Accredited Ranking SINTA 2
Decree of the Director General of Higher Education, Research and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online on: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 7 No. 5 (2023) 1239 - 1245 ISSN Media Electronic: 2580-0760

MSER-Vertical Sobel for Vehicle Logo Detection

Gamma Kosala[1], Agus Harjoko[2], Sri Hartati[3]
[1]School of Computing, Telkom University, Bandung, Indonesia
[2,3]Computer Science and Electronics, Faculty of Mathematics and Natural Science, Universitas Gadjah Mada, Yogyakarta, Indonesia
[1]gammakosala@telkomuniversity.ac.id, [2]aharjoko@ugm.ac.id, [3]shartati@ugm.ac.id

Abstract

Detecting a vehicle logo is the first step before the actual recognition of the logo. However, detecting logos can pose difficulties due to various factors, including logo variations, differing scales and orientations, background interference, varying lighting conditions, and partial obstruction. This paper presents a vehicle logo detection method using handcrafted features. We used a combination of Maximally Stable Extremal Region (MSER) and Vertical Sobel. We combine vertical Sobel with MSER to overcome MSER's limitation in recognizing objects of different sizes. These two features are merged using closing morphology operation to form blobs selected as logo candidate areas. Moreover, a Support Vector Machine (SVM) is implemented to choose a logo area by analyzing each candidate's Histogram of Oriented Gradient (HOG). The proposed method was compared to other methods by implementing them on the same dataset. The significant advantage of using MSER-Vertical Sobel is its fast computation time. It is faster than other approaches that use non-handcrafted features. The test results show that the MSER-Vertical Sobel can achieve high accuracy and the fastest computation time.

Keywords: *car logo detection; maximally stable extremal region; vertical sobel*

1. Introduction

In addition to a license plate, manufacturer information is often used to identify a car. If the vehicle license plate is illegible, information such as the vehicle manufacturer becomes necessary for vehicle identification. Its logo can determine manufacturer information from a car. Meanwhile, logo detection is the initial stage before the logo recognition process. Logo detection can be challenging due to various factors, such as variability in logo, scale and orientation, background clutter, lighting conditions, partial occlusion, etc.

As with other object detection, logo detection is strongly influenced by the features. Both handcrafted and non-handcrafted features were used in several earlier investigations. Deep learning network architecture is used to directly extract features from the original image to produce non-handcrafted features. This method's disadvantage is that it takes a lot of time to compute. in contrast to the extraction of handcrafted features using predetermined rules. The shape, texture, and color of the object are typically used to extract handcrafted features. This feature extraction is relatively fast in computation time compared to the non-handcrafted feature extraction.

The logo detection is affected by the selection of features and how to extract them. Logo detection is often preceded by selecting a Region of Interest (ROI) to limit the logo search area. ROI commonly is determined based on the position of the grill, rear lights, or license plate. In [1], they detect the logo regarding the license plate. After the plate area is found, the logo area is determined by dividing the area above the plate into five parts. The five parts consist of two parts for the lights, two parts for the grille, and one for the logo. The parts are determined based on the distribution of pixels on the horizontal axis. The logo area is set at the center of the five sections. In [2], they started searching for the logo by finding the car's position using the Viola-Jones method. Furthermore, the car number plate is searched by performing edge extraction and histogram analysis. The search ROI logo is then assigned to the top of the license plate. Histogram of Oriented Gradient (HOG) feature and Adaboost are used to detect logos on ROI. Both [1] and [2] can only detect logos located exactly above a license plate.

Some researchers try to combine the use of local features and global representation to find the logo area. [3] combines Speeded Robust Features (SURF) with grid-based representation, [4] combines Maximally Stable Extremal Region (MSER) with Scale Invariant Feature Transform (SIFT), and [5] combines HOG with Grid-based concatenation of features. The combination of local features and global representation can improve detection accuracy compared to using only local

Accepted: 08-04-2023 | Received in revised: 22-10-2023 | Published: 28-10-2023

features. However, this combination causes the computation time to be longer.

