

A Method of Scene Boundary Detection for Indexing Video Data Efficiently

Seok-Woo Jang, Sung-Youn Cho

Abstract: In a cell animation, a background scene is presented with one cell. When animation scenes are changed, a relatively large scene change occurs because their backgrounds are also changed. Unlike a camera-based real movie, a cell animation is made in the way of manual drawing. For this reason, such animation does not use many colors. In order to apply the characteristics of animations as most as possible and detect a scene change of a cell animation effectively, this study proposes a new scene change detection technique with the stepwise use of the color and block-based histograms. The proposed algorithm receives continuous animation images as input, changes the RGB color space into the HSI color space, executes the difference operation of color values of the two images, and thereby primarily determines whether the neighboring images are scene change candidates. If they are judged to be scene change candidates, the color histogram for each sub-region is made, and then a weight value is applied to finally determine whether a scene change occurs. To compare the proposed scene change detection method with conventional scene change detection methods, the block-based scene change detection method and the histogram-based scene change detection method as conventional ones were implemented. In this study, to qualitatively evaluate the performance of the proposed method of detecting scene boundaries, the accuracy measure was defined. The two conventional algorithms are not complicated relatively and are widely used to extract a scene change. The experiment of this study reveals that the proposed method more accurately detects an animation scene change than other conventional methods. The proposed animation scene boundary detection method is expected to be applied usefully to the fields of digital video data indexing and retrieval, dynamic motion analysis, and other related areas.

Keywords: Efficient indexing, Dynamic image, Color space, Histogram, Weighting factor.

I. INTRODUCTION

With the rapid development of the super-high speed-based wired and wireless Internet, smartphones, tablet PCs, and other popularized technologies and devices, there have been more demands for various types of video contents to create a new added-value in the entertainment area including movies, advertisements, and games [1-5]. Of these contents, animations are most popular with children, and young and middle-aged adults so that they are widely distributed [6-7].

Generally, an animation is a creative work directly drawn by a human. Depending on an animation creator, it is possible

Revised Manuscript Received on July 22, 2019.

* Correspondence Author

Seok-Woo Jang, Department of Software, Anyang University, Anyang, South Korea. Email: swjang7285@gmail.com

Sung-Youn Cho*, Department of Software, Anyang University, Anyang, South Korea. Email: scho@anyang.ac.kr

to make different shapes of main characters. Nevertheless, their overall structures follow a basic form so that they are mostly equal. Usually, animation is classified into model animation, 3D animation, and cell animation [8].

A model animation is created in the way of making objects with clay or earth and then capturing the motions of the objects. A 3D animation is created with 3D computer graphics. A cell animation, which is often watched on TV or in cinema, is created with multiple cells to express backgrounds, main characters, and objects. In particular, cell animation is often used to reduce the time, human resources, and cost of producing animation compared to other methods.

Animations are on the exponential rise with a lapse of time. Therefore, it is necessary to develop the image processing technique to manage and index animation contents effectively. Above all, for the systematic management of animations, the detection of animation scene change should be preceded. Generally, scene change is classified into abrupt scene change (cut) and into gradual scene change (e.g, fade-in, fade-out, dissolve, wipe) [9-12].

The conventional methods of detecting general scene changes are introduced in the references. In the research [13], the different values of the pixel unit are analyzed so as to detect a boundary between scenes. In other words, if the number of pixels that have a different pixel value or color value between two neighboring images is greater than a threshold, it is judged that a scene change occurs. However, this method is sensitive to noise or motion information. In the studies [14, 15], the histogram of contrast value or color value for an image is made; if the bin difference of the histograms for two neighboring images exceeds a defined threshold, it is recognized that a scene change occurs. This method is known to work well on the whole. However, if there are two different scenes with similar distribution of contrast values, the method can make an error. The study [16] designed the method of detecting multiple scene changes by using the ratio of new edges to existing and disappearing edges. However, if a bright value is changed by the influence of light, the method considerably lowers its accuracy of edge detection. In this circumstance, the method is known to fail to detect a scene change robustly. In the studies [17], the proposed method is to split an image in the unit of the $N \times N$ square block with equal size and shape, and then to extract motion information in the block unit. Subsequently, the method detects a scene change with the use of a motion information difference. This method is known to make more robust detection than the frame-unit based comparison method if a local or global motion exists. However, it makes an error if a contrast value changes.

