



Fire Detection on Video Using ViBe Algorithm and LBP-TOP

Febryanti Sthevanie¹, Gamma Kosala², Kurniawan Nur Ramadhani³, Ketut Sudyatmika Putra⁴

^{1,2,3,4}School of Computing, Telkom University

¹sthevanie@telkomuniversity.ac.id, ²gammakosala@telkomuniversity.ac.id, ³kurniawanr@telkomuniversity.ac.id*,

⁴sudyatmikaputra@telkomuniversity.ac.id

Abstract

In this research, we built a system to detect fire using the ViBe (Visual Background Extractor) algorithm to extract dynamic targets. The ViBe algorithm is better at detecting moving target objects such as flame combustion. In this research we combined the ViBe algorithm with three frame differencing to gain better results on movement object. The HSI color space model was applied after the movement object was obtained. We used Local Binary Pattern-Three Orthogonal Planes to obtain the feature extraction to be classified with Support Vector Machine. Our result has shown that the proposed system were able to detect the fire using the LBP-TOP and ViBe algorithm methods with an average accuracy rate of 88.10%, and the best accuracy was 90.37%. The parameters used to achieve this accuracy in the feature extraction process were $T=120$, $Radius=2$, and $frame\ gap=15$, then the threshold value parameter for three-frame difference parameter was 25.

Keywords: fire detection, ViBe algorithm, local binary pattern-three orthogonal planes, support vector machine

1. Introduction

Fire is one of the disasters that causes very high losses, both loss of funds and lives. Most of the fire accidents are happen in crowded neighborhood areas. Such accidents can be prevented with early detection of the fire. In the current time, there are a lot of surveillance cameras or CCTVs are installed almost on all buildings or houses. A fire detection system can be installed on surveillance cameras for better cost efficiency instead of installing new devices for fire detection sensors. With the increasing usage of CCTV surveillance video systems for security purposes in monitoring industrial environment, public and general environment, and other environments, many people consider this fire detection system as the early fire detector[1]. The color, shape, flicker frequency, and motion characteristics of the flame image are analyzed in the early flame detection[2].

There are many methods has been used by the researcher to detect fire using video[1]–[7]. A fire detection system proposed by H. Yamagishi and J. Yamaguchi based on HSV color space and neural network, which uses hue and saturation to identify the fire field, is one of the earliest methods. However, due to the high computational complexity of the operation,

the duration of the process is not real-time[1]. There is also research in[6], using Local Binary Pattern-Three Orthogonal Planes (LBP-TOP) and Grey-level Co-occurrence Matrix (GLCM) which resulting the method used is not sensitive to the object movement and cannot fully detect the object visually. Research using hidden Markov models in [7] showing a result that it is efficiently detects flames. But it may produce false alarms because the method used is only based on color information and ordinary movement detection. It can be reduced using separate Markov models. Another method using background subtraction is proposed in [5]. This method needs to declare the background image first. However, using this background subtraction method has disadvantages. It cannot deal with the sudden, drastic light changes. In this research we overcame some of the previously mentioned drawbacks by implementing ViBe algorithm. ViBe algorithm is better on detecting moving target objects such as flame combustion, but the result from ViBe algorithm processing still has noise[8]. In order to solve that problem, we combined the ViBe algorithm with three frame differences to reduce noise, improve accuracy, increase image morphology processing, and reduce voids in the recognition results [8].

ViBe algorithm was proposed by Olivier Barnich and Marc Van Droogenbroeck according to their study in [8]. The word ViBe stands for "Visual Background Extractor" which is a foreground detection algorithm that uses random strategy to detect background sample estimation. ViBe algorithm has more benefits in terms of detection speed and model background update than optical flow method and Gaussian model algorithm [9]. ViBe algorithm also has high initialization speed, less memory consumption, and less resource utilization which can be a benefit for a surveillance camera that usually has low-performance specification. A research from the person who proposed the ViBe algorithm in [9] concludes from the experiment of seven other algorithms that the ViBe algorithm was proven to outperform all seven other background subtraction algorithm. ViBe algorithm was the fastest of those seven algorithms.

