

ERASING VIDEO LOGOS BASED ON IMAGE INPAINTING

Wei-Qi Yan, Mohan S. Kankanhalli

Department of Computer Science
School of Computing, National University of Singapore, Singapore 117543
{yanwq, mohan}@comp.nus.edu.sg

ABSTRACT

A video logo is usually a declaration of the video copyright. However it sometimes causes visual discomfort due to the presence of multiple logos in videos that have been filed and exchanged by different channels. In this paper, we present an approach to erase logos from video clips. Based on the histogram energy analysis of the relevant video frames, we obtain the best quality logo frame that can be easily processed in the selected region of video frames. After that, we mark the logo area in the entire sequence of frames and inpaint each frame of the video logo based on color interpolation. We describe our technique in this paper and also provide experimental results.

1. MOTIVATION

A video logo is a trademark or a symbol of Intellectual Property Protection (IPP) and is popularly used by most television channels. It helps in identifying the owner of the video clips. Figure 1(a) gives some examples of video logos. Generally speaking, a video logo possesses the following characteristics:

- It has a special shape that makes it unique.
- It needs to be visible, so it is distinguishable from the background color.
- Its position is static with time.
- Its area is not large and does not significantly occlude the video content.

However, a video logo sometimes leads to annoying visual artifacts especially in cases where multiple logos exist in filed videos that are reused --- like in news videos



(a) Video Logo Examples



(b) Video Logos Superimposed Together

Figure 1: Some Video Logos

obtained from some other channel or in cases where some television stations superimpose their video logos on other. In Figure 1(b), the logos are overlapped directly, stacked on another mosaiced logo, and blended with the blurred one. This is visually not very appealing to the viewers. These artifacts provide the motivation for our research and our aim is to erase video logos from video frames. This is also very useful as an attack on visible watermarks in digital videos.

2. RELATED WORK

The term '*inpainting*' refers to the technique of filling in of any gaps in photographs [1]. The goal of inpainting is to restore the missed or damaged portions of the photos. In [1], the authors introduce an algorithm for digital still image inpainting that attempts to replicate the basic techniques used by professional restorators. The technique allows the users to fill-in numerous regions containing different structures and surrounding backgrounds.

In [2], the authors use the ideas from classical fluid dynamics to propagate isophote lines continuously from the exterior into the region to be inpainted. The method is based on the Navier-Stokes equations for fluid dynamics. It is a new approach that introduces ideas from computational fluid dynamics into problems in computer vision and image analysis.

In both [1] and [2], partial differential equations (PDE) are used to calculate the propagation of isophote lines iteratively. The calculation is compute-intensive when applied to inpainting of video logos. However, the difference between [1], [2] and our approach is that we simplify their approach by utilizing only the information of surrounding pixels in the marked area. This cuts down the computation time without affecting quality. We have also applied it to attack the visible watermarking of digital images [3].

Given that in a video the logo appears in several frames, we are able to select the best frame to start the erasure process in the sequence (i.e. the frame where the logo is the clearest). In order to find the best quality logo in video frames, we adapt ideas from [4] which describe an algorithm for caption recognition using contrast energy techniques.

In fact, erasing video logos can be considered to be a part of video restoration [5]. In [5], the author provides some classical methods that restore motion pictures by texture substitution based on motion vectors. This cannot be applied to logos because the precise logo region cannot be found in the preceding or succeeding frames all the time. If the logo area is overlapped with a similar region, the edge of logo region will still be obvious. Due to this reason, we shall restore video frames by the image inpainting technique.

3. OUR CONTRIBUTION

For our technique, first we manually select a rough rectangular region in a video clip that encloses the video logo. Then we transform the color frames of the selected region into grey-scale ones. In addition, we enhance the contrast in the grey-scale video and get the best quality logo frame in the video clip.

The best quality logo is determined by the maximum histogram energy of the logo area of video frames. The ideal logo frame will have low color variance and less texture details in the background which will display the logo clearly. Histogram energy can capture the requirements. In Figure 2 (from <http://www.sina.com.cn>), two frames with different quality logos are shown. For inpainting, we would like to start with the one in Figure 2(b) and not the one in Figure 2(a).



