1 convolution layer, and transposed convolutions. Also, it can describe as an encoder (a pre-trained model +1-by-1 convolutions) and decoder (transposed convolutions). For the encoder, we used VGG16 pertained model, which is trained to model on ImageNet dataset for classification, and also we replaced the 1-by-1 convolution layer with a fully-connected layer. For the decoder, we used transposed convolution layer to upsample the input to the original image size. We train FCN on

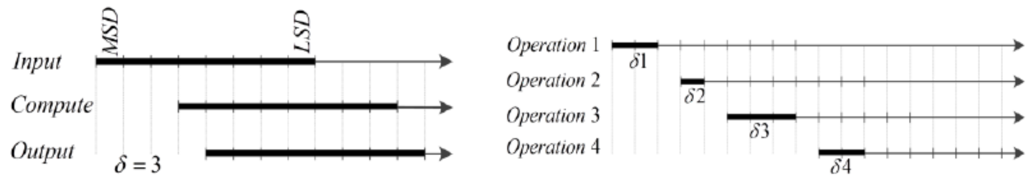

Figure 1: The timing diagram (Left chart) Serial timing with online delay δ (Right chart) Overlapping the execution of the dependent operations.

Figure 2: (Left picture) Input image (Middle picture) Mask of image segmentation (Right picture) Output image

Table 1: The dataset size for training FCN

Dataset	Features	Test Samples	Training
KITTI Road	784	124	384

Table 2: The hyper parameters of FCN training phase

#epochs	batch size	learning rate
100	8	0.0001

KITTI Road dataset [9], which consists of 508 road and lane estimation images (128x128 size) in the bird's-eye-view as shown in TABLE 1, and the hyperparameters are shown in TABLE 2. During the training phase, FCN learned to segment the road surface area in the image. For example, when our input is Figure 2 (Left picture) and the out of FCN is road surface masked area which is shown in Figure 2 (Middle picture), so we can crop the input image with the out mask and produce our Region Of Interest (ROI) which it is a region we want to process on just this region. ROI not only decreases the search area for solving our problem but also helps us to decrease power usage totally. By decreasing the search area, our latency decrease, and it brings us closer to solving the problem in real-time. This reduction in the search area makes us not do processing in irrelevant points, and the final power usage of solving the problem decreases. At the end of the first phase, we produce the ROI whit FCN which has 87% accuracy on test samples image.

3.2 The Second Phase

This section describes our design and implementation of the new SVM structure in detail. The main goal pursued in our design is to improve performance, as well as reduce energy consumption and resource utilization compared to state-of-the-art methods. In Equation 2, y_i and x_i are training samples, and the value of α_i and b are determined in the training phase which will describe in section 4. Considering Equation 3; it can be rewritten in the form of Equation 4.

$$Y_{X_t} = Sign\ (wX_t + b) \tag{4}$$

where w indicates the weight and has a fixed value during the test phase for all data points so that it can be computed during the preprocessing phase. For hardware acceleration of the inference phase, we have used a hardware-software co-design approach that takes advantage of specific custom hardware based on MSDF arithmetic beside the embedded processor of the FPGA fabric, as shown in Figure 3.

In the Selection Function, we take advantage of the Binary Signed-Digit (BSD) redundant number representation to present the output of dot-product. In this representation, the value of each position is defined by two same-weight bits, a Posibit and a Negabit [10]. The digit set of BSD is $\{-1, 0, 1\}$ where -1, 0, and 1 are represented by 01^-, 00^- (or 11^-), and 10^-. It is worth mentioning that we use Inverted Encoding for Negabit (IEN), in which-1, 0, and 1 are represented by 00^-, 01^- (or 10^-), and 11^-, respectively. In each cycle, the Selection Function's input is accumulated via a carry propagation adder, and the two most significant bits of the result (i.e., a BSD) are considered as the dot-product output. The other bits are formed as a partial remainder for the next cycle's computation. The partial remainder is initialized with zero for the first cycle's computation. The dot-product output is connected to the Sign Detection unit. In this unit, the leading zero values are ignored, and the classification result is determined by receiving the first non-zero value. Upon receiving the first non-zero value, the label of the classification result (i.e., + or $-$) is sent as the output of the classifier instance since just the sign of dot-product output is enough for classification based on the SVM algorithm. Besides