A research from [6] stated that by using Local Binary Pattern (LBP) for feature extraction method, the result is very sensitive against noise and has very high computational cost. Also, by using Volume Local Binary Pattern (VLBP) the result has complex computational and hard to extend. LBP-TOP is the better option amongst the other method such as the original LBP with Optical Flow and VLBP because both of those methods has high computational complexity. Therefore, LBP-TOP was used in this research. Another fire detection research in [10] is processed with testing video, fire color detection, and finding region of interest (ROI), LBP-TOP, and K-NN classification. The experiment of this research resulting in the LBP-TOP gives 92% of detection accuracy. But, in the early stages, some errors occurred around the ROI stages. Therefore, another color segmentation method is preferable.

For the first step of our research, we conducted a color space transformation from RGB format to HSI format. This improved the saturation identification and the average saturation of the image used instead of the originally fixed threshold, making it adapted to different light and dark environments to identify flames. Then, we combined the ViBe algorithm with three frame differencing method in order to reduce the noise, and other problems stated before. Then, LBP-TOP was applied as feature extraction. After that, we classified the features using SVM Classification. The previous research of fire detection according to [8] were processed with color segmentation, background subtraction, or moving target recognition and judge the roundness of the target graph. This previous research used RGB to HSI color rules and ViBe algorithm which was the reference of this research. Even though this previous research and method successfully detected fire efficiently, the judge of the target graph had disadvantages. This method may interfere with another object such as car lights, which has similar color with

flames and it is hard to differentiate it with flames when moving. In this research, the roundness of the target graph method was replaced with another feature extraction method called LBP-TOP.

The support vector machine (SVM) is a well-established classification algorithm. SVM conducts the classification process by constructing a hyperplane. SVM can be implemented on both linear and nonlinear classification problems [11]. According to research in [12], SVM classification had better accuracy and performance on LBP-TOP compared to LBP. SVM classification on LBP-TOP has higher testing accuracy at 96.54% compared to LBP which is 94%.

Our paper structured as follows. In section 2, we described the methods that we used in our research, from the HSI color conversion, background subtraction using ViBe algorithm, feature extraction using LBP-TOP and classification using SVM. In section 3, we showed the result of our experiment using the aforementioned methods. And section 4 explained our conclusion from this research.

2. Research Method

In this research, the objective is to detect fire when appears on the camera using ViBe algorithm and LBP-TOP. The system design consisted of four stages. First, three-frame differencing was used combined with ViBe (Visual Background Extractor) Algorithm to detect movement object, color segmentation using HSI color space model, feature extraction using Local Binary Pattern-Three Orthogonal Planes and after that classified the features using Support Vector Machine (SVM). In the classification stage, the process was split into 2 parts, the training and testing process. Before implement the testing, a training process was required to train the system. The process of training is described on figure 1 and for the testing process can be seen on figure 2.