(a) A Bad Quality Video Logo (b) A Better Quality Video Logo

Figure 2: Video Logos with Different Quality

After the logo is obtained, the region with logo is restored by the revised image inpainting technique introduced later. Thus, our technique is summarized as the following steps:

- (1) Select a logo region.
- (2) Analyze the histogram for this region for all the frames to select the frame with the best logo.
- (3) Enhance & binarize this frame to delineate the exact logo region.
- (4) Inpaint this region for all frames in the sequence.

This approach is different from the existing approaches. We utilize the fact that a video clip has many frames and has more data than a single image. So it is possible to find the best logo frame under certain conditions. Thus the video logo area can be inpainted without the iterative algorithm in [1][2]. We need to perform the interpolation of colors of surrounding pixels only.

3.1. Histogram Energy

In order to get the best grey-scale image, an approach based on histogram energy analysis is adopted. Suppose the grey-scale image has the resolution $R_x \times R_y$, the histogram of the image is $H = (h_0, h_1, \dots, h_{255})$, $\sum_{i=0}^{255} h_i = R_x \times R_y$.

Definition: Histogram energy of a grey-scale image $R_x \times R_y$ is defined as:

$$E_H = \sum_{i=0}^{255} h_i^2 \quad (1)$$

The following observation holds for histogram energy of two images with the same resolution:

Observation: If two images have the same resolution $R_x \times R_y$, the grey-scale image has less histogram energy than that of the binary one. This is easy to see because of triangle inequality.

This however tells us that the image with a single color has the maximum energy. So we should avoid such frames. Furthermore lesser the number of colors, higher is the histogram energy. Therefore, for finding the best logo frame, we perform the following steps:

- Step1.** Compute the grey-scale histogram of the selected region for all frames.
- Step2.** Normalize the histogram.
- Step3.** Calculate the histogram energy of every frame.
- Step4.** Choose the frame with the maximum energy.

3.2. Video Logo Inpainting

After transforming a color video into a grey-scale one, we binarize it using an adaptive threshold. This detects the logo region in the chosen best frame. This region is then marked for all the frames in the sequence. The techniques used for this are the standard ones found in most books on digital image processing.

Now we erase the logos from the video frames based on image inpainting. This problem in [1] and [2] is described as a discrete approximation of the PDE (Transport equation):

$$I_t = \nabla^\perp I \cdot \nabla \Delta I \quad (2)$$

where ∇^\perp denotes the perpendicular gradient $(-\partial_y, \partial_x)$, Δ denotes the Laplacian operator $\partial_{xx} + \partial_{yy}$, I is the image intensity, $I_t = DI / Dt$, $D / Dt = \partial / \partial t + v \cdot \nabla$ (material derivative) and $v = \nabla^\perp \Delta I$ (velocity field). Equation (2) projects the gradient of the smoothness of the image intensity in the direction of the isophote [2]. In order to solve the PDE, compute-intensive algorithms are provided in [1] and [2] based on fluid dynamics. However, in this paper, we simplify this computation and consider a revised algorithm.

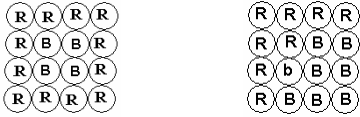
Figure 3 is helpful in understanding the inpainting algorithm. In the Figure 3(a), every circle is marked with an alphabetic character indicating a pixel in the video frame. The region with 'B' indicates that the area needs to

