International Journal on Electrical Engineering and Informatics - Volume 7, Number 2, June 2015

Color-Texture Based Object Tracking Using HSV Color Space and Local Binary Pattern

Priyanto Hidayatullah and Miftahuddin Zuhdi

Computer Engineering and Informatics Department,
Bandung State Polytechnic, Bandung,Indonesia
priyanto@polban.ac.id, miftah.jtkpolban09@gmail.com

Abstract: In this paper the object tracking method using the combination of object's color and texture feature is presented. The HSV color components areused as color features and the texture feature is extracted using Local Binary Pattern. The Object is modeled into hue histogram, saturation histogram, value histogram, and texture histogram. In the tracking process, there is a process of color based object detection using the histogram back projection method using hue histogram, saturation histogram and value histogram from the object model. The process is continued with similarity measure between the texture histogram of object model with a texture histogram of the candidate objects using Bhattacharyya coefficient. The HSV, LBP and joint color-texture tracking methods are also tested for comparison. The experiment showsencouraging results where the proposed method highest average tracking rate is 94.98% and the lowest is 57,56%. The average tracking rate is 84.59%.The overall results show that the proposed method is better than the other tracking methods.The proposed method is also successful to track the object under several challenging conditions such as similar color or color distributionbetweenthe target object and the background, partial occlusion, changes in object orientation, and moving camera.

Keyword: Object tracking, HSV, Local Binary Pattern, histogram back projection.

1. Introduction

Object tracking is a process of analyzing video frames to determine position of an object in the frames [1].There are a lot of computers and mobile applications that implement object tracking such as face tracking in video camera, augmented reality, automated surveillance, traffic monitoring, vehicle navigation, etc. [1][3] [4][18] There are also some challenges that may occur in the process of tracking objects such as complex object motion, geometric transformation, occlusion, cluttered background, illumination, etc. [18] Currently, there are a lot of object tracking methods that have been proposed to overcome these challenges. Yilmaz et.al[1] separating those methods into three main categories, Point Tracking, Kernel Tracking, and Silhouette Tracking. The overview of those three categories are shown in Figure 1.

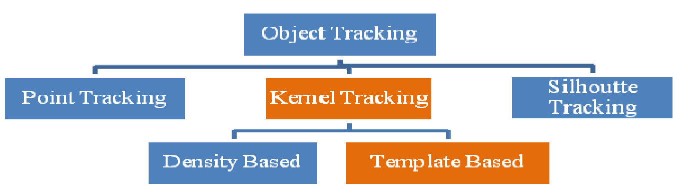


Figure 1. Overview of Object Tracking Method, focused on Kernel Tracking[18]

Among those three categories, kernel based tracking methods become very popular due to their simplicity and efficiency[1]. The most common approach in kernel tracking category is template matching [1].However, instead of using a template image, other object representations can be used for tracking, for instance using histogram[2][3] [4].Currently, color histogram is

Received: October 16th, 2013. Accepted: March 19th, 2015

the most common model used to represent the object. It is very robust in representing the object appearance[2]. Color histogram can be generated using properties of different color space like RGB, HSV, and others [5]. From those color spaces, RGB is the most commonly used. On the other hand, the HSV (Hue, Saturation, Value) color space is closely corresponds to the human visual perception of color [5]. Some researchers use only some of the HSV components [20] [21] but Hidayatullah et.al. [19] use all HSV components which give more robust results.

But there are a lot of cases which showusing color featureal one is still note nough to differentiate the target object from other objects. For example, when the target object has similar color or color distribution with image's background or with other objects, it can cause a mistake in tracking object.

One way to overcome this problem is by combining the color features with other features such as texture features [2] [4]. Texture feature describesthe surface and structure of anarea. Combination between color feature and texture feature would enrich the object information and can be used to increase the robustness of object's representation. There are some texture descriptors available such as Local Binary Pattern (LBP) [10], gray concurrence matrices [16] and Gabor filtering [17]. Unfortunately the last two methods need high computational complexity and cannot be directly used together with color histogram [2].

There are several researches about combining color-texture features in object tracking. Takala and Pietikainen [4] use color, texture and motion to do multi-object tracking in surveillance video which used a static camera. In the research, RGB color histogram and correlogram (auto correlogram) are exploited as color cues and texture properties are represented by Local Binary Patterns (LBP). The LBP is also used to detect moving object via background subtraction method. The tracking process is done by matching color, texture and motion features. Unfortunately, this method can work only in video with static camera when there is no changing in the video's background.