Nowadays, non-handcrafted features are often implemented in image processing applications. The use of non-handcrafted features has been widely applied in various fields, such as health [6]–[9], plantations [10]–[12], surveillance [13]–[16], psychology [17]–[19], etc. Non-handcrafted features can also be used to detect logo areas. One of the studies that utilize deep learning to detect logos has been carried out by Rong et al. [20]. In that research, the Single Shot Detector (SSD) architecture becomes a raw image as input without having to go through the manual feature extraction stage. SSD consists of several layers of neural networks. Some initial neural network layers act as feature extractors while the later layers act as classifiers. SSD utilizes several boxes to search for objects in the entire image area. The detection rate in this study reached 98.33%.

Deep learning provides high detection accuracy but requires a longer computation time. Non-handcrafted features frequently locate logos with great accuracy, but they need expensive computing effort. We created a new handcrafted feature to achieve more effective computational time. On MSER, this new function is based. To address the MSER issue in identifying an item with different sizes, a vertical edge has been added.

In this study, a method for identifying potential logos using hand-crafted feature extraction is presented. The manually created features were built using MSER [21]. We integrate MSER with vertical Sobel to address MSER's shortcomings in identifying objects of variable sizes [22]. Using the closure operation, blobs produced from MSER and vertical Sobel are combined. To choose the logo candidate area, each blob is examined based on area and aspect ratio. The idea to combine MSER and vertical Sobel has been tested for license plate detection [23]. Only the dominant vertical Sobel is used in [23]. In plate detection, the character with the plate background has a contrasting color. While in logo detection, some cases reveal that the color of the logo with the base color of the car does not contrast too much. To cope with that problem, both strong and weak edges from vertical Sobel extraction are used. After the candidate extraction stage, each candidate's HOG feature is retrieved during the candidate selection stage. The SVM architecture is then utilized to classify candidates as logos or non-logo.

This research aims to modify the merging of two non-handcrafted features, MSER and Vertical Sobel, which have their respective advantages. The modified proposed method will then be compared with other methods regarding f-measure values and computation time.

2. Research Methods

Our suggested solution divides the logo-detecting process into four basic parts. Preprocessing, candidate logo extraction, and candidate selection make up the first three steps of the procedure. Overall, Figure 1 depicts the steps of suggested logo detection.

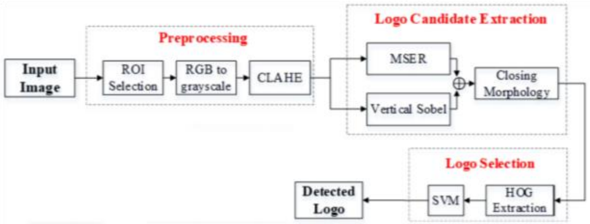


Figure 1. The overall stages of our proposed method

2.1 Preprocessing

This stage begins with determining the ROI. The ROI is selected by referring to the location of the license plate. The assumption used is that the logo area is above the license plate. If the length of the plate is p pixels, the ROI length includes p to the right of the plate and p to the left of the plate. The ROI width is determined to be 5 times the detected plate height (t). Based on this assumption, the ROI is rectangular with 3p×5t. Figure 2 shows an illustration of ROI selection.

Figure 2. ROI selection

The image covered in the ROI is then converted into a greyscale image. Furthermore, the Contrast-Limited Adaptive Histogram Equalization (CLAHE) algorithm [24] is implemented on grayscale images. Figure 3 shows that the results from CLAHE have better contrast than the grayscale image.

(a) input image

(b) grayscale image

(c) after CLAHE is put into effect

Figure 3 The output difference between grayscale and CLAHE

DOI: https://doi.org/10.29207/resti.v7i5.5034

2.2 Candidate Extraction

At the candidate extraction stage, the feature used to search for logo candidates is the MSER-Vertical Sobel (MSER-VS) which is our proposed method. After MSER-VS is extracted, a closing morphology operation is employed to merge the blobs.

The MSER and the Vertical Sobel are two features that make up the MSER-VS. After the thresholding technique, the MSER extraction is applied by examining each section of the image. It applied each threshold value between 0 and 255. The region that fluctuates a little, or even remains constant, is chosen as MSER. The mathematical definition of linear MSER [21] is in Formula 1.

$$q(i) = \frac{|Q_i - Q_{i-\Delta}|}{|Q_{i-\Delta}|} \quad (1)$$

Q_i is the linked area of threshold i, Δ represents the slight variation in the gray value, and $q(i)$ represents the rate at which region Q_i changes about threshold i. Where $q(i)$ is the local minimum, the MSER is represented by Q_i.