Generally, in a cell animation, a background scene is presented with one cell. When a scene change occurs, the background is also changed. As a result, a relatively large change occurs at the time of the scene change. Unlike an animation made with a camera, a cell animation is created through manual drawings so that it does not use many colors and most scene changes are made with cut. Therefore, in order to apply these characteristics as most as possible and detect scene changes of a cell animation more effectively, this study proposes a new scene change detection technique of animation with the stepwise use of the color and block-based histogram.

In Chapter 2, the method of detecting scene change candidates is described. In Chapter 3, we explain the method of detecting a scene change of animation in the way of splitting an image into sub-regions and making the color histogram in the split region unit. In Chapter 4, we present the experimental results for the performance evaluation of the proposed method. In Chapter 5, we suggest the study conclusion and future research direction.

II. DETECTION OF SCENE BOUNDARY CANDIDATES

Fig. 1 illustrates the overview of the proposed scene change detection system of animation with the stepwise use of the color and block-based histogram.

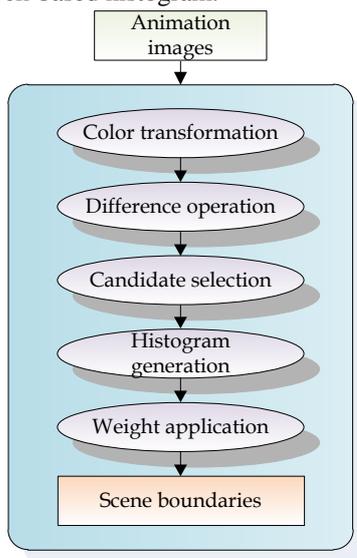


Fig. 1. Overview of the proposed method

As shown in Fig. 1, the proposed algorithm receives continuous animation images as input, changes color space into the HSI space, executes the difference operation of color values of the two images, and thereby primarily determines whether the neighboring images are scene change candidates. If they are judged to be scene change candidates, the color histogram for each sub-region is made, and then a weight value is applied to finally determine whether a scene change occurs.

The proposed scene change detection algorithm first changes the RGB color space of the input image into the HSI color space. Usually, the RGB color space is used for coding a color image. Since all three color values R, G, and B need to be processed for image processing, it is inefficient to use the

RGB color space. In fact, the RGB color space is sensitive to a bright change of an image, and a color value for the same region of the same image can be very different depending on a light size or direction. For this reason, the RGB color space is known to be inappropriate for image processing [18-20]. Therefore, this study makes use of HIS color space rather than the RGB color space. Equation (1) is used to change the RGB color space to the HSI color space. In (1), H (hue) has an angle ranging from 0 to 360 degrees; I (intensity) has a value ranging from 0 (black) to 255 (white); S (saturation) has a value between 0 and 1, meaning a degree of color purity [21-23].

$$H = \cos^{-1} \left(\frac{[(R-G) + (R-B)]}{2\sqrt{(R-G)^2 + (R-B)(G-B)}} \right) \quad (1)$$

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

$$S = 1 - \frac{3}{(R + G + B)} \times [\min(R, G, B)]$$

Fig. 2 visually shows the structure of the HSI color space.

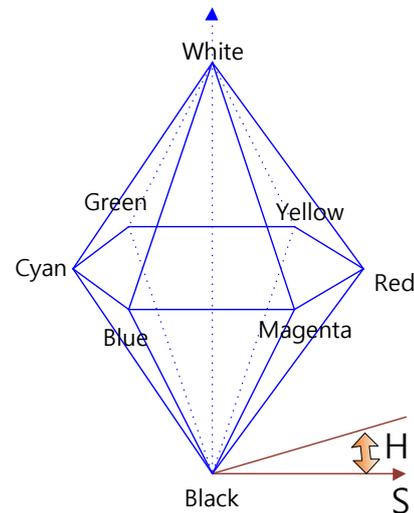


Fig. 2. Structure of the HSI color space

In order to detect scene change candidates primarily, this study uses the color scale as shown in (2).