Ning et.al [2]usejoint color-texture histogram with mean shift framework [3]. Ning uses RGB color space for color features and RIU-LBP[10] to extract texture information. Those color and texture information are used to build a 4D histogram. The RIU-LBP is also used to form a mask for joint color-texture feature selection.

Maenpaa et.al [11] [12] stated that the color and texture features should be processed separately. However, their research is about image classification with color and texture features only and not directly related to object tracking.

Based on those researches, we propose an object tracking method which uses the combination of color and texture features that is processed separately. For color feature we use HSV color space which is more similar tohuman eye perception about color, and for texture descriptor we use the original LBP operator.

2. Local Binary Pattern

Local Binary Pattern (LBP) is an operator that can be used to extract texture information from the image [10]. The advantages of the LBP operator lie in it'saccuracy and computational complexity in many empirical studies [10].

Local Binary Pattern operator work by calculating each pixel's gray level with the graylevel of its neighborhood. The original version of the LBP operator is defined as follows:

$$LBP(x_c, y_c) = \sum_{P=0}^{p-1} s(g_p - g_c)2^p \tag{1}$$

Where:

$$s(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \tag{2}$$

g_c is the graylevel of center pixel (x_c, y_c), and g_p is the graylevel of its neighborhood. Figure 2 illustrates the process of calculating the pixel's LBP value.

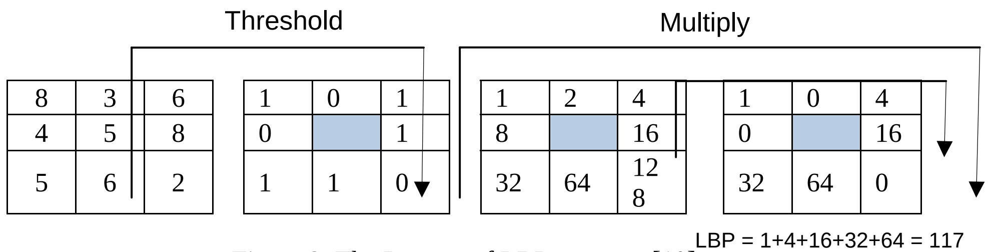


Figure 2. The Process of LBP operator [10]

3. HSV Color Space

HSV color space separating luminance component with chrominance component. The luminance component lies in V(intensity value) and the chrominance component lies in H(Hue) and S(Saturation). In HSV model, Hue defines the kind of color like red, yellow, green, blue, or the combination between those colors. Hue value is between 0-360. Saturation gives a measure of the degree by which a pure color is diluted by white light, and value show the intensity of light-dark color [5]. HSV color model is shown in Figure 3.

HSV color space is more similar to human eye perception about color and more ideal model to be used in color-based image and video processing algorithm [10].

4. The Proposed Method

In this paper, we process the color and texture feature in separate ways. First, we eliminate the incompatible pixel using color based object detection, and then we do texture matching to find the object position.

A. Object Selection

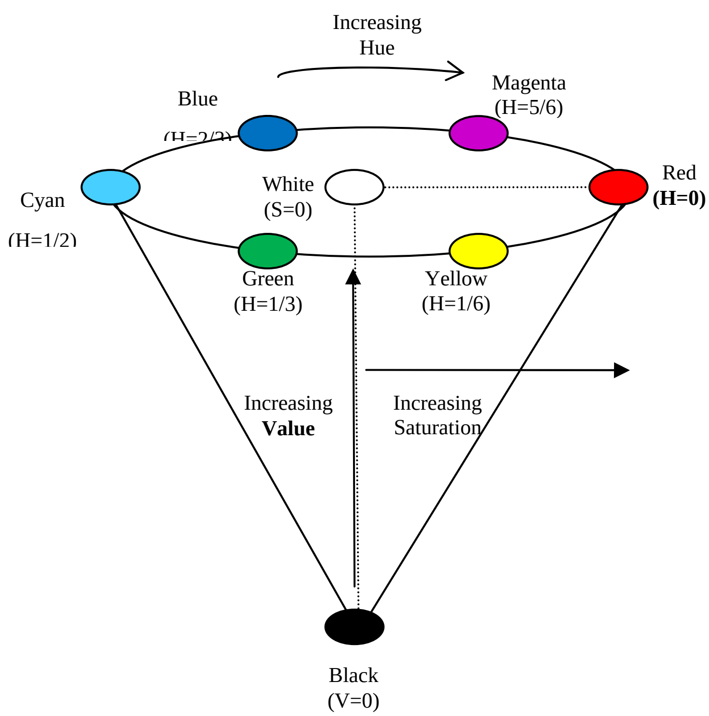


Figure 3. HSV Color Space [15]

The first step in this proposed method is object selection. In this step we use the most common method to do object localization by making a rectangle surrounding the object manually. The rectangle is used as an initial window for the next step. From the rectangle we can get information about the rectangle's upper left corner and bottom right corner coordinate as shown in Figure 4.