The logo area tends to have a specific characteristic range of aspect ratio values. We modified the MSER by restricting the aspect ratio of the extracted blobs. The limitation on the aspect ratio of the MSER results aims to reduce the blob that is not the logo area. Modified MSER is formulated in Formula 2.

$$Q_{i\_modified} = \begin{cases} 255, & if \quad \frac{Q_{iw}}{Q_{ih}} < \tau \\ 0, & otherwise \end{cases} \quad (2)$$

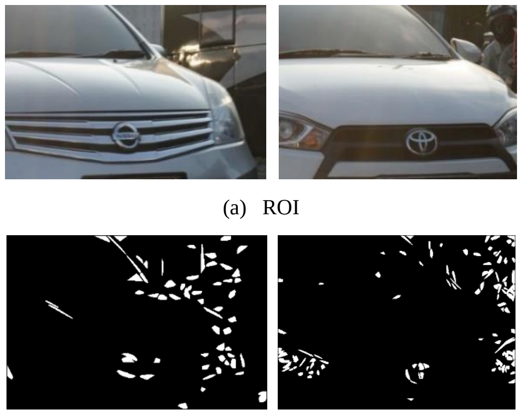
The aspect ratio in each blob is obtained by dividing the blob width Q_{iw} by the blob height Q_{ih}. As the majority of logos have a consistent length and height, we set the aspect ratio requirement at $\tau = 3$. Only blobs with an aspect ratio under are employed.

MSER is capable of detecting things from a variety of angles. When the camera is not always in the same position as the car, it is highly useful for spotting logos. However, in some cases, the extracted blobs from MSER do not cover the entire logo area, as shown in Figure 4.

We attempted to solve the issue by combining MSER and Vertical Sobel. Convolution ($\otimes$) between CLAHE image I and Sobel Vertical operator G_y as in Formula 3 is used to extract the Vertical Sobel.

$$S = I \otimes G_y = I \otimes \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (3)$$

The extraction of Vertical Sobel can produce a blob that can support the extracted blob from MSER so that the blob produced by both can cover the entire logo area. After extracting each blob, merging MSER and dominant vertical Sobel is done using OR logic operations in Formula 4.



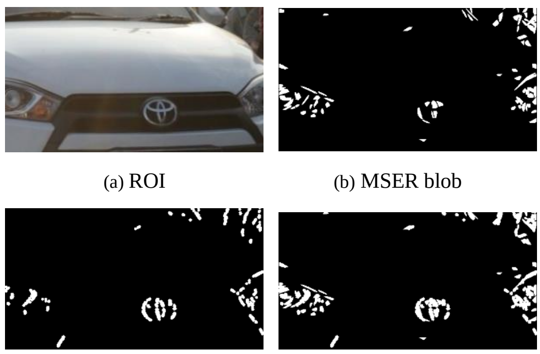
(a) ROI

(b) extracted blobs from MSER

Figure 4. MSER extraction: The blobs cannot cover the entire logo area.

$$MSER\_VS = Q_{i_{modified}} \vee S(x) \quad (4)$$

Figure 5 shows an example of the MSER-VS merging process. The extracted blob from Vertical Sobel can fill the space between the extracted blob from MSER so that the whole blob covers the entire logo area.