In (2), $H_t(x, y)$ and $H_{t-1}(x, y)$ are the images with the extracted H value only after the color space of image at the t and t-1 points of time was changed to the HSI space. Accordingly, $DoH_{t,t-1}(x, y)$ is the image created after difference operation was applied to the two images. After the change from the RGB color space to the HSI color space, only H was used. Therefore, the value is expressed between 0 and 360. After difference operation is applied to the color value '1' and the color value '360', the result becomes the large value '359' even if the colors are almost equal. To correct the phenomenon, $NoH_{t,t-1}(x, y)$ is generated as shown in (2). The generated $NoH_{t,t-1}(x, y)$ are then added up for all pixels so as to create the color scale. If the color scale value outnumbers a given threshold [24-26], the images are judged to be the primary scene change candidates of animation.

$$\Phi_{t,t-1}(x, y) = \sum_{x,y} NoH_{t,t-1}(x, y) \quad (2)$$

$$NoH_{t,t-1}(x, y) = \begin{cases} DoH_{t,t-1}(x, y) & \text{if } DoH_{t,t-1}(x, y) \leq 180 \\ 360 - DoH_{t,t-1}(x, y) & \text{if } DoH_{t,t-1}(x, y) > 180 \end{cases}$$

$$DoH_{t,t-1}(x, y) = |H_t(x, y) - H_{t-1}|$$

III. VERIFICATION OF ANIMATION SCENE BOUNDARIES

Generally, unless scenes change in an animation, neighboring images have almost the same number of colors and the regions of the colors are similar. Unlike a real life film, an animation has a clear difference of colors for easy human identification. Therefore, the conventional scene change detection methods using color histogram often mistake the same scenes of animation for a scene change. For example, in case of the scenes that have the same background, if there is a relatively large area of characters and there are many moves in one frame, such phenomenon can easily occur. In particular, a character is completely rotated in one frame, it is possible to cause malfunction often.

To solve the problem, this study uses two techniques. Firstly, the animation image judged to be a candidate with a high probability of scene change occurrence is received, and then the entire image is split by the 3x3 square block. After that, the histogram of the color H for each block is made. At this time, since the H value is between 0 and 360 degrees, the proposed method of this study divides 360 degrees into eight equal parts each of which has 45 degrees, and makes the color histogram [27-30] for each split range, as shown in Figure 3. In the figure, the length of the arrow means the cumulative sum of color values in the eight-split range.

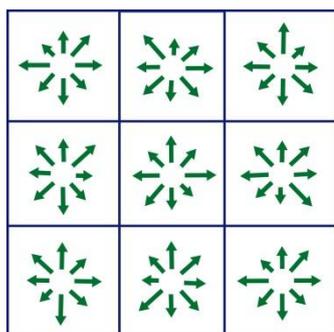


Fig 3. Color Histogram

As presented in (3), the sum of the histogram differences [31-33] obtained from the neighboring two images is calculated for each one of 9 blocks. In (3), $H_t^k(i)$ represents the histogram for the k-th block of the image at the time of t.

$$SoH_{t,t-1}(k) = \sum_{i=1}^8 |H_t^k(i) - H_{t-1}^k(i)| \quad (3)$$

Secondly, in an animation, a main character is usually located at the center of an image, and the background scene is at its rest regions. As shown in Fig. 4, a given weight value [34-36] is applied to $SoH_{t,t-1}(x, y)$ which is the sum of the

histograms of the split regions.

