International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Smoke Detection using Local Binary Pattern

D.A Agrawal[Á] and Prachi Mishra[Á*]

[Á]Shri Sant Gadge Baba Collage of Engg. & Technology Bhusawal, India

Accepted 30 Nov 2014, Available online 01 Dec 2014, **Vol.4, No.6 (Dec 2014)**

Abstract

To realize quick and robust fire detection with image information of real scenes, smoke is a key feature information in detection methods. Since smoke does not keep stationary shape, it is difficult apply ordinal image processing techniques such as the edge or contour detection directly. Image information of smoke is also affected from its environmental conditions such as illumination changes and background objects. In this study, we adopt Local Binary Patterns (LBP), which is defined as a simple texture operator computed using the center pixel value and its neighborhood pixel values. From its definition, LBP is a robust image descriptor against the illumination change. The adaptive detection for real–scene situations is realized by AdaBoost. Results using with real scene data show that the presented method can provide accurate results against the various conditions of real world situation.

Keywords: *Smoke detection, Binary Pattern etc.*

1. Introduction

For fire alert system, the most important problem is how quick the existence of fire is detected. The fire detection in open areas, image information is mainly used due to the fact that other sensing devices such as a gas senser become high– cost and cannot set them in some places. Some methods of detecting a fire or flame directly (Hidenori Maruta *et al*, 2013; H. Maruta *et al*, 2011; H. Maruta *et al*, 2010) are proposed, however, they have some difficulties to catch fire with cameras depending on the relation of places between a camera position and a fire generating place.

For such purpose, smoke is a key feature since we can observe it even if fire is not in the field of view of the camera. There are some computer vision based methods for smoke detection which use the edge or color information based technique (B.U.Toreyin, Y. Dedeoglu13th European Signal Processing Conference, Antalya, Turkey, 2005.) They have difficulties in treating characteristic properties of smoke and needing high-cost computations to detect smoke whole of wide-view images or image sequences. There are other approaches which combine texture features and its time–series properties (H. Maruta, A. Nakamura and F. Kurokawa *et al* 2010 IEEE). It is shown that they can provide accurate results of smoke detection in open areas. However, they need relatively long time to obtain the result as they use the time accumulation technique, which becomes the problem in some real world situations. Additionally, these methods have a common problem, which is that the detection result is affected from an illumination change. This becomes a serious problem for open area image processing as whether conditions, for examples, clouds, lighting sources, wind, etc., are always changing dynamically.

LBP is defined as a simple texture operator computed using the center pixel value and its neighborhood pixel values. As it is considering the result as a binary number, LBP is robust against the illumination change. we prepare a training set, we need to consider that the image information of smoke is strongly affected from its background. That is, we have difficulties to gather smoke images and non–smoke images in a wide range of variations. We consider smoke detection as a two–class problem in the context of learning problem.

2. Smoke Detection Using LBP

A. Preprocessing

In the presented method, we detect moving objects in an im- age sequence as candidates of smoke regions in a preprocesing. As a growth–speed of smoke is considered, we obtain 1 frame–per–second rate image sequences $\{(t)|$ t =0 ,1,2,⋯} from original image sequences obtained from a single camera. We use an image subtraction technique to extract regions of moving objects, which are candidates of smoke. The subtracted image frame is written as $(t)=f(t)-f(t-1)$. This(t) is used in a following main processing.

B. Local Binary Patterns

Local binary patterns of LBP deals with the neighbor pixels in an image by using thresholding the neighborhood of each pixel and consider the result as a binary number,

*Corresponding author: **Prachi Mishra**

and thus, is used for segmentation and clustering in an image. The Algorithm is as follows

- The LBP feature vector, in its simplest form, is created in the following manner:
- Divide the examined window into cells (e.g. 16x16 pixels for each cell).
- For each pixel in a cell, compare the pixel to each of its 8 neighbors (on its left-top, left-middle, left-bottom, right-top, etc.). Follow the pixels along a circle, i.e. clockwise or

Where the center pixel's value is greater than the neighbor's value, write 1. Otherwise, write 0. This gives an 8-digit binary number (which is usually converted to decimal for convenience).

- Compute the histogram, over the cell, of the frequency of each number occurring (i.e., each combination of which pixels are smaller and which are greater than the center).