(a) ROI (b) MSER blob

(c) Vertical Sobel blob (d) MSER- Vertical Sobel blob

Figure 5. MSER-VS extraction

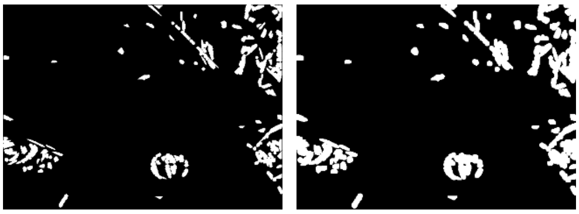
The closing morphology operation is employed to merge the extracted blob from MSER and the extracted blob from Vertical Sobel. Both blobs are merged to build the bigger blob that covers the entire logo area. Accordingly, the closing operation of the image I by structuring element E, denoted $I \bullet E$, is defined as Formula 5.

$$I \bullet E = (I \oplus E) \ominus E \quad (5)$$

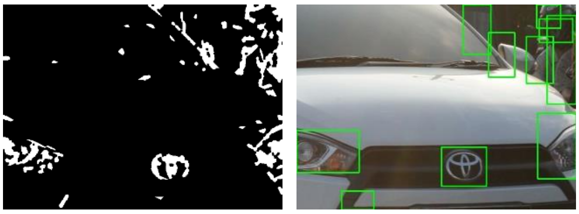
$\oplus$ and $\ominus$ are dilation and erosion operations in (5), respectively. Dilation is used to thicken pixels as part of the closing operation, and erosion is used to remove separated, little pixels. By using the related component extraction approach, the blob from the closing operation is found. Based on the area and aspect ratio, the blob that is picked as a logo contender is selected. After that, a bounding box was used to isolate the chosen

DOI: https://doi.org/10.29207/resti.v7i5.5034

candidate. The procedure for finding the candidate among the retrieved blobs is shown in Figure 6.



(a) ROI (b) MSER blob



(c) Vertical Sobel blob (d) MSER- Vertical Sobel blob

Figure 6. MSER-VS extraction

2.3 Logo Selection

After passing the candidate extraction stage, each candidate will be analyzed whether the logo area or not. The candidate's selection is implemented by extracting HOG in each candidate and then classifying it using SVM architecture. As a rule, HOG extraction uses a fixed cell size in the fixed size of the input image. As the candidates have various sizes, we used a changeable size cell that the size of the cell depends on the candidate's size. To equate the number of HOGs in each candidate, the number of cells in each candidate will be equalized. In each cell, the total gradient is computed as in Formula 6, and the gradient direction is determined as in Formula 7

$$R = \sqrt{{G_x}^2 + {G_y}^2} \tag{6}$$

$$\theta = \tan^{-1}\left(\frac{G_y}{G_x}\right) \tag{7}$$

G_x is the gradient in the vertical axis and G_y is the gradient in the horizontal axis. Next, a histogram of the gradient is produced by dividing it into 9 bins along the gradient's direction. Depending on the gradient direction, each pixel's entire gradient is then collected into one of nine bins. The candidates are divided into logo area and non-logo area using SVM, which is applied based on the HOG in each candidate.

3. Results and Discussions

In this section, we experimentally evaluate the proposed method. The whole experiment was conducted on an i5 CPU and NVIDIA GTX 1070 GPU with 8GB memory. The proposed method is also compared with several other methods using the same dataset on the same machine. Performance evaluation of the compared methods includes 2 aspects, the F-measure and the computation time.

Logo detection is claimed to be successful if the bounding box between the detection results and the ground truth has an Intersection over Union (IoU) value above 0.5. Meanwhile, computation time calculation is calculated from reading the input image until the bounding box area of the logo is determined.

3.1 Dataset

We designed our dataset that will be used to test the performance of the proposed method. The dataset used in this study is a collection of vehicle images on the arterial road in Yogyakarta, Indonesia. The camera used to obtain the image is placed on the roadside. The data collection time varies from morning to evening. The camera captures the front or rear view of a four-wheeled vehicle. When the image is captured, the camera height is fixed in the range of 1m - 1.5m. We call this dataset as UGM-LP. Detailed information about UGM-LP can be seen in Table 1.

Table 1. Detailed Information About UGM-LP Dataset

Subset	Image quantity	Total Plates	Total Logos	Logo Size (height × width) px
1	511	596	582	(10 ~ 76) × (12 ~ 87)
2	510	597	561	(6 ~ 81) × (11 ~ 188)
3	491	555	544	(10 ~ 79) × (9 ~ 138)
4	490	557	547	(9 ~ 67) × (9 ~ 105)

3.2 MSER-VS Extraction

Three parameters affect the MSER-VS blob extraction result as a logo candidate, namely the value of the change in MSER intensity (Δ) (Formula 1), the ratio threshold value (τ) (Formula 2), and the closing kernel value (E) (Equation 5). Those three parameter values were varied to get the best results. At this stage, the best parameter value is determined from the percentage of logos that fail to be extracted as candidates. The fewer logos that fail to extract, the better the candidate extraction process.