3	2	3
2	1	2
3	2	3

Fig 4. Weight value

Equation (4) presents the process. $M(t, t-1)$ becomes the scale of determining whether neighboring images have a scene change. If the value of $M(t, t-1)$ outnumbers a given threshold, a scene change is judged to occur. Otherwise, it is judged that there is no scene change.

$$M(t, t-1) = \sum_{k=1}^9 SoH_{t,t-1}(k) \times w(k) \quad (4)$$

IV. EXPERIMENTAL RESULTS

In this study, the computer used for an experiment has the following specification: Intel Core(TM) i7 2.93 Ghz CPU, 8GB main memory, and Windows 7 operating system. To implement the proposed algorithm, Microsoft Visual Studio and OpenCV as an open library were used. The three animations used for the experiment had the resolution of 640x480, 23.976 frames per second, and 24 minutes of running time.

To compare the proposed scene change detection method with conventional scene change detection methods, the block-based scene change detection method [17] and the histogram-based scene change detection method [14] as conventional ones were implemented. The two conventional algorithms are not complicated relatively and are widely used to extract a scene change.

The animations used for this experiment had a total of 103,548 frames, and the detected scene changes had 792 frames. As shown in (5), the accuracy of scene change was measured with the percentage as to whether or not a scene change was accurately detected.

$$SoH_{t,t-1}(k) = \sum_{i=1}^8 |H_t^k(i) - H_{t-1}^k(i)| \quad (5)$$

According to the experiment, the conventional methods had the scene change detection accuracy rate of an animation similar to that of a real movie if an animation was complicated as much as a real movie; they lowered the accuracy if an animation was simple.

Fig. 5 illustrates the graphs of the accuracy of the animation scene change detection results drawn by the conventional methods and the proposed method. As shown in Fig. 5, the proposed scene change detection method had better performance than the conventional detection methods.

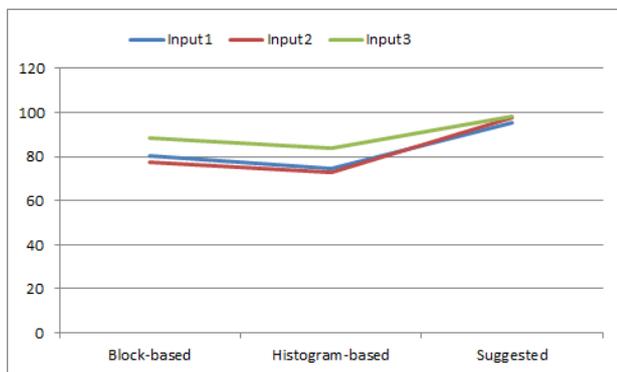


Fig 5. Performance evaluation

V. CONCLUSION

In the entertainment area including movies, games, UCC, and IPTV, there have been more demands for various types of contents for creating a new added-value. Of these contents, animations are most popular with children, and young and middle-aged adults so that they are widely distributed. In this circumstance, it is necessary to develop the technique to manage and index animation contents effectively.

This study newly proposed the algorithm of detecting scene boundaries in an animation with the stepwise use of the color and region-based histograms. The proposed system receives continuous animation images as input, changes color space into the HSI space, executes the difference operation of color values of the two images, and thereby primarily determines whether the neighboring images are scene change candidates. If they are judged to be scene change candidates, the color histogram for each sub-region is made, and then a weight value is applied to finally determine whether a scene change occurs. The experiment of this study reveals that the proposed method more accurately detects an animation scene change than conventional methods.

The future research is to evaluate the performance of the proposed method with the use of different animations, to stabilize multiple parameters used in the proposed algorithm, and to find a plan for automation. In addition, we will make efforts to improve the performance of animation scene change detection by combining the features used in this paper and other good features.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2019R1F1A1056475).

REFERENCES

1. Y. Yang, L. Shen, H. Yang, P. An, "A content-based rate control algorithm for screen content video coding," *J. Vis. Comm. Image Represent.*, vol. 60, 2019, pp. 328-338.
2. J. Guna, G. Gersak, I. Humar, J. Song, M. Pogacnik, "Influence of video content type on users' virtual reality sickness perception and physiological response," *Future Gener. Comp. Sy.*, vol. 91, 2019, pp. 263-276.
3. S. Wazarkar, B. N. Keshavamurthy, "A survey on image data analysis through clustering techniques for real world applications," *J. Vis. Comm. Image Represent.*, vol. 55, 2018, pp. 596-626.