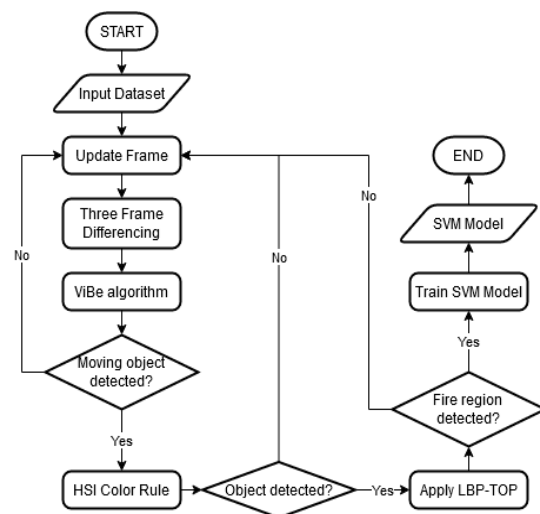


Figure 1. Training Flowchart

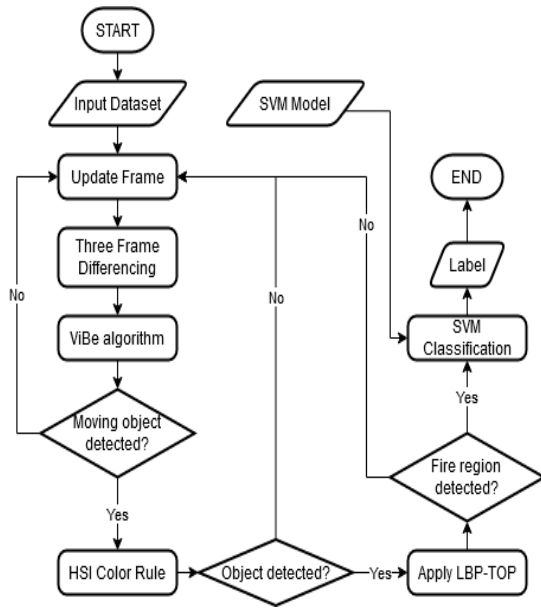


Figure 2. Testing Flowchart

2.1. ViBe Algorithm with Three Frame Differencing

The first step for ViBe algorithm is background subtraction by using the three-frame differencing method. The process of ViBe algorithm can be seen on figure 3. Three-frame differencing uses three frames to be processed. In this case, those 3 frames are $k-1$, k , and $k+1$. Each of those frames has five frame gaps to be processed. After the 3 frames are initialized, the difference between $k-1$ with k and k with $k+1$ is processed. From those differences, And operation is used to find the three-frame difference result.

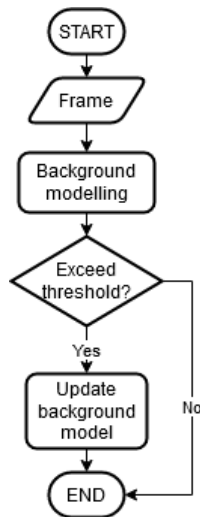


Figure 3. ViBe Algorithm Flowchart

The ViBe Algorithm will be applied to frame $k-1$ and frame k difference results and will be processed with the three-frame difference results with And operator. The process of three-frame differencing with the ViBe algorithm can be seen in Figure 4.

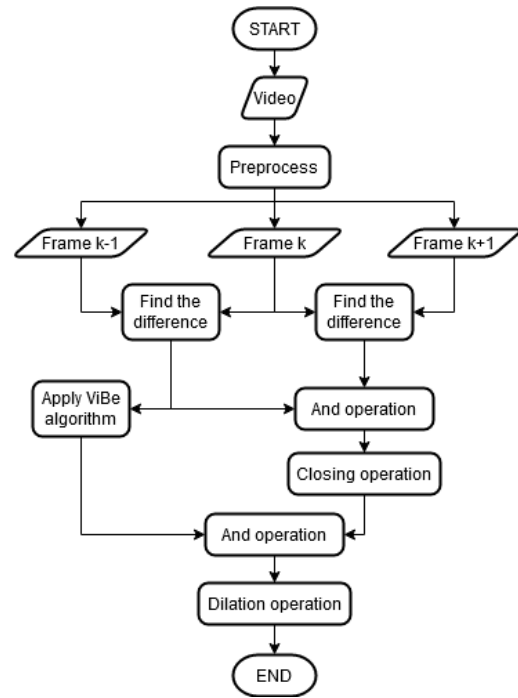


Figure 4. ViBe Algorithm with three frame differencing flowchart

Figure 5 shows the distance of the frame to be processed. We can see at a distance of 5 frames, moving objects are detected less than the others. This is because with fewer frames, changes in moving or moving objects that are detected will be less in accordance with the 5 frames and will be more accurate. However, processing time is slower due to more frames being processed. At a distance of 15 frames, the detected area is wider because within 15 frames, more moving or moving objects will be detected according to those 15 frames. The advantage of this 15 frame distance is that the processing time is faster than the 5 frame distance. However, the detected object area becomes wider and causes objects moving outside the fire position to be detected more.