This parameter Δ affects the results of the MSER blobs produced. This Δ is the value of increasing intensity for the search for a stable extremal region. The greater the value Δ, the fewer blobs are produced. On the contrary, the smaller the value Δ, the more blobs are produced. An illustration of the influence Δ on the formation of candidate logos can be seen in Figure 9. In Figure 9, there is a silver logo on the white base paint of the car. When the value of Δ=1, MSER blobs were successfully extracted in almost all parts of the logo. The larger the value Δ, the fewer MSER blobs are extracted. This causes the logo candidate extraction failure when the value of Δ is too large.

DOI: https://doi.org/10.29207/resti.v7i5.5034

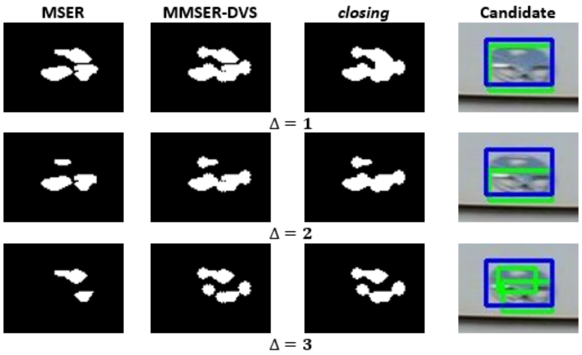


Figure 9. Illustration of the influence of the Δ

The parameter τ also affects the MSER blob results. The τ is the aspect ratio limit value on the selected MSER blob. If the MSER blob has an aspect ratio above τ, the blob will be omitted. The smaller the τ, the fewer MSER blobs are formed. An illustration of the effect of the value of τ on the formation of candidates can be seen in Figure 10. In Figure 10 when the value is $\tau \leq 2{,}5$, the combined blob formed after the closing process does not cover the entire logo area so that the candidate formed is not perfect. When the value is $\tau \leq 2{,}5$, MSER blobs with aspect ratio values above 2.5 are omitted. This causes some MSER blobs in the logo area to disappear. Sobel's vertical blobs were also unable to support the missing logo area, so after the closing process, the combined blobs that were formed did not cover the entire logo area. Meanwhile, when the value is $\tau \geq 4$, additional blobs around the logo cause the extraction results to widen.

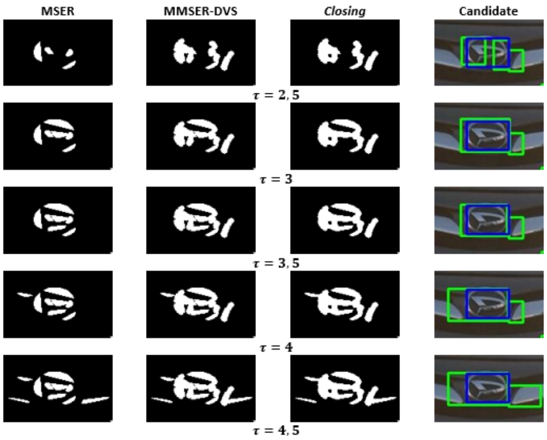


Figure 10. Illustration of the influence of the τ

Parameter E affects the size of the kernel during the closing process of the results of the MSER-VS blob. The reference to getting the parameter E is obtained from the percentage of plate height. An E value that is too small causes the width of each blob not to be long enough to unite the small blobs in the logo area, while an E value that is too large causes the blob in the logo area to expand and merge with other blobs around the logo area.

Figure 11 shows the failure of logo extraction when the value of E is too small or when the value of E is too large. When $E \leq 5\%$, the closing kernel has not been able to unite the logo's upper and lower blobs, which causes the logo to split into two parts. Meanwhile, when $E \geq 20\%$, the logo blob is combined with the sticker blob at the top of the logo so that the combined blob that is formed expands and fails to be extracted to become a candidate.