Optionally normalize the histogram

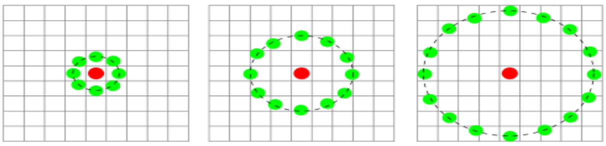

Fig.1 LBP based texture identification Problem

The LBP was proposed as a two-level version of the texture unit to describe the local textural patterns The original version of the local binary pattern operator works in a 3×3 pixel block of an image. The pixels in this block are thresholded by its center pixel value, multiplied by powers of two and then summed to obtain a label for the center pixel.

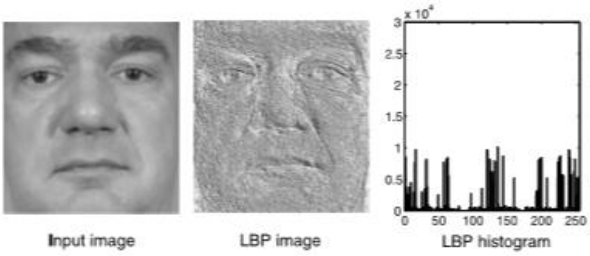

Fig.2 Example of an input image. The corresponding LBP image and Histogram

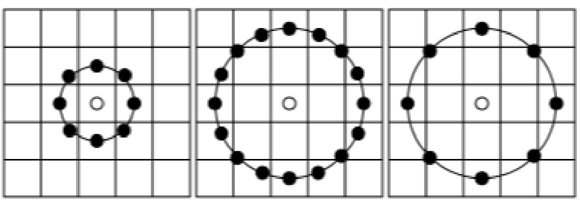

Fig.3 The Circular (8,1), (16,2) And (8,2) Neighborhoods. The Pixel Values Are Bilinearly Interpolated Whenever The Sampling Point is not in the Center of a Pixel

As the neighborhood consists of 8 pixels, a total of 28 =256 different labels can be obtained depending on the relative gray values of the center and the pixels in the neighborhood. The basic LBP using 8 pixels in a 3×3 pixel block, this generic formulation of the operator puts no limitations to the size of the neighborhood or to the number of sampling points.

After the preprocessing, we compute local binary patterns (LBP) from subtracted images (t). LBP is a simple tex- ture operator computed using the center pixel value and its neighborhood pixel value (T. Ojala, M. Pietika¨ainem and D. Harwood: et a1996.). Previous researches showed that LBP operator works well in both reducing computational complexity and reasonable feature extraction of images com- pared to conventional texture features (T. Ojala, M. Pietik¨ainen and T. M¨aenp¨a¨a: IEEE Trans. *et al* 2002),. Especially, it is suggested that LBP is effective to illumination.