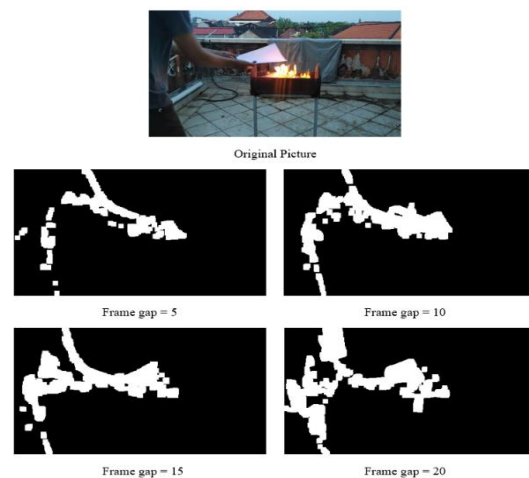


Figure 5. Frame Gap

2.2. RGB to HSI Color Space Model

Secondly, The color segmentation is done using RGB to HSI color space model method. The original input will have RGB color space model and will be converted into HSI color space model. Red (R), Green (G), Blue (B) is the component of RGB color space model. The flame is decomposed according to the RGB color model during detection, and the alleged flame area is calculated by examining the color variable. Since each color component of the RGB color space model often includes brightness detail, it is simple to misjudge the image of a complex background, so the HSI color model is combined in use [11]. The HSI color model is represented by hue (H), saturation (S), and brightness (I). The HSI color rules can be converted using formula (1) to (3)[13]:

$$H = \begin{cases} \theta, B \leq G \\ 360 - \theta, B > G \end{cases} \quad (1)$$

$$S = 1 - \frac{3}{R+G+B} \min(R, G, B) \quad (2)$$

$$I = \frac{1}{3}(R + G + B) \quad (3)$$

where,

$$\theta = \arccos \frac{\frac{1}{2}[(R-G)^2(G-B)]}{[(R-G)^2 + (R-G)(G-B)^2]^{\frac{1}{2}}} \quad (4)$$

2.3. LBP-TOP

From the chosen dynamic target processed from the background subtraction model in the previous step, the next process is to begin the feature extraction. Local Binary Pattern-Three Orthogonal Plane (LBP-TOP) will be used as the feature extraction method. LBP-TOP needs input as a volume data. Therefore, the previous frame until the next frame from three-frame difference is needed and created into a volume data[14]. Figure 6 illustrates the process of LBP-TOP and figure 7 shows the example of XT, XY and YT plane respectively.

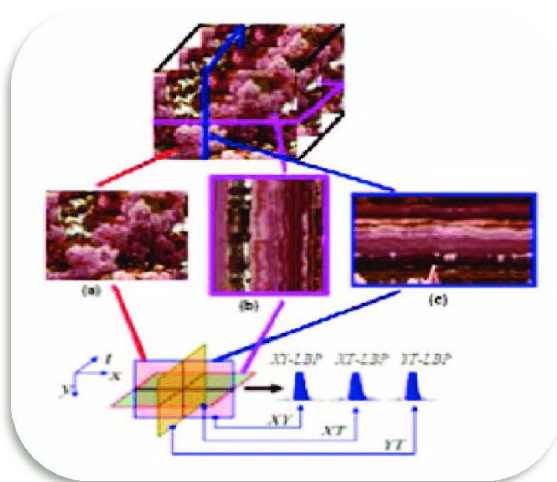


Figure 6. LBP-TOP process[12]