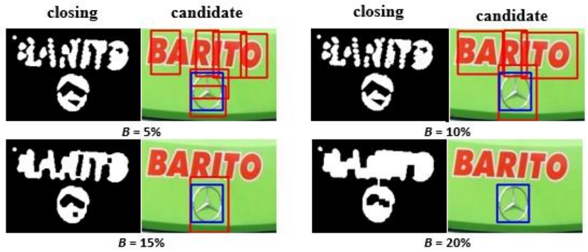


Figure 11. Illustration of the influence of the E

3.3 Logo Detection Performance Evaluation

The result of logo detection using MSER-VS shows that this method can detect logos with various colors. Because MSER-VS does not use color features, the logo's color difference does not affect the detection results. The proposed method is also capable of detecting logos of various sizes. Logo detection is successfully carried out when the vehicle is close to the camera so that the logo size is large as well as when the vehicle is far from the camera, so the logo size is small. Some of the successful logo detection results can be seen in Figure 12.

The proposed method still has a weakness when the logo reflects the scorching sun. This condition causes some parts of the logo to have high intensity so that the blobs cannot be extracted. In addition to bright light conditions, detection also fails when other objects around the logo cause the logo to be partially blocked.

Figure 13. Successful logo detection using MSER-VS. The green box is the detection box while the red box is the ground truth.

DOI: https://doi.org/10.29207/resti.v7i5.5034

This condition causes the resulting blob to widen and become one with the other object's blob. Some of the results of failed logo detection can be seen in Figure 13.

The performance of logo detection using MSER-VS has been compared with several other methods. The comparison methods are Vertical Sobel, MSER, Modified-MSER, and Single Shot Detector (SSD).

Figure 14. Failed logo detection using MSER-VS. The green box is the detection box while the red box is the ground truth. Detection failure due to insufficient illumination and false positives.

Comparisons are made regarding the value of precision, recall, F-measure, and computational time during the testing process. The summary of the performance test comparison of the five methods can be seen in Table 2. The best precision, recall, and F-measure test values were achieved by the SSD method with a precision value of 99.31%, a recall value of 89.34%, and an F-measure value of 94.05%. While the MSER-VS method proposed in this study achieved a precision value of 93.97%, a recall value of 82.26%, and an F-measure value of 87.72%.

MSER-VS achieves a lower precision value than the precision value in the SSD method by a difference of 5.34%. In the test concerning the recall value, MSER-VS achieved a lower value than the SSD method with a difference of 7.08%. Overall, when viewed from the F-measure value, the MSER-VS method achieved a lower value with a difference of 6.33% than the SSD method. Although the MSER-VS method achieves precision, recall, and F-measure values under the SSD method, the computation time used by MSER-VS is about 13× faster than the computation time used by SSD.

Vertical Sobel, MSER, Modified-MSER, and MSER-VS use handcrafted features, while the SSD method is classified as a deep learning method that uses non-handcrafted features. As shown in Table 2, among other handcrafted features, MSER-VS achieved the best values in precision, recall, and f-measure. This indicates that MSER and Vertical Sobel combination in MSER-VS can produce better blobs as logo candidates than other handcrafted features. Judging from the computing time, MSER-VS also achieved the fastest time. This is because the merging of MSER with Vertical Sobel causes many blobs to become a single blob so the number of logo candidates decreases. The decrease in candidate logos causes the computation time used in MSER-VS to be faster.

The SSD method achieved the best scores for precision, recall, and f-measure values. This method uses a convolution layer and a max-pooling layer as the initial stage of feature extraction. Furthermore, the convolution filter is used for object detection at various scales with the help of anchor boxes. This process uses several layers of a neural network so that the SSD can detect logos better than other handcrafted methods. The many layers of the neural network also affect the computation time used. Compared to handcrafted methods, SSDs require a much longer computing time.