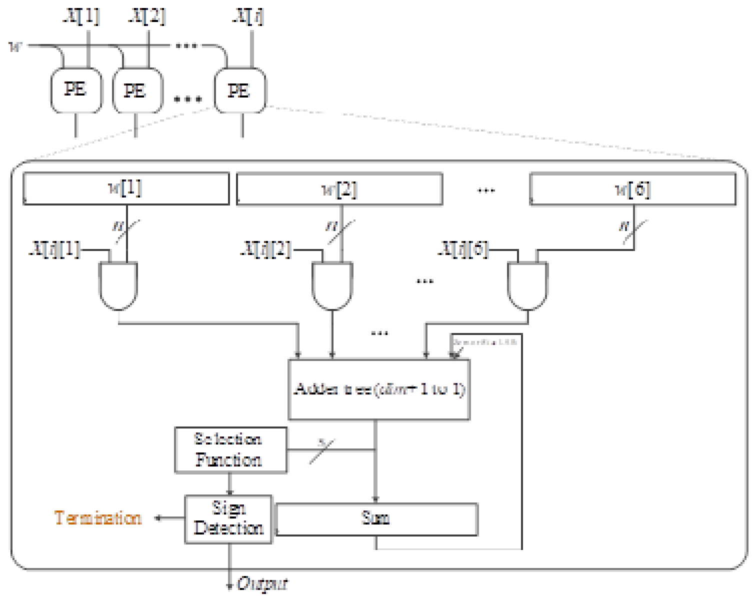

Figure 3: The abstract view of the MSDF-SVM architecture beside the MSDF dot-product unit architecture with dot-notation clarification (*dim* = 6, *n* = Precision, s = 4)

Table 3: The properties of datasets used for the evaluation

Dataset	Features	Test Samples	Training
ROAD-SURFACE	784	100	900

the output, a termination signal is also generated. After raising the termination signal, the classifier instance stops the current computation and starts processing a new data sample. This early termination significantly improves performance, and meaningfully reduces power and energy consumption. These operations are applied in a pipelined digit serial manner and sweep all the test data points.

3.3 Implementation

In this section, we have employed our proposed SVM design accelerator on ICED or NOT-ICED surface classification. First, apply FCN just once and generate our road mask, which was proposed in the previous section. This mask has been saved on each CCTV embed memory board for generating ROI in each 30th frame. In the next step, we train our SVM model on the dataset, which consists of 1000 road surface images (with 128x128 size) in two classes: iced road surface (iced-label) and normal (noticed-label) road surface, shown in TABLE 3. In this dataset, its iced-label images consist of five types of usual ice on highways: fresh iced, melt ice, black ice, ice with the cover of snow, and rare ice. Grid Search in the SVM training phase finds the optimal value of α, w , b, with the CV algorithm and saves these parameters on CCTV embed memory board. In the end, we apply FCN-MASK on each 30th frame and feed the masked image, Figure 2 (Right picture), to our SVM structure which proposes in the previous section.

4 EVALUATION AND COMPARISON

This section assesses our suggested design and contrasts it with the most effective state-of-the-art approaches. We have conducted various experiments and evaluations to demonstrate the efficacy of our suggested design. On a Zynq UltraScale+ XCZU7EV MPSoC from Xilinx, our design was implemented. To ensure a fair and accurate comparison under the same circumstances and with the same resources, we also included the best past designs in our studies.

4.1 Experimental Setup

We select the best state-of-the-art implementation of linear SVM on FPGA [11] (i.e., SVM-Parallel). This implementation encounters a challenge in processing high-dimensional datasets where the required hardware exceeds available FPGA resources. We implement our proposed method with VHDL. We compare the *latency*, *Resource Utilization,* and *Accuracy* of our method with KNN, Alex Net, ResNet18, and google net, which are trained on the same dataset and used for ice recognition in previous work [12].

Table 4: Comparison of power usage and memory usage between our proposed methods and previous related models.

Models	Static power consumption (W)	Memory usage (MB)
k-NN	0.51	4.8
Alex Net	1.71	159
Google Net	2.54	254
ResNet18	5.71	298
Our SVM	0.59	5.1

Table 5: Comparison of latency between our proposed methods and previous related models

Models	Latency(ms)
k-NN	50
AlexNet	200
GoogleNet	241
ResNet18	256
Our SVM	78

4.2 Comparison of Resource Utilization

Resource efficiency is an essential aspect of our design. A comparison between our proposed design and other related models regarding resource usage has been reported in Table 4 to indicate the improvement in our design. Other models have been deployed on a processor which has 1.2GHz Quad Core ARM Cortex-A53 CPU and 900MHz 1GB RAM LPDDR2 Memory.

As indicated in Table 4, our proposed design consumed the lowest resources in terms of the number of LUT and the number of Flip-Flops (FF) compared to the other designs. This improvement would be more critical when the SVM classifier needs to be run on edge devices and portable devices with limited resources and areas.

4.3 Comparison of Latency

Latency is the most important factor in our evaluation since the main aim of our proposed design was to improve the speed of SVM classification, especially when it should run in real-time. We have measured the amount of *latency* for our design by running these models on a test sample video. The results of this comparison are reported in Table 5.