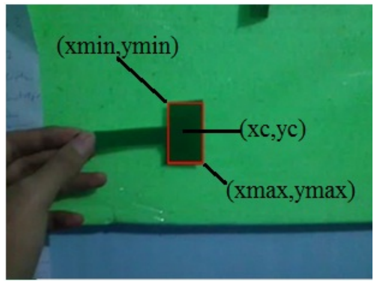


Figure 4. Information from the rectangle

From that information we can get information about the size of the initial window(rectangle) and the center box's coordinate (xc, yc).

$$width = (x\max - x\min) + 1;$$
$$height = (y\max - y\min) + 1;$$
$$xc = x\min + (width - 1)/2;$$
$$yc = y\min + (height - 1)/2;$$

B. Object Modeling

In this step, the object that has been chosen is modeled into color histogram and texture histogram. As mentioned before, HSV color space is used as color model. In this paper the HSV componentsare separated into three histograms: hue histogram, saturation histogram and value(intensity) histogram. We adopt Shural et.al [5] method that quantizes hue scale into 8 bins, saturation scale into 2 bins, and value (intensity) into 4 bins.

Texture histogram is computed from LBP image which is the result of texture extraction from the image using Equation 1. In this paper,the LBP histogram uses 256 bins.

The steps of this process can be summarized as below:
1. Convert RGB image from first frame into HSV image and LBP image.
2. Separate HSV image into hue image, saturation image and value image
3. Set image ROI on HSV's component images(hue, saturation, and value image) and LBP image based on the initial window from object localization step.
4. Calculate the color histogram from each HSV's component image and LBP histogram from LBP image.

C. Object Tracking

Our Tracking process consists of two steps: detection and tracking.

C.1. Detection

The goal of this step (object detection) is to eliminate pixels in the image that does not have the same color to the object model. Our detection process consists of several steps:

a. *Image Conversion*
The first step in this detection process is image conversion. In this step RGB frame is converted into HSV image and LBP image. This step is similar to image conversion step in object modeling process.

b. *Histogram Back Projection*
In this step we adopt and modify the method in [19] which separating each of HSV image components into hue image, saturation, image and value image and then do histogram back projection to each of those images using histograms that we get from the object modeling process. The difference is we use Hue instead of Hue-distance in [19]. This process is shown in Figure 5.

From this process we would have three new images, they are hue back projection image, saturation back projection image, and value back projection image.

Every pixel of those back projection images show the probability of the pixel whether included as the part of the object or not. If the pixel's value is high, it means the pixel is more probable to be part of the object.

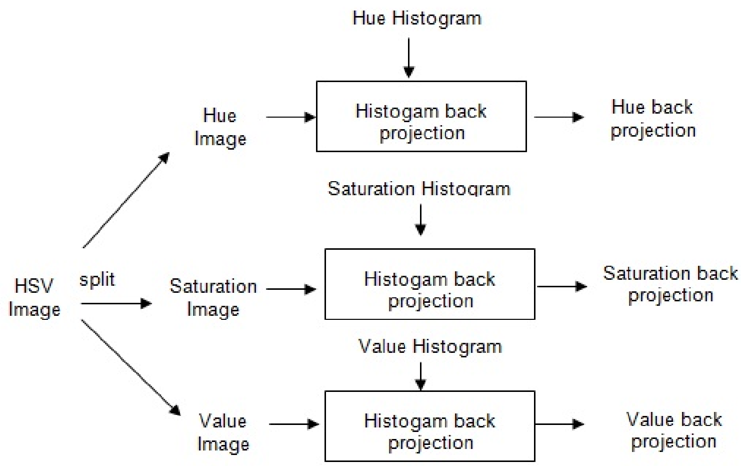


Figure 5. Separate the three components of HSV image

At the end of this step, those three back projection imagesare combined into one image using AND operator. This new image is the input for the next process.

c. *Masking LBP Image*
The next step is masking LBP image that we get from image conversion step using back projection image from the histogram back projection step. This process is the link between color based operation and texture based operation.
This process is carried out by checking each pixel's value in the back projection image. If pixel's value is less than the threshold, then the pixel's value in LBP image that have the same position with the checked pixel is changed into "0". In our experiment, we set the threshold to "255". The threshold is used to eliminate the pixel that hasa low probability as object's part. The result of this step is the input image for tracking step.
After this step, we start to switch from color based operation into texture based operation.