Table 2. Logo Detection Performance Comparison

Method	Performance Evaluation			
	Precision	Recall	F-Measure	Time (ms)
Vertical Sobel	87,60%	78,02%	82,51%	44
MSER [25]	77,83%	70,67%	74,07%	75
Modified MSER	85,96%	72,19%	78,45%	52
SSD [20]	99,31%	89,34%	94,05%	586
MSER-VS	93,97%	82,26%	87,72%	43

4. Conclusion

In this paper, we design the MSER-VS feature to detect vehicle manufacturer logos. The MSER-VS feature is formed by combining the MSER blob with the Vertical Sobel blob. The addition of Vertical Sobel to MSER blobs is proven to produce denser blobs as a candidate for the logo area. Compared with MSER, MSER-VS can increase the F-measure value at the time of logo detection by 11.54%. When compared to other handcrafted feature methods, MSER-VS achieved the best f-measure value with a value of 87.72%.

We also compared MSER-VS with the non-handcrafted feature method, namely SSD. When compared to SSD, the F-measure value of MSER-VS is 6.33% lower. However, the computation time required by MSER-VS is 13× faster than the computation time required by the SSD. Whereas MSER-VS employs modified handmade features to provide quick computational time, SSD uses deep learning architecture, which requires several layers. Our approach might be utilized for real-time activities because the computation takes just a short time.

Future Works

The proposed method must still be discovered when the logo reflects the scorching sun. Future research can do a combination of handcrafted features and deep

DOI: https://doi.org/10.29207/resti.v7i5.5034

learning. The deep learning layer is expected to be able to get information from each pixel of the logo so that if some parts of the logo are exposed to bright light, through several other parts, it can still be identified as a logo.

References

[1] A. Psyllos, C. N. Anagnostopoulos, and E. Kayafas, “Vehicle model recognition from frontal view image measurements,” *Comput Stand Interfaces*, vol. 33, no. 2, pp. 142–151, 2011, doi: 10.1016/j.csi.2010.06.005.

[2] H. Pan and B. Zhang, “An integrative approach to accurate vehicle logo detection,” *Journal of Electrical and Computer Engineering*, vol. 2013, 2013, doi: 10.1155/2013/391652.

[3] J. W. Hsieh, L. C. Chen, and D. Y. Chen, “Symmetrical SURF and Its applications to vehicle detection and vehicle make and model recognition,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 6–20, 2014, doi: 10.1109/TITS.2013.2294646.

[4] V. P. Le, “Logo Detection, Recognition and Spotting in Context by Matching Local Visual Features,” 2015. [Online]. Available: https://tel.archives-ouvertes.fr/tel-01373417

[5] L. C. Chen, J. W. Hsieh, Y. Yan, and D. Y. Chen, “Vehicle make and model recognition using sparse representation and symmetrical SURFs,” *Pattern Recognit*, vol. 48, no. 6, pp. 1979–1998, 2015, doi: 10.1016/j.patcog.2014.12.018.

[6] S. C. Huang, A. Pareek, S. Seyyedi, I. Banerjee, and M. P. Lungren, “Fusion of medical imaging and electronic health records using deep learning: a systematic review and implementation guidelines,” *NPJ Digit Med*, vol. 3, no. 1, 2020, doi: 10.1038/s41746-020-00341-z.

[7] A. Sanaat, H. Arabi, I. Mainta, V. Garibotto, and H. Zaidi, “Projection Space Implementation of Deep Learning-Guided Low-Dose Brain PET Imaging Improves Performance over Implementation in Image Space,” *J Nucl Med*, vol. 61, no. 9, pp. 1388–1396, 2020, doi: 10.2967/jnumed.119.239327.

[8] F. Zhang *et al.*, “A Single Model Deep Learning Approach for Alzheimer’s Disease Diagnosis,” *Neuroscience*, vol. 491, pp. 200–214, May 2022, doi: 10.1016/j.neuroscience.2022.03.026.

[9] A. Priyatama, Z. Sari, and Y. Azhar, “Deep Learning Implementation using Convolutional Neural Network for Alzheimer’s Classification,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 2, pp. 310–217, Mar. 2023, doi: 10.29207/resti.v7i2.4707.

[10] R. I. Hasan, S. M. Yusuf, and L. Alzubaidi, “Review of the state of the art of deep learning for plant diseases: A broad analysis and discussion,” *Plants*, vol. 9, no. 10, pp. 1–25, 2020, doi: 10.3390/plants9101302.