4. G. Horsman, "Reconstructing streamed video content: a case study on YouTube and Facebook Live stream content in the Chrome web browser cache," *Digit. Invest.*, vol. 26, 2018, pp. s30-s37.
5. Z. A. Sohi, S. A. Torabi, "Integrated home video content procurement and distribution planning under uncertainty," *Comput. Ind. Eng.*, vol. 106, 2017, pp. 329-337.
6. F. C. Izzo, A. Carrieri, G. Bartolozzi, H. V. Keulen, M. Picollo, "Elucidating the composition and the state of conservation of nitrocellulose-based animation cells by means of non-invasive and micro-destructive techniques," *J. Cult. Herit.*, vol. 35, 2019, pp. 254-262.
7. Z. Liu, L. Zhou, H. Leung, H. P. H. Shum, "High-quality compatible triangulations and their application in interactive animation," *Comput. Graph.*, vol. 76, 2018, pp. 60-72.
8. W. F. Koning, "Teaching 3D computer animation to illustrators: the instructor as translator and technical director," *IEEE Comput. Graph.*, vol. 32, 2012, pp. 81-83.
9. R. Dadashi, H. R. Kanan, "AVCD-FRA: A novel solution to automatic video cut detection using fuzzy-rule-based approach," *Comput. Vis. Image Understand.*, vol. 117, 2013, pp. 807-817.
10. M. Ramalingam, N. A. M. Isa, "A data-hiding technique using scene-change detection for video steganography," *Comput. Electr. Eng.*, vol. 54, 2016, pp. 423-434.
11. K. D. Seo, S. J. Park, S. H. Jung, "Wipe scene-change detector based on visual rhythm spectrum," *IEEE T. Consum. Electr.*, vol. 55, 2009, pp. 831-838.
12. C. W. Su, H. Y. M. Liao, H. R. Tyan, K. C. Fan, L. H. Chen, "A motion-tolerant dissolve detection algorithm," *IEEE T. Multimedia*, vol. 7, 2005, pp. 1106-1113.
13. M. S. Lee, Y. M. Yang, S. W. Lee, "Automatic video parsing using shot boundary detection and camera operation analysis," *Pattern Recogn.*, vol. 34, 2001, pp. 711-719.
14. C. C. Lo, S. J. Wang, "A histogram-based moment-preserving clustering algorithm for video segmentation," *Pattern Recogn. Lett.*, vol. 24, 2003, pp. 2209-2218.
15. H. J. Zhang, A. Kankanhalli, S. W. Smoliar, "Automatic partitioning of full-motion video," *Multimed. Syst.*, vol. 1, 1993, pp. 110-28.
16. T. Y. Liu, K. T. Lo, X. D. Zhang, J. Feng, "A new cut detection algorithm with constant false-alarm ratio for video segmentation," *J. Vis. Comm. Image Represent.*, vol. 15, 2004, pp. 132-144.
17. C. A. Dhawale, S. Jain, "Motion compensated video shot detection using multiple feature experts," *ICGST International Journal on Graphics, Vision, and Image Processing*, vol. 8, 2009, pp. 1-11.
18. Z. Lu, X. Jiang, A. Kot, "Color space construction by optimizing luminance and chrominance components for face recognition," *Pattern Recogn.*, vol. 83, 2018, pp. 456-468.
19. N. Alamgir, K. Nguyen, V. Chandran, 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, 2018, pp. 1-10.
20. D. Hiyama, T. Shimobaba, T. Kakue, T. Ito, "Acceleration of color computer-generated hologram from RGB-D images using color space conversion," *Optic. Comm.*, vol. 340, 2015, pp. 121-125.
21. K. Muhammad, M. Sajjad, I. Mehmood, S. Rho, S. W. Baik, "Image steganography using uncorrelated color space and its application for security of visual contents in online social networks," *Future Gener. Comp. Sy.*, vol. 86, 2018, pp. 951-960.
22. N. Alamgir, K. Nguyen, V. Chandran, 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, 2018, pp. 1-10.
23. H. Zhao, Q. Li, H. Feng, "Multi-focus color image fusion in the HSI space using the sum-modified-laplacian and a coarse edge map," *Image Vis. Comput.*, vol. 26, 2018, pp. 1285-1295.
24. M. N. Zakaria, N. A. A. Wahab, N. Maamor, B. Jalaei, A. A. A. Dzulkarnain, "Auditory brainstem response (ABR) findings in males and females with comparable head sizes at supra-threshold and threshold levels," *Neurology, Psychiatry and Brain Research*. Vol. 32, 2019, pp. 4-7.
25. Y. Yang, X. Wang, Z. Xu, "The multiplicative consistency threshold of intuitionistic fuzzy preference relation," *Inform. Sci.*, vol. 477, 2019, pp. 349-368.
26. D. Zhang, Z. Xia, "Weighted-averaging estimator for possible threshold in segmented linear regression model," *J. Stat. Plann. Infer.*, vol. 200, 2019, pp. 102-118.