Figure 7. LBP-TOP plane

From the volume data obtained, focusing on the area of the flames, 3 planes are taken, namely the XY plane, the XT plane, and the YT plane. Plane XY is 2D data taken from the x-axis and the y-axis at the center value of the t-axis which represents spatial information, while XT and YT, respectively, are 2D data taken from the x-t axis with the center value of the y-axis and the y-t axis with the mean of the x-axis, that represent temporal information on the row (for XT) and the column (for YT). Each plane is converted to grayscale because LBP accepts data input in the form of grayscale. The process of LBP-TOP is described on figure 8.

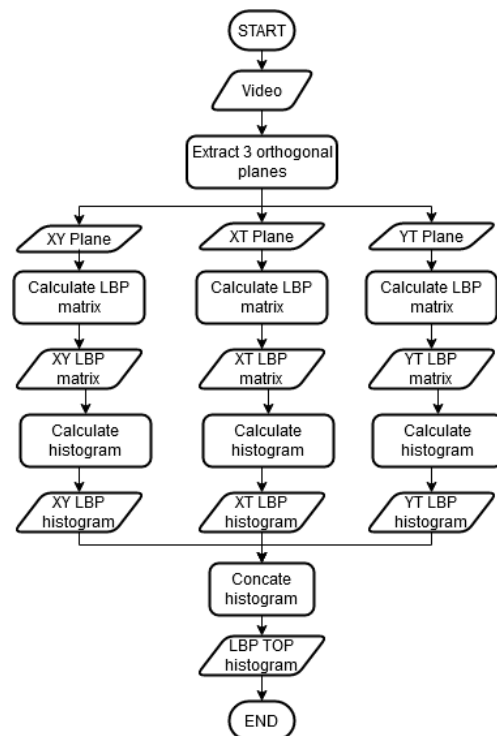


Figure 8. LBP-TOP flowchart

2.4. SVM Classification

Support Vector Machine (SVM) is selected as the classifier for the next process. SVM is known as high performance and accurate classification results with limited training dataset[15]. SVM can be simply explained as an attempt to find the best hyperplane in the input space that serves as a separator of two classes. SVM is unique in that it employs a neural network to find a hyperplane separator between classes. Hyperplane is a separator from 2 different data classes, on a hyperplane dimension called a point, in 2

dimensions is called a line, in 3 dimensions called the plane, and for higher dimensions only called hyperplane only[11]. Figure 9 illustrates the hyperplane constructed by SVM algorithm. In this research, we used liner kernel for SVM because it does not need a transformation process for the input, hence the classification process faster compared to other kernel function.

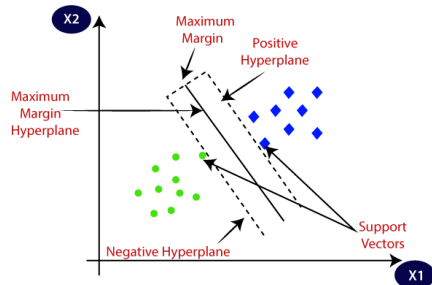


Figure 9. Support Vector Machine Hyperplane[11]

In this step, the process is split into 2 part, training and testing. The training and classification process uses the SVM model done with the product in between two data vectors (x) using a kernel function K . Figure 10 shows the process of training an SVM model.

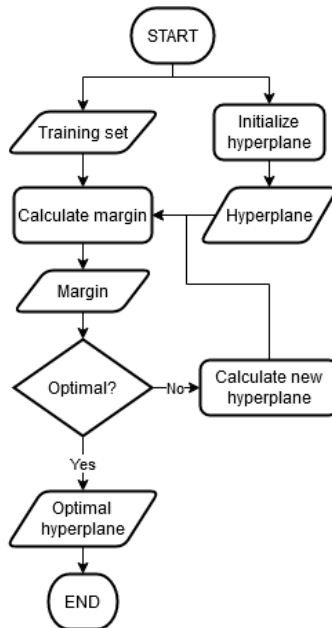


Figure 10. SVM flowchart

3. Results and Discussions

For evaluation metric, we used accuracy score in this research. Accuracy can be calculated as the total of True Positive (TP) and True Negative (TN) case divided by all of cases, whether it is true or false. True Positive (TP) means the case when a real fire is detected successfully by our system as fire and True Negative (TN) means the case when non fire object is not detected by our system as fire.