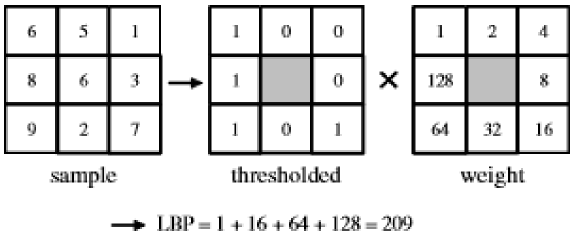

Fig.4 The Original LBP

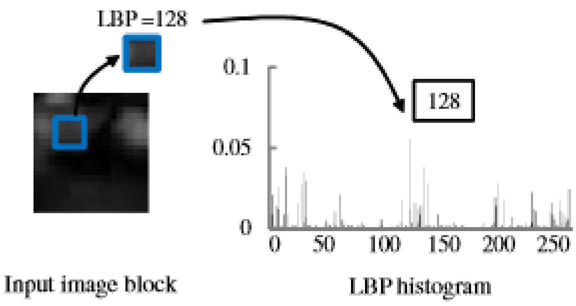

Fig. 5 Computation of LBP

The smoke detection in open areas, the lighting condition always changes. Thus, we consider that the technique using LBP is effective in smoke detection in open areas which has influence of environmental conditions such as weather, day or night time, and so forth. In this study, we adopt the original LBP value (T. Ojala, M. Pietika¨ainem and D. Harwood: *et al* 1996.). The original LBP is simply computed using the center pixel value and its neighborhood pixel value .The LBP operator works in a 3×3 neighborhood, using the center pixel value as a threshold. The neighborhood pixels value of the center pixel are multiplied by the binomial weights given to the corresponding pixels and obtained values are summed for LBP value (LBP = 1·20 + 0·21 + 0·22 + 0·23 + 1·24 + 0·25 + 1·26 + 1·27 = 209) of this texture unit. Using LBP values from an image block divided into 24 × 24 pixels, we create a LBP histogram. The above process to compute the LBP histogram is shown in Fig. 2. We compute the LBP histogram of each image block and treat it as a 256-dimensional feature vector of the image block. Therefore, the detection result is block–based one.

The local binary pattern operator is an image operator which transforms an image into an array or image of integer labels describing small-scale appearance of the image. These labels or their statistics, most commonly the histogram, are then used for further image analysis. The most widely used versions of the operator are designed for monochrome still images but it has been extended also for color (multi channel) images as well as videos and volumetric data. The actual LBP operator in spatial domain ,while deals with spatiotemporal LBP .

When we use the LBP histograms as a feature vector, the components of input vector consist of each histograms bin. We define the similarity of the LBP histogram which measure the distance from smoke–class and non–smoke class with Bhattacharyya coefficient (K. Fukunaga: *et al*, 1990.)

We prepare the averaged LBP histogram *HS* and *HN* from manually selected smoke and non–smoke image blocks. The Bhattacharyya coefficient represents the similarity of two histograms in 0 to 1. In this study, Bhattacharyya coefficients of input LBP histogram and averaged histograms of both

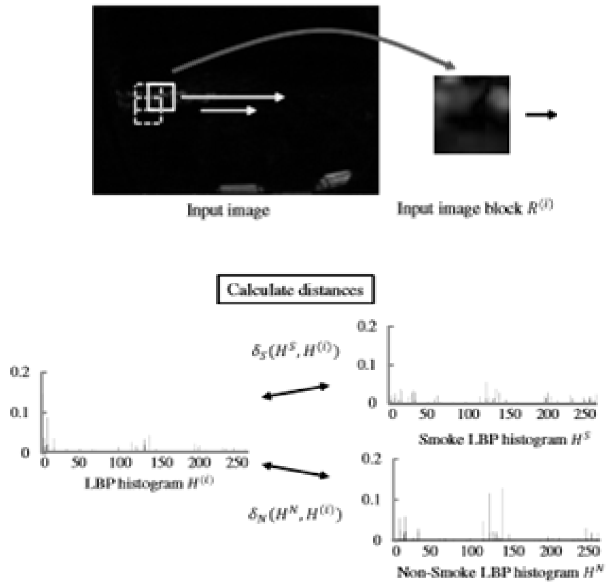

Fig.6 Calculation of Bhattacharyya Coefficients As The Distance From Smoke Class and Non–Smoke Class.

classes are written as Eq. (1) and (2)

$$\delta_s(H^s, H^{(i)}) = \sum_{k=1}^{256} \sqrt{HsH(i)} \quad (1)$$

$$\delta_N(H^N, H^{(i)}) = \sum_{k=1}^{256} \sqrt{HNH(i)} \quad (2)$$

where $H^S =(H^S{}_1 ,\cdots,H^S{}_{256})$ and $H^N =(H^N{}_1 ,\cdots,H^N{}_{256})$ are average histograms of smoke and non–smoke, $H^{(i)} = (H^{(i)}{}_1 ,\cdots,H^{(i)}{}_{256})$ is an i-th histogram of image block obtained from an input image. This calculation process is shown in Fig. 3. using Eq. (1) and (2).