[11] A. Ahmad, D. Saraswat, and A. El Gamal, “A survey on using deep learning techniques for plant disease diagnosis and recommendations for the development of appropriate tools,” *Smart Agricultural Technology*, vol. 3, p. 100083, Feb. 2023, doi: 10.1016/j.atech.2022.100083.

[12] D. O. Pribadi *et al.*, “Mapping smallholder plantation as a key to sustainable oil palm: A deep learning approach to high-resolution satellite imagery,” *Applied Geography*, vol. 153, p. 102921, Apr. 2023, doi: 10.1016/j.apgeog.2023.102921.

[13] F. Pérez-Hernández, S. Tabik, A. Lamas, R. Olmos, H. Fujita, and F. Herrera, “Object Detection Binary Classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance,” *Knowl Based Syst*, vol. 194, p. 105590, 2020, doi: 10.1016/j.knosys.2020.105590.

[14] C. Jyotsna and J. Amudha, “Detection from Surveillance Video,” *ICIMIA 2020 IEEE International Conference on Innovative Mechanisms for Industry Applications*, no. Icimia, pp. 335–339, 2020.

[15] P. Khaire and P. Kumar, “A semi-supervised deep learning-based video anomaly detection framework using RGB-D for surveillance of real-world critical environments,” *Forensic Science International: Digital Investigation*, vol. 40, Mar. 2022, doi: 10.1016/j.fsidi.2022.301346.

[16] A. Berroukham, K. Housni, M. Lahraichi, and I. Boulfrifi, “Deep learning-based methods for anomaly detection in video surveillance: a review,” *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 1, pp. 314–327, Feb. 2023, doi: 10.11591/eei.v12i1.3944.

[17] S. Angadi, “Hybrid Deep Network Scheme for Emotion Recognition in Speech,” *International Journal of Intelligent Engineering and Systems*, vol. 12, no. 3, pp. 59–67, 2019, doi: 10.22266/ijies2019.0630.07.

[18] F. Zhang, Z. Li, B. Zhang, H. Du, B. Wang, and X. Zhang, “Multi-modal deep learning model for auxiliary diagnosis of Alzheimer’s disease,” *Neurocomputing*, vol. 361, pp. 185–195, Oct. 2019, doi: 10.1016/j.neucom.2019.04.093.

[19] M. Jyotiyana, N. Kesswani, and M. Kumar, “A deep learning approach for classification and diagnosis of Parkinson’s disease,” *Soft Comput*, vol. 26, no. 18, pp. 9155–9165, Sep. 2022, doi: 10.1007/s00500-022-07275-6.

[20] F. Rong, L. Shasha, X. Qingzheng, and L. Kun, “Object detection method on station logo with single shot multi-box detector,” *The Journal of Engineering*, vol. 2020, no. 13, pp. 607–614, 2020, doi: 10.1049/joe.2019.1213.

[21] J. Matas, O. Chum, M. Urban, and T. Pajdla, “Robust wide-baseline stereo from maximally stable extremal regions,” in *Image and Vision Computing*, 2004, pp. 761–767. doi: 10.1016/j.imavis.2004.02.006.

[22] A. Śluzek, “Śluzek - 2016 - Improving performances of MSER features in matching and retrieval tasks.pdf,” in *European Conference on Computer Vision*, Springer, 2016, pp. 759–770.

[23] G. Kosala, A. Harjoko, and S. Hartati, “Robust license plate detection in complex scene using MSER-Dominant Vertical Sobel,” *IAENG Int J Comput Sci*, vol. 47, no. 2, pp. 214–222, 2020.

[24] S. M. Pizer *et al.*, “Adaptive Histogram Equalization and Its Variations.,” *Comput Vis Graph Image Process*, 1987, doi: 10.1016/S0734-189X(87)80186-X.

[25] G. Tong, M. Dong, X. Sun, and Y. Song, “Natural scene text detection and recognition based on saturation-incorporated multi-channel MSER,” *Knowl Based Syst*, vol. 250, p. 109040, Aug. 2022, doi: 10.1016/j.knosys.2022.109040.

DOI: https://doi.org/10.29207/resti.v7i5.5034