C.2. Tracking

In this step we do texture matching between object model's texture and candidate object's texture. This tracking step consistsof two steps: target localization and similarity measure.

a. Target Localization

To find the target object in the frame we do object localization process, we adopt the sliding window method [8]. But instead of using the original sliding window process which usesa brute force method and give a long time processing, we restrict the search region using this algorithm:

```
1.  While iteration <maxIteration, do
2.t= Z*iteration;
3.For  j (-1; j<2; j++:
4.For i←-1;i<2;i++ :
5.If (j !=0 &&i!=0) then
6.                    window(x,y)= window(x+(t*i),y+(t*j));
7. End If
8. End For
9.End For
10.  iteration=iteration+1;
11.End while
```

where $window(x, y)$ represents the center of the window. Using this algorithm, the window only slides into its 8 neighborhood's directions with the distance of Z pixel/iteration. With that, the search region is expanded along with the increase of the iteration. In this paper, we set the value of Z to 5 with 15 iterations. Sliding windowtoward its 8 neighborhood is illustratedin Figure 6 where the center box is the window's initial position.

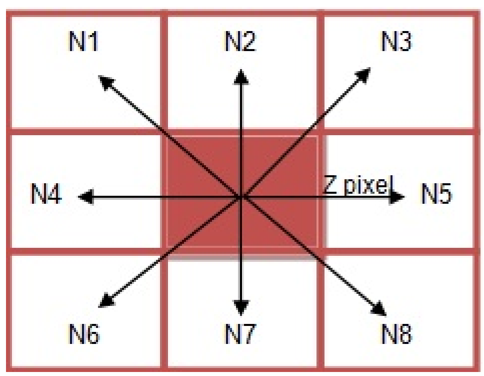


Figure 6. Sliding window

b. Similarity Measure

Similarity measure is one way to find out which object candidate histogram is the most similar to object model. Similarity measure can be carried out using Bhattacharyya coefficient [6] as defined below:

$$\rho(p, p') = \sum_{i=1}^{N} \sqrt{p(i)p'(i)} \qquad (3)$$

Bhattacharyya coefficient can be used to do similarity measure between two histograms where p(i) and p'(i) show the probability of bin i's value in histogram p and histogram p'[7].
In this process, Bhattacharyya coefficient is used to do texture similarity measure betweenthe texture histogram of the object model and texture histogram of the target object. Equation 3 would return the similarity value between 0..1, where the value of 1 means that the two compared histogramsare *perfectlymatched*. The most similar target's histogram is considered

as the representation of the object in current frame. The position of the target is marked as the object position using a rectangle. The complete algorithm for tracking step is summarized as follows:

Given:
- LBP image from masking process.
- Initial window from target localization step.
- Texture histogram of object model.
- bc=0, wp = position of the initial window

*bc = best similarity's value, *wp = position of the window's center

```
While iteration <maxIteration, do
t= z*iteration;
For j←-1; j<2 ; j++ :
For i←-1;i<2;i++ :
If (j !=0 &&i!=0) then
window(x,y)= window(x+(t*i),y+(t*j));
Set image ROI using window position
Calculate target's histogram from image ROI
Calculate similarity measure between histogram of texture's model and
target's histogram using equation 3
If   ρ(p, p') >bc, then
bc = ρ(p, p');
wp = w ;
End If
End If
End For
End For
iteration=iteration+1;
End while
Mark the position of the object using the rectangle based on wp.
```

The complete tracking algorithm is summarized in Figure 7:

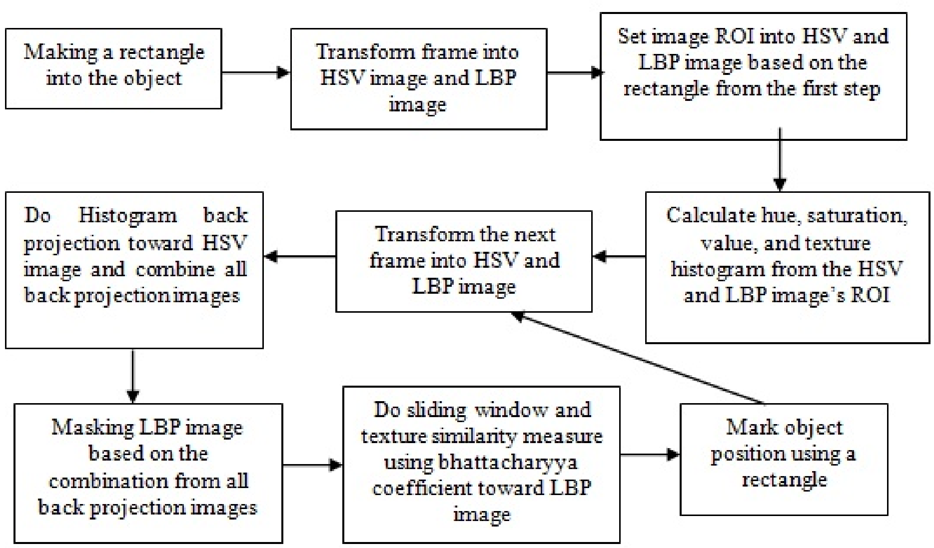


Figure 7. The schema of the proposed method

5. Experiment Setup

Our proposed method is implemented on Intel Pentium Dual-Core CPU T4300 @2.1GHz and 2.0GB RAM laptop. We use the Emgu CV [14] framework to help the implementation process. The list of test video is shown in the Table 1.

The object in the first video is a green paper that moves to the front of light green and old green background. The challenge of this video is the object's hue and texture are similar to the background's hue and texture..

In the second video we try to track a man's head that changes in orientation and encounters partial occlusion. The challenge in this video is when the head is partially occluded by the palm of the hand, which both have similar color but probably different texture.

In the third video we track the patterned paper that is blocked with another patterned paper that has similar color distribution but different pattern. This is to test the effectiveness of texture feature used in the tracking.

In the last video that we get from [13], we track the running cheetah in savanna with impala as the tracking distractor. The impala has a similar color to the cheetah, but has a different skin pattern. Grass that surrounding them is also part of the challenge. This is the most challenging test video. It tests all the features used.

Table 1. List of test video

Video	Size	Fps	Length(s)
Green Paper	480 x360	30	19
Head	480 x360	15	28
Patterned paper	480 x360	30	18
Running Cheetah	480 x360	15	27

In these experiments, we test the accuracy of 4 object tracking methods: using color only (HSV), texture only(LBP), joint color-texture, and the proposed method which separates the color and the texture. To test the accuracy of the methods, we create ground truth data for each video manually. Some examples of ground truth data are given in Figure 8. We calculate the tracking accuracy of the methods using overlap measure [9] between the ground truth data and our tracking results using Equation 4.

$$overlap = \frac{area(B1 \cap B2)}{area(B1 \cup B2)} \tag{4}$$

If the overlap measure's value is larger than 0.5 then it means the proposed method is successfully tracked the object in the current frame and categorized as true positive.

Figure 8. Some examples of ground truth data (frame 48, 161, and 376)

6. Results and Discussion

The color based object detection that carried out using histogram back projection in this proposed method successfully eliminate pixels that do not have the same color with the object model. The result in Figure 9 (d) shows that the process can separate pixels in the image into different colors. The black one means the pixel is not included as a part of the object, the gray color pixel means that the pixel has only a low probability as a part of the object, and the white pixel means that the pixel has a high probability as part of the object

(a) (b) (c) (d)

Figure 9. (a) Hue back projection image, (b) Saturation back projection image, (c) Value back projection image, (d) Combined of all back projection image

The use of three kinds of color histogram also give advantage. As shown in Figure 9 (a) and (b), if we only use hue or saturation histogram, it is still hard to differentiate between pixels that belongs to the object(green paper) and pixels that belongs to the background(old green background) because both have white color in the back projection image. The similar result is also shown if we use value histogramonly. Combining those three images gives big differentiation between pixels that belong to the object and those that are not.

The next process is masking LBP image with the combined back projection image. It helps to eliminate pixels that have a low probability as part of the object. Using 255 as the threshold means that we only take the pixel that have the highest probability and disregard others. The result ofthe masking process is shown in Figure 10.

(a) (b) (c)

Figure 10. (a) LBP image, (b) Combined all back projection image (c) Masked LBP image

As we can see in Figure 10 (c) that we only take the pixel that have the highest probability (white color in Figure 10 (b)) as part of the object. So even though there is also background's part that have been detected in histogram back projection process (Figure 10 (b)), it does not matter as long as the probability is below 255. From Figure 10 (b) we can also see that there are other objects that have high probability and taken into masked image.It means that the pixels have similar color with the object, even if the pixels belong to other objects.