be inpainted. The region with ‘B’ and ‘b’ in Figure 3(b) has to be filled with the information from the surrounding pixels with ‘R’. The region with the ‘R’ pixels represents the undamaged area of an image. When we decide to fill a ‘B’ pixel, we average the colors of the surrounding ‘R’ pixels. For example in Figure 3(b), the current pixel B (marked by ‘b’ in the figure) is surrounded by ‘R’ pixels in 4 out of 8 directions. We fill the ‘b’ pixel with the average color of all these surrounding ‘R’ pixels. Our algorithm can be described as the following pseudo-code:

```

Subroutine LogoInpaint(){
  For RingNo = OutermostRing to InnermostRing
    If((PixelColor == B) && (PixelInCurrentRing)){
      Count number of surrounding pixels;
      Sum up color value of surrounding R pixels;
      Compute average color of R pixels;
      Set current pixel Color to average color;
    }
  For RestPixelHor = Left to Right
  For RestPixelVer = Top to Bottom
    If(CurrentPixelColor == B){
      Count number of surrounding R pixels;
      Sum up color value of surrounding R pixels;
      Compute average color of R pixels;
      Set current pixel color to average color;
    }
  }
}

```



(a) Whole of Inpainted Area (b) Part of Inpainted Area

Figure 3: Inpainted Region Configurations

The above algorithm works as follows. Suppose only one pixel needs to be filled, the average color of the surrounding pixels is the new filled color. If there are many concentric rings, all pixels are filled by the average color starting from the outermost ring to the innermost ring. The remaining unfilled pixels, if any, are found by raster-scanning and are filled by the average color of the neighboring unmarked R pixels. After this step, the region is completely filled.

The above revised algorithm is slightly different from the algorithm described in equation (2). In [1] and [2], the gradient in the direction of the isophote is considered and a complex and heavy computation is needed. [1] achieves its goal with an iterative algorithm while [2] does this by finding the discrete solutions of a designed Navier-Stokes equation. In this paper, we simplify the complex computation and fill the marked area of the logo by the average value of surrounding pixels according to ring filling order and raster-scan order. In other words, we substitute the gradient information of image edge in continuity with discrete values of neighboring pixels around the area. The advantage of our revised algorithm is simplicity while the disadvantage is that it perhaps cannot handle logos with a large area. Since the logo appearing in videos generally does not change its position and does not

occupy a large area, consequently our method is applicable.

The averaging process produces a pleasing visual effect and its error is bounded. Assume that the color of a grey-scale pixel to be filled is $I(i, j)$, then

$$I(i, j) = 1/N \cdot \sum_{k=1}^N I_k(i, j) \quad (3)$$

where $I_k(i, j)$, $i = 0, 1, \dots, R_x - 1$, $j = 0, 1, \dots, R_y - 1$ are the surrounding unmarked pixels of $I(i, j)$. Thus the error of the filled pixel with respect to the surrounding pixels is:

$$\varepsilon = \sum_{k=1}^N |I(i, j) - I_k(i, j)| \quad (4)$$

$N(N \leq 8)$ is the total number of these surrounding pixels.

$$\varepsilon \leq NI(i, j) + \sum_{k=1}^N I_k(i, j) \leq 2 \sum_{k=1}^N I_k(i, j) \quad (5)$$

(4) and (5) tell us that if the surrounding pixels have a low variance, the error is small and bounded, so that the logo area can be inpainted reasonably accurately.

4. IMPLEMENTATION

This section will provide our experimental results and the analysis to them. Figure 4, 5 and 6 are the experimental results based on our method. Figure 4 (a) is an image damaged by lines, Figure 4(b) is the restored image. Figure 5(a) is an image marked by a text logo, Figure 5(b) is the corresponding restored image. Figure 6 is a restored frame marked by graphic logo, Figure 6(a) is a frame with marked video logo, Figure 6(b) is the restored video frame.



(a) A Damaged Image (b) An Inpainted Image

Figure 4: Inpainting Damaged Image



(a) A Marked Image (b) An Inpainted Image

Figure 5: Erasing Text Logo



(a) A Marked Image (b) An Inpainted Image

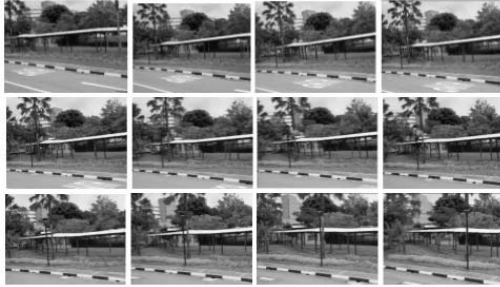
Figure 6: Erasing Graphic Logo

In order to perform a quantitative evaluation, we take 5 logos and a sample video. We stack the 5 logos on the video and obtain 5 videos with logos. Subsequently we

erase the logo in each video based on the method proposed in this paper and obtain 5 restored videos. In addition, we sum up the RMS (Root Mean Square) error values between the original video and 5 videos with logos, and the RMS values between the original video and 5 restored videos. Figure 7 shows our test data.