We define weak classifiers on the basis of the difference of similarity of the Bhattacharyya coefficient $(H^S,{}^{(i)})$ and $\delta_N(H^N,H^{(i)})$ as shown in Eq. (3)

$$\xi_m(H^{(i)}) = \begin{cases} +1 \\ -1 \end{cases}$$
+1 (δs-δ_N >Θ_m)
-1 (otherwise) (3)

where θm is the threshold value adjusted to every iteration step m of AdaBoost so that an error rate serves minimum value when the sample data is identified. When the input LBP histogram (i) is similar to smoke, it is classified to smoke (+1), else it is classified to non–smoke (−1), and then we attach a label (smoke or non–smoke) to the image block. The weak classifier is defined by this labeling. That is, the weak classifier is defined by the similarity measure with the Bhattacharyya coefficient (Eq. (1) and (2))and the classification result is obtained from Eq. (3)

3. Smoke Detection Method Using LBP

- Image Pre-Processing for Motion Detection (Smoke) using direct Subtraction
- Local Binary Pattern based Map Generation using Histogram

Ada-Boost algorithm based Artificially Intelligent Learning implementation over thousands of images of the same place with smoke.

Proposed Algorithm

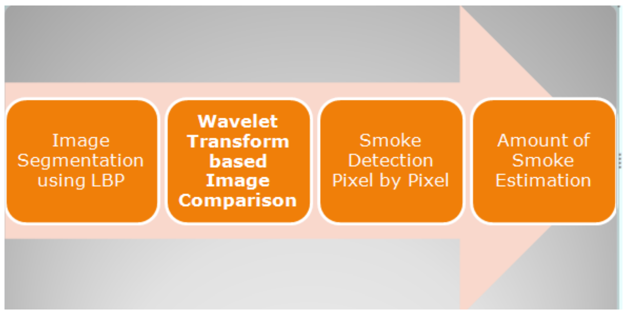

Fig.7 Flow of Algorithm

4. Image Segmentation using 2D Osto Method

Image segmentation is one of the basic techniques of image processing and computer vision. It is a key step for image analysis, comprehension and description. Among all the segmentation techniques, thresholding segmentation method is the most popular algorithm and is widely used in the image segmentation field. The basic idea of automatic thresholding is to automatically select an optimal or several optimal gray-level threshold values for separating objects of interest in an image from the background based on their gray-level distribution. Automatic thresholding techniques can be roughly categorized as global thresholding and local thresholding. Otsu thresholding technique is one of the global thresholding method and has been cited as an effective technique . One of them is its sensitivity to the object size Say in brief, if the object proportion is much less than background, the pixels in background will be wrongly classified as object; on the contrary, if the object proportion is much more than background, the pixels in object will be wrongly classified as background. As for our renal biopsy samples, the object size is much less than background, the wrong classification of pixels by traditional Otsu method will lead to the failure segmentation

5. Threshold Selection Method from Gray Level Histogram

A nonparametric and unsupervised method fautomatic threshold selection for picture segmentation is presented. An optimal threshold is selected by the discriminator criterion, namely, so as to maximize the separability of the resultant classes in gray levels. The procedure is very simple, utilizingonly the zeroth-and the first-order cumulative moments of the gray-level histogram. It is straightforward to extend the method to multithreshold problems. Several experimental results are also presented to support the validity of the method.

It is important in picture processing to select an adequatethres hold of gray level for extracting objects from their background. A variety of techniques have been proposed in this regard. In an ideal case, the histogram has a deep and sharp valley between two peaks representing objects and background, respectively, so that the threshold can be chosen at the bottom of this valley . However, for most real pictures, it is often difficult to detect the valley bottom precisely ,especially in such cases as when the valley is flat and broad, imbued with noise, or when the two peaks are extremely unequal in height, often producing no traceable valley. There have been some techniques proposed in order to overcome these difficulties. They are, for example, the valley sharpening technique, which restricts the histogram to the pixels with large absolute values of derivative (Laplacian or gradient), and the difference histogram methode ,which selects the threshold at the gray level with hexadecimal amount of difference. These utilize information concerning neighboring pixels (or edges) in the original picture to modify the histogram so as to make it useful for thresholding. Another class of methods deals directly with the gray-level histogram by parametric techniques. For example, the histogram is approximated in the least square sense by a sum of Gaussian distributions, and statistical decision procedures are applied . However, such a method requires considerably tedious and sometimes unstable calculations. Moreover, in many cases, the Gaussian distribution sturn outtobeameager approximation of the real modes. In any event, no goodness of threshold has been evaluated in most of the methods so far proposed. This would imply that it could be the right way of deriving an optimal thresholding method to establish an appropriate criterion forevaluatingthe goodness of threshold from a more general standpoint. It is not only important as a standard technique in picture processing, but also essential for unsupervised decision problems in pattern recognition. A new method is proposed from the viewpoint of discriminant analysis; it directly approaches the feasibility of evaluating the goodness of threshold and automatically selecting an optimal threshold.