As illustrated by Table 5, our proposed design was placed at the first rank compared to the related model. Our design indicated 5X improvements in *latency* against the Alex Net and Google Net. Also, this improvement was 2X against K-NN classifier.

4.4 Comparison of Accuracy

Another aspect of comparison in our paper is accuracy, which is highly important for modern smart and portable devices. We have measured the accuracy for each model on the same test images. Table 6 illustrates the results of this evaluation. As indicated in Table 6, our design has more accurate compared to the KNN model. It can be inferred that our design had better performance in terms of *latency* and *Resource Utilization* compared to other related models and also had a small reduction in accuracy.

Table 6: The comparison of accuracy between our proposed methods and related works

Models	Precision (%)	Recall (%)	Specificity (%)
k-NN	89.6	89.7	89.6
AlexNet	94.7	94.7	97.3
GoogLeNet	96.9	97.0	98.4
ResNet18	99.2	99.6	99.2
Our SVM	93.4	93.3	95.2

5 RELATED WORKS

Numerous researchers have proposed a number of laser radar and infrared detector-based systems for detecting road conditions. The laser radar approach evaluates road conditions based on backscattering ratios for various polarizations or the requisite time difference for laser reflection. In the infrared detector method, multiple wavelengths are irradiated onto the road surface, and road surface conditions are estimated based on the difference in the intensity of the reflected light [13], [14].

Due to the high deployment costs, these approaches have not been widely used to detect the state of public highways. On the other hand, the method of using visible cameras does not require expensive equipment and is, therefore, a realistic way of detecting road surface conditions. This approach detects road surface conditions using the particle and direction of texture or polarization properties of the image; these technologies employing visible light cameras were only utilized during the day. The following evening, a visible camera was used to suggest a road surface condition-detecting approach [15]. This research presents a system for continuously sensing road surface conditions. It was possible by combining daylight and twilight data.

Lai and Yung [16] suggested a standard lane recognition approach based on road markings. Based on the assumption that a lane is always the center line of two parallel edge lines, the orientation discrimination and length of traffic lane markings and curb structures are computed and utilized to determine the lane position.Kim [17] utilized the random sample consensus with a particle filter to remove lane markings and a probabilistic-grouping technique to recognize the lanes.

The hybrid method described by Daigavane and Bajaj [18] is based on edge detection. In engineering extraction lanes, they use ant colony optimization to connect edges that need to be linked, and the Hough transform to generate two mathematical problems. Although these approaches are more flexible than manual methods, they may result in a rapid decline in accuracy or even failure if the markings are obscured by contaminations or cannot be seen in the dark. Therefore, they cannot be used with cameras, whose settings may alter in these circumstances. The activity-map-based approaches [19], [20] can avoid being affected by the legibility of road markings, but the results for lane detection are imprecise. Consequently, they are not well suited for the stringent precision requirements of lane-changing rate detection.

Kim [21] suggested an approach for object detection and tracking that includes feature grouping and background subtraction. The precision of feature grouping is enhanced, but the computation required increases substantially.

6 CONCLUSION

The main goal of this work was to create ice detection models that could recognize road conditions in real-time, which was both dependable and inexpensive. Using the FCN model, which trains on the KITTI road dataset, and using the VGG16 pertained model for providing ROI in each CCTV and it runs just for one time, and the mask output of FCN saved in embed memory using the SVM algorithm for ice recognition. This SVM algorithm trained on a dataset that was created from the webcam images with two categories, iced and non-iced. 80% of the data were used during training and validation, and the rest were used to test the detection quality of the trained models based on several performance indices, such as precision, recall, and specificity rate. Our proposed method has 93% accuracy, but its power consumption and memory usage are very low, and it is very suitable for running on edge devices. We also used the MSDF computation method in SVM computation. This technique gave us the chance of early termination, which means that we can find the result before all of the computations are done and terminate them. So our proposed design significantly improved the latency. The comparison results also proved that our proposed approach could surpass the related works and state-of-the-art designs on real-time ice detection.

ACKNOWLEDGMENTS

This research was supported by Brain Pool program funded by the Ministry of Science and ICT through the National Research Foundation of Korea (NRF-2021H1D3A2A02040040) and in part supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2020R1I1A3063857).

REFERENCES

[1] S. Gorgin, M. H. Gholamrezaei, D. Javaheri, and J.-A. Lee, "An Efficient FPGA Implementation of k-Nearest Neighbors via Online Arithmetic," in *2022 IEEE 30th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*, 2022, pp. 1–2, doi: 10.1109/FCCM53951.2022.9786165.

[2] V. V Viikari, T. Varpula, and M. Kantanen, "Road-Condition Recognition Using 24-GHz Automotive Radar," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 639–648, 2009, doi: 10.1109/TITS.2009.2026307.