3.1. Experiment

Experiment I, Background subtraction threshold. The result can be seen on figure 11.

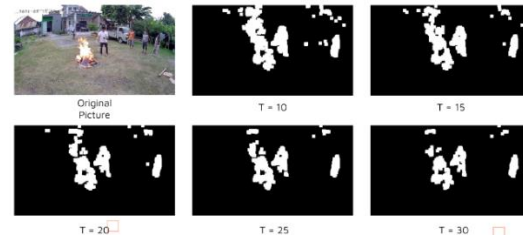


Figure 11. Background Subtraction Threshold

From the experimental results in Figure 11, using a threshold of 10, the results show that many areas were detected so that many unnecessary objects were masked. While at the threshold value of 30, the detected area is less but some important objects such as fire are not detected. In this study, the threshold used is 25, where fire objects can still be detected properly and unnecessary objects can be filtered properly.

Experiment II, ViBe Algorithm, 3 frame differencing, ViBe + 3 frame differencing result. The result is shown on figure 12.



Figure 12. Background subtraction result

The results from the image above show that the results of the ViBe algorithm are still not very complete. Therefore, the 3 frame differencing method is combined in this study.

Experiment III, HSI Color Rules.

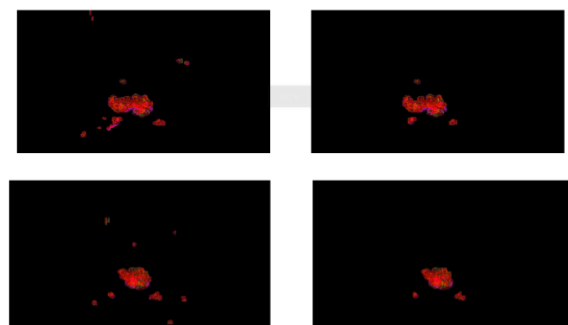


Figure 13. HSI color rules test (Left = wider range, Right = tighter range)

In Figures 11, it can be seen that the image on the left has a wider range, while the image on the right has a narrower range. The results above show that in a wider

range, many pixels outside the fire area are obtained. Whereas in a narrower range the fire area can be obtained better. The results show that with a narrower range, the fire area results can be obtained better. So, the narrower range of the color rules are used in this study.

For convenience of discussion and image processing, the hue, saturation, and intensity components of the HSI model are normalized into the ranges: $0^\circ < h < 180^\circ$, $0 < s < 255$, and $0 < i < 255$. The color rules used for saturation values are distributed in range [51, 255], hue values are in the range [0, 30], and intensity values are in range [100, 255]. This research mainly focuses on lower temperature fire flames. Higher temperature blue flames are not considered in this color rules[14]. For the wider range as shown on Figure 13, the value of the color rules are [20, 255] for saturation, [0, 30] for hue, and [50, 255] for intensity.

3.2. Test Result

In this study, there are 2 test scenarios. Here is the scenario:

1. Scenario I, LBP-TOP with SVM using linear kernel is tested. The result of scenario 1 experiment can be seen on table 1.

Table 1. LBP-TOP Test

Radius	T	Attribute				Accuracy
		TP	FP	TN	FN	
1	60	184	21	507	61	89.39%
	120	186	22	496	63	88.91%
2	60	207	39	490	42	89.58%
	120	211	29	484	45	90.37%
3	60	191	47	472	58	86.32%
	120	196	38	484	56	87.85%
4	60	142	35	489	104	81.94%
	120	173	25	489	82	86.31%

On the table above, it can be seen that the best result shown is LBP-TOP with radius parameter is 2 and T is 120. T is the number of stacked frames used as plane in LBP-TOP. So this parameter will be chosen to be process in the next scenario test.