6. Avantages

Most of the algorithms do not take care of changing values of brightness due to change in weather conditions, and thus are not so practical for open area's, like forests, and hills etc. (Forest fires being a very big issue in United States).

No algorithm has tried to estimate the amount of smoke getting generated, so as to make an early warning system. If multiple camera's are synchronized together, then the camera's away from the source will register more smoke, but are not useful for ringing an alarm, as they do not have any information about the source of the fire

Conclusion

Smoke detection method based on local binary patterns. In smoke detection, LBP is not only a simple operator but also effective to illumination variations, which can contribute to obtain accurate results. We also combine with AdaBoost, which is one of the machine learning techniques, to improve the accuracy of detection results. As the preprocessing, we extracted the moving objects as candidate smoke regions in images. From the subtracted images, we computed LBP values and LBP histograms from the image blocks. Using LBP histograms as the input vector of AdaBoost, we detect whether the image blocks are smoke or non–smoke.

Acknowledgment

I cannot complete my work without support of my internal guide.

References

Hidenori Maruta, Yusuke Iida, Fujio Kurokawa (2013), Smoke Detection Method Using Local Binary Patterns and AdaBoost, IEEE paper.

H. Maruta, Y. Kato and F. Kurokawa (2011), Smoke detection using time series property of texture features, IEEE Trans. Electrical and Electronic Engineering Volume 6, Issue 4, pp. 390–39

H. Maruta, A. Nakamura and F. Kurokawa (2010), A new approach for smoke detection with texture analysis and support vector machine, Proceedings of 2010 IEEE International Symposium on Industrial Electronics, pp. 1550 – 1555.

B. U. T¨oreyin, Y. Dedeoglu and A. E. Enis (2005) Wavelet based real–time smoke detection in video, 13th European Signal Processing Conference, Antalya, Turkey.

G. Zhao and M. Pietika ¨inen (2007.), Dynamic Texture Recognition Using Volume Local Binary Patterns, Proc. Workshop Dynamical Vision, pp. 165-177

H. Maruta, A. Nakamura and F. Kurokawa (2010)A new approach for smoke detection with texture analysis and support vector machine, Proceedings of 2010 IEEE International Symposium on Industrial Electronics, pp. 1550 - 1555.

K. Fujita and S.K. Nayar (2003), Recognition of Dynamic Textures Using Impulse Responses of State Variables, Proc. Third Int'l Workshop Texture Analysis and Synthesis. pp. 31-36

T. Ojala, M. Pietika¨ainem and D. Harwood (1996) A comparative study of texture measures with classification based on feature distributions, Pattern Recognition, Vol. 29, pp. 51-59.

T. Ojala, M. Pietik¨ainen and T. M¨aenp¨a¨a (2002) Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 24, pp. 971–987

K. Fukunaga (1990*) Introduction to statistical pattern recognition*, Morgan Kaufmann, San Francisco.

G. Doretto, A. Chiuso, S. Soatto, and Y.N. Wu (2003), Dynamic Textures, Int'l J. Computer Vision, vol. 51, no. 2, pp. 91-109.

H. Schneiderman and T. Kanade (2004) Object Detection Using the Statistics of Parts, International Journal of Computer Vision, Vol. 56, pp. 151–177

M. Pietika¨ainen, G. Zhao, A. Hadid and T. Ahonen (2011), *Computer vision using local binary patterns*, Springer, London

B. U. T¨oreyin, Y. Dedeoglu and A. E. Enis (2005)Wavelet based real–time smoke detection in video, 13th European Signal Processing Conference, Antalya, Turkey.

Rahul Sasidharan, Menaka D (April 3-5, 2013) Dynamic Texture Segmentation of Video Using Texture Descriptors and Optical Flow of pixels for Automating Monitoring in Different Environments, International conference on Communication and Signal Processing.

G. Zhao and M. Pietika ¨inen (2007.), Dynamic Texture Recognition Using Volume Local Binary Patterns, Proc. Workshop Dynamical Vision, pp. 165-177