Tracking process that use texture similarity process will help us to find the object location based on its similarity with the texture of the object model. Our target localization algorithm will set the candidate's location that is near with initial window's position. It can help us to reduce the search area and processing time. Bhattacharyya coefficient will show the similarity value's between those candidate targets. Candidate's location which has the highest value will be considered as the location of the object in the current frame. The illustration of this process is shown in Figure 11. We can see there are 5 examples of candidate's location with their similarity values. From those examples, the middle one (head) which has the highest value (0.838) is considered as the object in current frame. This process can help us to distinguish between objects that have been detected by previous steps and to find the most appropriate object.

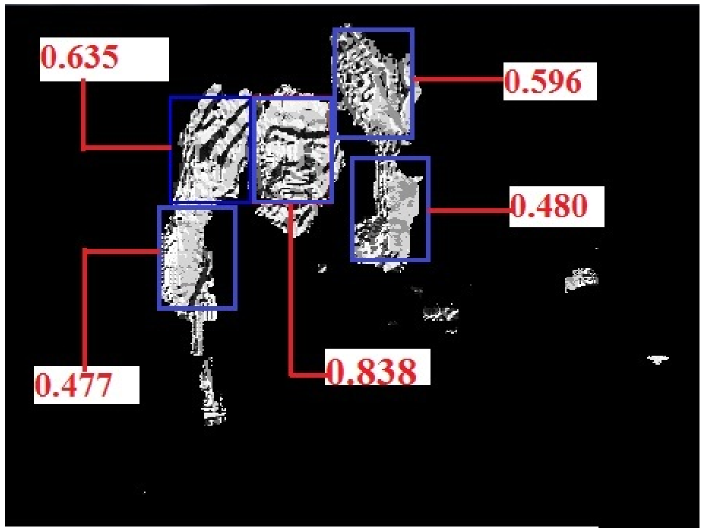


Figure 11. Texture similarity measure using Bhattacharyya coefficient

The first experiment on video 1 (green paper) showsthat color feature gives a very nice result (Figure 12) while texture feature failed to detect the object. It is very logical because the object has similar texture to the background. But we also note that hue component is not enough to detect the object because the object has a similar hue to the background. The result of the proposed method, HSV and joint color-texture are very good with insignificant difference.

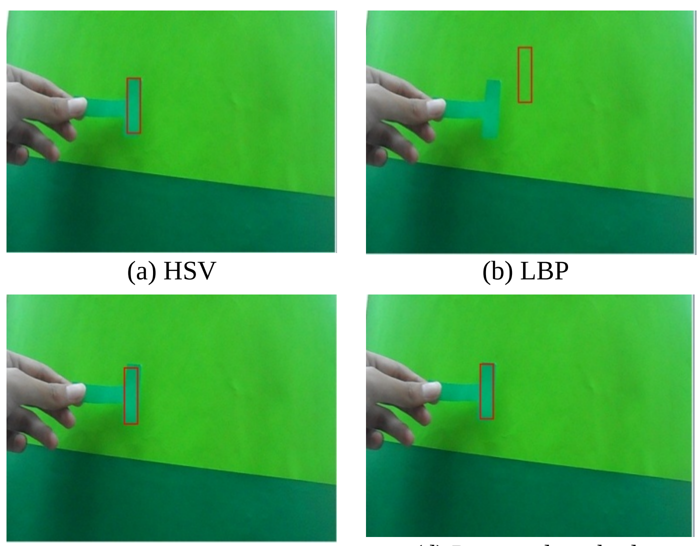

(a) HSV (b) LBP
(c)Joint color-texture (d) Proposed method

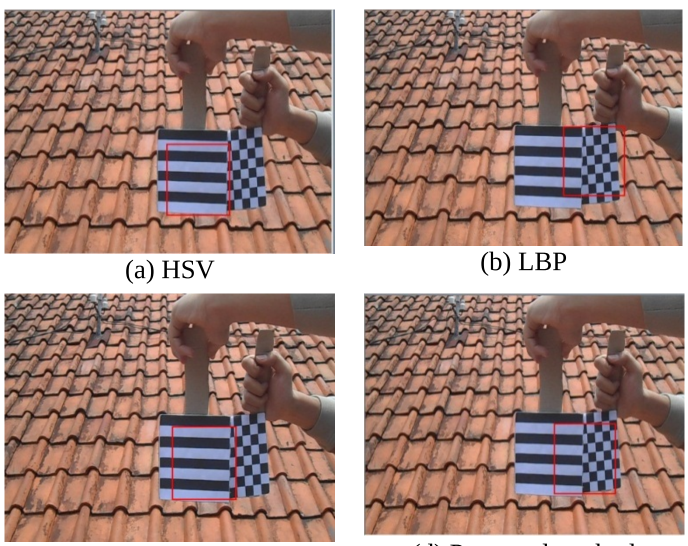
Figure 12. Tracking result of video 1 in frame 275

The result from the second experiment also shows an encouraging result. In this video (Figure 13) the proposed method, HSV and joint color-texture method successfully track the head even if the head changes its rotation and occluded by hand which has a similar color with the head.