2. Scenario II, Fire detection system using the chosen parameter from the passed in the previous scenario test. The result of scenario II can be seen on table 2.

Table 2. Fire Detection Test

Dataset	Attribute				Accuracy
	TP	FP	TN	FN	
Video 1	324	44	213	29	88.03%
Video 2	58	6	85	0	95.97%
Video 3	73	5	217	27	90.06%
Video 4	290	23	112	53	84.10%
Video 5	160	48	584	198	75.15%
Video 6	76	24	181	15	86.82%
Video 7	32	4	3	0	89.74%
Video 8	31	10	203	13	91.05%
Video 9	35	4	103	2	95.83%
Video 10	225	44	615	113	84.25%

In the table above, the test results show that the lowest level of accuracy is at 75.15% and the highest level of accuracy is at 95.97%. The test results use the best selected parameters from the previous test scenario. That is, Radius is 2, T is 120, and frame gap is 15.

3.3. Analysis of Test Result

The results of the overall system test above can be taken the average accuracy for videos containing fire objects is 88.10%. it has the greatest accuracy, which is 95.97% in Video 2 which is an ideal fire video with an indoor location, a little color similar to fire, and video shooting is done during the day. Then for the smallest accuracy, which is 75.15% in the Video 5, which is a fire video with an outdoor location, where there are many objects with colors that are similar to fire and many objects are found moving around the fire. The decrease in accuracy occurs due to many moving objects that have colors that pass by the fire color rule for example an arm is false detected as a fire in this scenaro test. Figure 14 shows the example of true and false detection result.



Figure 14. True and False detection (Left is True, Right is False)

4. Conclusion

In this research, we have built a system to detect fire using ViBe algorithm and LBP-TOP for feature extraction. The system detected fire objects quite well with an average accuracy rate of 88.10%. The best accuracy for LBP-TOP from the test results was 90.37%. The parameters used to achieve this accuracy in the feature extraction process were T=120, Radius=2, and frame gap=15, then the threshold value parameter for three-frame difference parameter=25. Moving objects detected by the three frame differencing method with the ViBe algorithm provided good results for detecting moving objects, especially fire objects. With the right morphology, a more intact moving object could be obtained. The C value in the SVM classification had no effect on the accuracy of this system because after testing the C value the change in the accuracy value was very small.

In this system, the minimum contour size to be processed into feature extraction and LBP-TOP was 125 pixels with the video datasets resolution is 480x270 pixels after resized. With this, the size of the fire would be more balanced on every video. However, this system sometimes cannot detect big size of fire that appear in the video. One of the reasons is the moving object detection process can't detect movement in the center of

the fire because it doesn't detect any pixel changes in that area. In future research, what will be done is experiment with various size of fire in the video and also determine what kind of fire that is detected.