27. P. Liu, J. M. Guo, K. Chamnongthai, H. Prasetyo, "Fusion of color histogram and LBP-based features for texture image retrieval and classification," *Inform. Sci.*, vol. 390, 2017, pp. 95-111.
28. J. Pradhan, S. Kumar, A. K. Pal, H. Banka, "A hierarchical CBIR framework using adaptive tetrolet transform and novel histograms from color and shape features," *Digit. Signal Process.*, vol. 82, 2018, pp. 258-281.
29. S. Borjigin, P. K. Sahoo, "Color image segmentation based on multi-level Tsallis-Havrda-Charvat entropy and 2D histogram using PSO algorithms," *Pattern Recogn.*, vol. 92, 2019, pp. 107-118.
30. C. Y. Wong, G. Jiang, M. A. Rahman, S. Liu, T. Wu, "Histogram equalization and optimal profile compression based approach for colour image enhancement," *J. Vis. Comm. Image Represent.*, vol. 38, 2016, pp. 802-813.
31. W. Lin, Y. Wang, Y. Zhuang, S. Zhang, "Evaluate the number of clusters in finite mixture models with the penalized histogram difference criterion," *J. Process Contr.*, vol. 23, 2013, pp. 1052-1062.
32. S. Dubuisson, "Tree-structured image difference for fast histogram and distance between histograms computation," *Pattern Recogn. Lett.*, vol. 32, 2011, pp. 411-422.
33. D. B. L. Bong, B. E. Khoo, "Blind image blur assessment by using valid reblur range and histogram shape difference," *Signal Process. Image Comm.*, vol. 29, 2014, pp. 699-710.
34. C. Zhu, C. Mei, R. Zhou, "Weight-based label-unknown multi-view data set generation approach," *Inform. Process. Lett.*, vol. 146, 2019, pp. 1-12.
35. G. Beliakov, D. Gomez, S. James, J. Montero, J. T. Rodriguez, "Approaches to learning strictly-stable weights for data with missing values," *Fuzzy Set. Syst.*, vol. 25, 2017, pp. 97-113.
36. R. Burduk, "Classifier fusion with interval-valued weights," *Pattern Recogn. Lett.*, vol. 34, 2013, pp. 1623-1629.

AUTHORS PROFILE



Seok-Woo Jang received the B.S., M.S., Ph.D. degrees in Computer Science from Soongsil University, Seoul, Korea, in 1995, 1997, and 2000, respectively. From October 2003 to January 2009, he was a Senior Researcher with the Construction Information Research Department at Korea Institute of Construction Technology (KICT), Ilsan, Korea. Since March 2009, he has been a Professor in the Department of Software, Anyang University, Korea. His primary research interests include robot vision, augmented reality, video indexing and retrieval, cluster computing, biometrics and pattern recognition.



Sung-Youn Cho received his B.S. and M.S. degrees in Electronic Engineering from Hanyang University, Seoul, Korea, in 1987 and 1989, respectively. He received his Ph.D. degree in Computer Engineering from University of Wales, England in 1998. He is currently a professor in the Department of Software at Anyang University, Korea. His research focuses on image processing, pattern recognition, artificial intelligence, and situational awareness system.