Using color components makes head rotation not a problem. The use of all three components and texture feature makes the tracker robust to occlusion by hands which have similar color. However, LBP result is not very good because when the head changes its rotation, the texture is also changed which makes the tracker fails.

Figure 13. Tracking result of video 2 in frame 110, 206, 336

The object in the third video is a checkerpatterned paper. The paper is occluded with different patterned paper, buthas a similar color distribution. In this experiment, again our proposed method is able to track the object. The methods that use texture for one of its features gives better result, while the use of only HSV gives worse result. The HSV tracker sometimes failed to detect the object and track the occluder. This experiment result gives evidence that texture feature is better than color feature in this kind of video. The result is shown in Figure 14.



(a) HSV (b) LBP

(c) Joint color texture (d) Proposed method

Figure 14.Tracking result of video 3 in frame 522

In Figure 15, we can see the difference of texture similarity's value between object model with the two objects. The object on the right is the target object in the frameand the left one is the occluder. The target object gives the Bhattacharyya coefficient 0.874 while the left one gives only 0.707. This experiment shows that even if the two objectshave similar color distribution, the pattern is different which means the texture is also different. This helps the tracker to differentiate which one is the target object and which one is not.

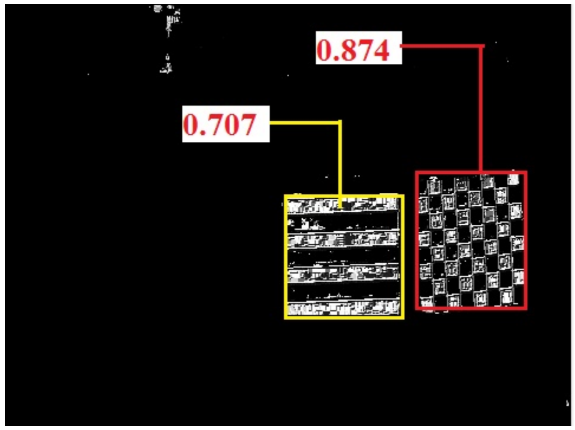


Figure 15. The Difference of texture similarity measure's value between the two objects.

In the fourth video,all of the methods tend to fail to track the cheetah. Our proposed method succeeds to track the cheetah only in 57.56% frames of the video. There are some failures in tracking such as in Figure 16 frame 331. This is due to the similar color between the cheetah and the impala. The other reason is because the cheetah moves too far which makes the cheetah skin's texture becomes not clear enough and reduce its texture similarity to the object model.

Figure 16. Tracking result of video 3 in frame 38, 171, 331

The detail result of the experiments is shown in Table 2.

Table 2. The tracking rate result

Video	HSV	LBP	Joint	Proposed Method
Green paper	96.32	3.84	96.32	94.98
Head	92.53	66.50	94.93	89.39
Patterned paper	72.87	78.01	74.64	89.36
Running cheetah	51.95	30.48	54.87	57.56

From Table 2, we can see that the proposed method givesoverall better result from the other methods. The tracking rate is 89.36% up to 94.98%, except on the fourth video that only reached 57.56% due to the reason that we have stated before. The overall average tracking rate is 84.59%.

7. Conclusion

In this paper, we have proposed an object tracking method using a combination of color and texture feature that processed in separated way. We use HSV color space and texture feature that is extracted using the LBP operator. In the proposed method,the objectis modeled into hue, saturation, value, and texture histogram. The tracking process consistsof color based object detection using histogram back projection and continued with kernel based object tracking using template matching that does texture similarity measure using Bhattacharyya coefficient.

The proposed method shows encouraging results with the highest tracking rate is 94.98% and the lowest is 57.56%. The average tracking rate is 84.59%. The overall results show that the proposed method is better than HSV, LBP and joint color texture tracking method. The proposed method is also able to track the object in several challenging conditions such as when the object has a similar color to the background, the object has similar color distribution with other objects, the object is occluded by partial occlusion, the object changes its orientation, and the object is recorded with moving camera.