[3] S. Kumar, S. Singh, and J. Kumar, "Multiple Face Detection Using Hybrid Features with SVM Classifier BT - Data and Communication Networks," 2019, pp. 253–265.

[4] J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez, "A comprehensive survey on support vector machine classification: Applications, challenges and trends," *Neurocomputing*, vol. 408, pp. 189–215, 2020, doi: https://doi.org/10.1016/j.neucom.2019.10.118.

[5] A. A. A. Ali and S. Mallaiah, "Intelligent handwritten recognition using hybrid CNN architectures based-SVM classifier with dropout," *J. King Saud Univ. - Comput. Inf. Sci.*, 2021, doi: https://doi.org/10.1016/j.jksuci.2021.01.012.

[6] M. D. Ercegovac, "On left-to-right arithmetic," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*, 2017, pp. 750–754, doi: 10.1109/ACSSC.2017.8335445.

[7] M. D. Ercegovac, "On Reducing Module Activities in Online Arithmetic Operations," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 524–528, doi: 10.1109/IEEECONF51394.2020.9443576.

[8] S. Gorgin, M. Gholamrezaei, D. Javaheri, and J.-A. Lee, "kNN-MSDF: A Hardware Accelerator for k-Nearest Neighbors Using Most Significant Digit First Computation," in *2022 IEEE 35th International System-on-Chip Conference (SOCC)*, 2022, pp. 1–6, doi: 10.1109/SOCC56010.2022.9908102.

[9] J. Fritsch, T. Kühnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013, pp. 1693–1700, doi: 10.1109/ITSC.2013.6728473.

[10] G. Jaberipur, "Redundant number system-based arithmetic circuits," in *Arithmetic Circuits for DSP Applications*, John Wiley & Sons, Ltd, 2017, pp. 273–312.

[11] X. Song, H. Wang, and L. Wang, "FPGA Implementation of a Support Vector Machine Based Classification System and Its Potential Application in Smart Grid," in *2014 11th International Conference on Information Technology: New Generations*, 2014, pp. 397–402, doi: 10.1109/ITNG.2014.45.

[12] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi, "Traffic monitoring and accident detection at intersections," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 2, pp. 108–118, 2000, doi: 10.1109/6979.880968.

[13] P. Jonsson, "Remote sensor for winter road surface status detection," in *SENSORS, 2011 IEEE*, 2011, pp. 1285–1288, doi: 10.1109/ICSENS.2011.6127089.

[14] T. Nakashima, H. Nakajima, K. Sumi, and O. Matoba, "Development of Road Surface Condition Detection System using Near-infrared Light- Absorption and Polarization Characteristics of Water," *Trans. Soc. Instrum. Control Eng.*, vol. 46, no. 12, pp. 746–753, Jan. 2011.

[15] K. Shibata, T. Furukane, S. Kawai, and Y. Horita, "Distinction of Wet Road Surface Condition at Night Using Texture Features," *Electron. Commun. Japan*, vol. 97, no. 6, pp. 51–57, Jun. 2014, doi: https://doi.org/10.1002/ecj.11562.

[16] A. H. S. Lai and N. H. C. Yung, "Lane detection by orientation and length discrimination," *IEEE Trans. Syst. Man, Cybern. Part B*, vol. 30, no. 4, pp. 539–548, 2000, doi: 10.1109/3477.865171.

[17] Z. Kim, "Robust Lane Detection and Tracking in Challenging Scenarios," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 1, pp. 16–26, 2008, doi: 10.1109/TITS.2007.908582.

[18] P. M. Daigavane and P. R. Bajaj, "Road Lane Detection with Improved Canny Edges Using Ant Colony Optimization," in *2010 3rd International Conference on Emerging Trends in Engineering and Technology*, 2010, pp. 76–80, doi: 10.1109/ICETET.2010.128.

[19] B. D. Stewart, I. Reading, M. S. Thomson, T. D. Binnie, K. W. Dickinson, and C. L. Wan, "Adaptive lane finding in road traffic image analysis," in *Seventh International Conference on Road Traffic Monitoring and Control, 1994.*, 1994, pp. 133–136, doi: 10.1049/cp:19940441.

[20] T. N. Schoepflin and D. J. Dailey, "Dynamic camera calibration of roadside traffic management cameras for vehicle speed estimation," *IEEE Trans. Intell. Transp. Syst.*, vol. 4, no. 2, pp. 90–98, 2003, doi: 10.1109/TITS.2003.821213.

[21] Z. Kim, "Real time object tracking based on dynamic feature grouping with background subtraction," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–8, doi: 10.1109/CVPR.2008.4587551