Mosaic Region Detection in Pornographic Images

Jung-Jae Yu*, Seung-Wan Han**
*,**ETRI, Korea
E-mail : jungjae @etri.re.kr*, hansw@etri.re.kr**

1. Introduction

This work is aimed at the automatic detection of mosaic regions in pornographic images to identify and block objectionable contents for children and teenagers. We propose a novel method to detect the mosaic region using a new cross-shaped feature extraction and a new cost function related to the regularity of the locations of extracted features. Proposed method is fast and our experimental results show that it can be used for filtering out the pornographic contents including mosaic regions. Previous trials to detect mosaic regions in [1] and [2] are not suitable for the purpose to identify pornographic contents including mosaic regions because they are prone to the noise effect and image degradation caused by video compression process.

2. Proposed Method

Usually, mosaic regions in pornographic images are composed of repetitive squares filled with average color in each squares. But the edges between the squares are frequently blurred because of the effect of compression and the interpolation for the change of the image size. Traditional edge operations such as simple sobel edge and canny edge operations are not suitable for the detection of the mosaic regions. So we propose a new method to detect cross-shaped corner features in mosaic regions. First step is to generate edge images using sobel operation. The edge image is the sum of absolute values of horizontal and vertical sobel images. The second step is to detect cross-shaped features using the below macro blocks. Figure 1 shows the notation of a macro block.

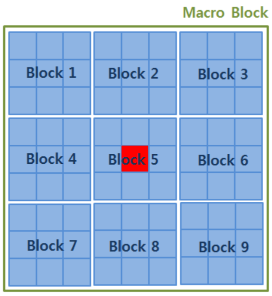

Fig. 1 A macro block to distinguish a cross-shaped feature at the central pixel.

A macro block is composed of 9 small blocks and each small block contains 9 pixels as in the Figure 1. To distinguish a cross-shaped feature at the center pixel (a red square in the Figure 1), all pixels in one macro block are used. $Edge_{up}$, $Edge_{down}$, $Edge_{left}$ and $Edge_{right}$ are calculated by Equation 1. If all of these 4 edge components are bigger than a predefined threshold value, $th_{edge1}$and the $Edge_{mean}$ in Equation 1 is bigger than another threshold value, th_{edge2} then the central pixel is regarded as a cross-shaped feature. These processes are repeated at all pixels in the input image.

$$Edge_{up} = \frac{Edge_{B2}}{(Edge_{B1} + Edge_{B3})}, \quad Edge_{down} = \frac{Edge_{B8}}{(Edge_{B7} + Edge_{B9})}$$

$$Edge_{left} = \frac{Edge_{B4}}{(Edge_{B1} + Edge_{B7})}, \quad Edge_{right} = \frac{Edge_{B6}}{(Edge_{B3} + Edge_{B9})}$$

$$Edge_{mean} = \frac{Edge_{up} + Edge_{down} + Edge_{left} + Edge_{right}}{4} \quad \ldots(1)$$

The third step is to extract and verify mosaic candidate regions using the previously extracted cross-shaped features. The features at which many other features centralized and the density of the feature distribution is bigger than a threshold value, $th_{density}$ are regarded as mosaic candidate regions. To verify the mosaic candidate regions,a cost function related to the regularity of the locations of extracted features are used. If a mosaic candidate region is really a mosaic region, cross-shaped features included in the region should be located at regular grid points. Equation 2 shows the method to calculate the cost function. In Equation 2, x_{center} and y_{center} are estimated by selecting the coordinates in x, y axis where the number of counting is highest with the features included in the candidate region.

$$
\begin{aligned}
& Cost = \sum_{i=1}^{N_f} \left\| \mathbf{x}_i - \tilde{\mathbf{x}}_i \right\| \\
& \tilde{x}_i = x_i + k_{x,i} * x_{step} \\
& \tilde{y}_i = y_i + k_{y,i} * y_{step} \\
& x_{step} = \arg\min_{x_{step}} \sum_{i=1}^{N_f} \left| x_i - \left(x_{center} + x_{step} * k_{x,i} \right) \right| \\
& y_{step} = \arg\min_{y_{step}} \sum_{i=1}^{N_f} \left| y_i - \left(y_{center} + y_{step} * k_{y,i} \right) \right| \\
& k_{x,i} = \arg\min_{k_{x,i}} \left| x_i - \left(x_{center} + x_{step} * k_{x,i} \right) \right| \\
& k_{y,i} = \arg\min_{k_{y,i}} \left| y_i - \left(y_{center} + y_{step} * k_{y,i} \right) \right|
\end{aligned}
\quad \ldots (2)
$$

3. Experimental Results

We performed an experiment to detect mosaic regions using the proposed method. 500 mosaic images captured from 40 pornographic videos and 47,251 non-mosaic images capture from non-objectionable videos were used. Non mosaic images were collected from TV contents, movies, CFs and animations etc. Figure 2 shows the test results of some mosaic .

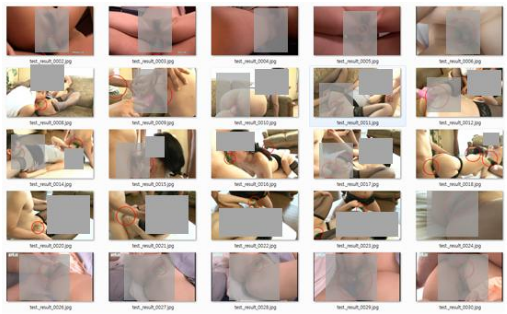

Fig. 2 Test results for objectionable mosaic images

The true-positive rate (Recall) was 92.4% and false-positive rates was 7.65%. The average computation time was 0.04 sec/frame in our quadro 2.7 GHz CPU.

4. Aknowledgment

This research was funded by the MSIP(Ministry of Science, ICT & Future Planning), Korea in the ICT R&D Program 2014

5. Referencese

[1] Z. Wei, J. Lin, L. Zhang and S. Song "Mosaic Defect Detection Based on Macro Block Solid Edge Detection", Research Journal of Applied Sciences, Engineering and Technology, vol. 5, April 15, 2013, pp.3549-3553.

[2] S. Shui-Fa1, H. Shu-heng, W. Gang, X. Yi-chun, L. Bang-Jun "Mosaic Defect Detection in Digital Video", Chinese Conference on Pattern Recognition, October, 2010.