8. Acknowledgement

We would like to thank to Mr Dewa Gede Parta, Mr Setiadi Rachmat and Mr Iwan Awaludin for meaningful feedbacks and comments

9. References

[1]. Yilmaz, O. Javed, M. Shah. 2006. “Object tracking: a survey”, *ACM Computing surveys*, vol. 38, no. 4, pp.1-45.
[2]. J. Ning, L. Zhang, David Zhang and C. Wu. 2009, “Robust Object Tracking using Joint Color-Texture Histogram”. *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 23, No. 7 (2009), World Scientific Publishing Company 1245–1263.
[3]. Comaniciu, d., Ramesh,v.,and Meer, p. 2003. *Kernel-based object tracking.IEEE Trans. Patt.Analy. Mach. Intell*. 25, 564–575.
[4]. Takala, V, Pietikainen, M. MultiObject Tracking Using Color, Texture and Motion. University of Oulu, Finland.
[5]. Vadivel, A., Majumdar, A.K., & Sural, S.2005.Histogram Generation from the HSV Color Space Encyclopedia of Information Science and Technology, 1333-1337.
[6]. A. Bhattacharyya, “On a measure of divergence between two statistical populations defined by their probability distributions,” Bull. Calcutta Math. Soc., no. 35, pp. 99–110, 1943.
[7]. Aherne, F., Thacker, N. &Rockett, P. 1997. The Bhattacharyya metric as an absolute similarity measure for frequency coded data Kybernetika, 32(4), 1–7.
[8]. C. H. Lampert, M. B. Blaschko, and T. Hofmann. Beyond sliding windows: Object localization by efficient subwindow search. *In IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.
[9]. M. Everingham, L. Van Gool, C. Williams, J. Winn, and A. Zisserman. 2010. The pascalvisual object classes (VOC) challenge. *International Journal of Computer Vision*, 88(2):303–338.
[10]. Mäenpää, Topi.2003. The Local Binary Pattern Approach To Texture Analysis. University of Oulu, Finland.
[11]. Mäenpää, T., dan Pietikäinen. M.2003. Classification with color and texture: jointly or separately?. University of Oulu, Finland.
[12]. Mäenpää, T., Pietikäinen,M., danViertola,J. Separating Color and Pattern Information for Color Texture Discrimination. University of Oulu, Finland.
[13]. Youtube.CHEETAH CHASING-KILLING IMPALA. www.youtube.com/watch?v=RiorXTpV08Q
[14]. EmguCV. EmguCV Main Page.http://www.emgu.com/wiki/index.php
[15]. NMT. Introduction to Color. http://infohost.nmt.edu/tcc/help/pubs/colortheory/web/hsv.html.
[16]. C. C. Gotlieb and H. E. Kreyszig, Texture descriptors based on co-occurrence matrices,Comput. Vis. Graph. Imag. Process 51(1) (1990) 70–86.

[17]. G. Wouwer, P. Scheunders and D. Dyck, Statistical texture characterization from discrete wavelet representations, *IEEE Trans. Imag. Process*.8(4) (1999) 592–598
[18]. Godec, Martin.2008. Robust Object Tracking Using Semi-Supervised Online Boosting. University of Technology. Erzherzog-Johann-Universitat.Austria.
[19]. Hidayatullah, P.; Konik, H., "CAMSHIFT improvement on multi-hue and multi-object tracking," *Electrical Engineering and Informatics (ICEEI), 2011 International Conference on* , vol., no., pp.1,6, 17-19 July 2011
[20]. Bradski, G. R. “Computer Vision Face Tracking for Use in aPerceptual User Interface”. Intel Technology Journal, 2(2), 13-27,1998
[21]. J. A. Corrales, P. Gil, F. A. Candelas, F. Torres. “Tracking based onHue-Saturation Features with a Miniaturized Active Vision System”.In Proceedings Book of 40th International Symposium on Robotics,Asociación Española de Robótica y Automatización Tecnologías de la Producción – AER-ATP, Barcelona, Spain. pp.107, 2009.

Priyanto Hidayatullah was born in Garut Indonesia in 1981. He received Bachelor degree from Bandung Institute of Technology in 2004 and a double degree M.Sc. from Universite Jean Monnet France and University of Eastern Finland in 2010. Currently he is a lecturer, writer and researcher in Computer Engineering and Informatics Department, Bandung State Polytechnic, Indonesia. His research interestare image processing, computer vision, machine learning and especially biomedic.

Miftahudin Zuhdi was born in Cirebon Indonesia in 1991. He obtained Bachelor degree from Computer Engineering and Informatics Department, Bandung State Polytechnic, Indonesia in 2013. Currently he is a freelance researcher and developer with main research interest are image processing, computer vision, and machine learning.