Reference

- [1] P. Barmpoutis, P. Papaioannou, K. Dimitropoulos, dan N. Grammalidis, "A Review on Early Forest Fire Detection Systems Using Optical Remote Sensing," *Sensors*, vol. 20, no. 22, hlm. 6442, Nov 2020, doi: 10.3390/s20226442.
- [2] M. Hashemzadeh dan A. Zademehti, "Fire detection for video surveillance applications using ICA K-medoids-based color model and efficient spatio-temporal visual features," *Expert Syst. Appl.*, vol. 130, hlm. 60–78, Sep 2019, doi: 10.1016/j.eswa.2019.04.019.
- [3] A. Gaur, A. Singh, A. Kumar, A. Kumar, dan K. Kapoor, "Video Flame and Smoke Based Fire Detection Algorithms: A Literature Review," *Fire Technol.*, vol. 56, no. 5, hlm. 1943–1980, Sep 2020, doi: 10.1007/s10694-020-00986-y.
- [4] M. Torabian, H. Pourghassem, dan H. Mahdavi-Nasab, "Fire Detection Based on Fractal Analysis and Spatio-Temporal Features," *Fire Technol.*, vol. 57, no. 5, hlm. 2583–2614, Sep 2021, doi: 10.1007/s10694-021-01129-7.
- [5] H. Wu, D. Wu, dan J. Zhao, "An intelligent fire detection approach through cameras based on computer vision methods," *Process Saf. Environ. Prot.*, vol. 127, hlm. 245–256, Jul 2019, doi: 10.1016/j.psep.2019.05.016.
- [6] N. Alamgir, K. Nguyen, V. Chandran, dan W. Boles, "Combining multi-channel color space with local binary co-occurrence feature descriptors for accurate smoke detection from surveillance videos," *Fire Saf. J.*, vol. 102, hlm. 1–10, Des 2018, doi: 10.1016/j.firesaf.2018.09.003.
- [7] S. Ali, Md. H. Rahman, dan N. Bouguila, "Fire Detection in Images with Discrete Hidden Markov Models," dalam *Hidden Markov Models and Applications*, N. Bouguila, W. Fan, dan M. Amayri, Ed. Cham: Springer International Publishing, 2022, hlm. 81–101. doi: 10.1007/978-3-030-99142-5_4.
- [8] Y. Gao dan P. Cheng, "Full-Scale Video-Based Detection of Smoke from Forest Fires Combining ViBe and MSER Algorithms," *Fire Technol.*, vol. 57, no. 4, hlm. 1637–1666, Jul 2021, doi: 10.1007/s10694-020-01052-3.
- [9] D. Niharika dan J. Mohana, "Rule Based Method to Detect Fire and Compare the Accuracy and Precision with Vibe Method," dalam *2022 International Conference on Business Analytics for Technology and Security (ICBATS)*, Feb 2022, hlm. 1–5. doi: 10.1109/ICBATS54253.2022.9759061.
- [10] M. Jamali, N. Karimi, dan S. Samavi, "Saliency Based Fire Detection Using Texture and Color Features," dalam *2020 28th Iranian Conference on Electrical Engineering (ICEE)*, Agu 2020, hlm. 1–5. doi: 10.1109/ICEE50131.2020.9260659.
- [11] Y. Bazi dan F. Melgani, "Convolutional SVM Networks for Object Detection in UAV Imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 6, hlm. 3107–3118, Jun 2018, doi: 10.1109/TGRS.2018.2790926.
- [12] J. Olivares-Mercado *dkk.*, "Early Fire Detection on Video Using LBP and Spread Ascending of Smoke," *Sustainability*, vol. 11, no. 12, hlm. 3261, Jun 2019, doi: 10.3390/su11123261.
- [13] T. Oinosho, M. Kameyama, dan A. Taguchi, "Color Conversion Formulae between RGB Color Space and HSI Color Space for Color Image Processing," dalam *2021 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, Nov 2021, hlm. 1–2. doi: 10.1109/ISPACS51563.2021.9651118.
- [14] M. Saddique, K. Asghar, U. I. Bajwa, M. Hussain, dan Z. Habib, "Spatial Video Forgery Detection and Localization using Texture Analysis of Consecutive Frames," *Adv. Electr. Comput. Eng.*, vol. 19, no. 3, hlm. 97–108, 2019, doi: 10.4316/AECE.2019.03012.
- [15] V. K. Chauhan, K. Dahiya, dan A. Sharma, "Problem formulations and solvers in linear SVM: a review," *Artif. Intell. Rev.*, vol. 52, no. 2, hlm. 803–855, Agu 2019, doi: 10.1007/s10462-018-9614-6.