(a) The 5 Logos



(b) Some Frames of Original Video (Campus of NUS)

Figure 7: The Test Data

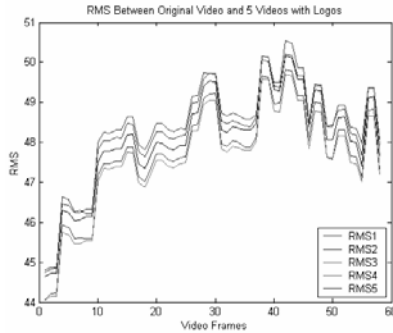


Figure 8: RMS between Original Video and 5 Videos Overlapped with Logos

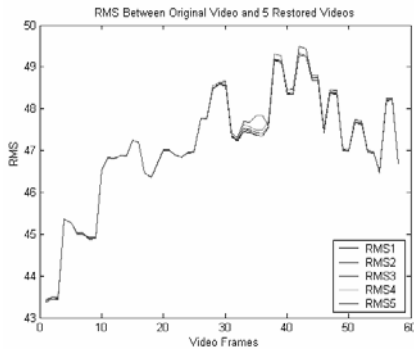


Figure 9: RMS between Original Video and 5 Restored Videos without Logos

Figure 8 illustrates the RMS between the original video and 5 videos overlapped with different logos. Since the logos are different from each other, the variant amplitude of the 5 RMS curves in the graph is obvious.

Figure 9 illustrates the RMS between the original video and 5 videos with erased logos. Because our algorithm is quite effective, the differences between inpainted frames and correspondence frames of original video are minor. From Figure 9, we can see that the 5

curves are almost coincident. Moreover, we take the 5 logos and apply them on 5 frames and restore these 25 frames. The RMS values are shown in Figure 10.

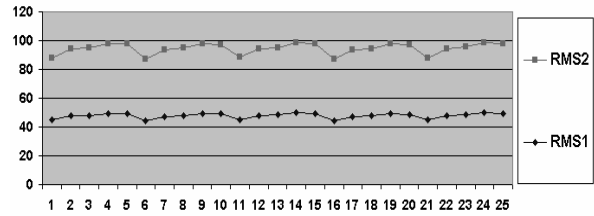


Figure 10: RMS between Original Frames and Frames Overlapped with Logos (RMS2), RMS between Original Frames and Restored Frames without Logos (RMS1).

From Figure 10, we see that the frames without logos have less difference of RMS than the frames with logos. This indicates that our restored frames are quite good approximations of the original frames.

5. FUTURE WORK

This paper introduces the method of erasing logos from video clips. The maximum histogram energy algorithm is helpful in selecting the best logo frame and has been found to be quite effective. The algorithm of image inpainting is simple yet can satisfy our requirement. Based on this work, we conclude that erasing logos in video clips is indeed possible. Animated logos in video can be handled by parsing the video into single frames and erasing them frame-by-frame. Our future work will focus on automatically finding and removing the logos in videos.

Acknowledgement We would like to thank Chitra Madhwacharyula for her helping with the preparation of this paper and the anonymous reviewers for their constructive comments.

6. REFERENCES

- [1] Marcelo Bertalmio, Guillermo Sapiro, Vicent Caselles, and Coloma Ballester, "Image Inpainting", *Computer Graphics, Proceedings of Siggraph' 2000*, pp.417-424, July 2000.
- [2] Marcelo Bertalmio, A. L. Bertozzi, and Guillermo Sapiro, "Navier-Stokes, Fluid Dynamics, and Image and Video Inpainting", *Proceedings of the International Conference on Computer Vision and Pattern Recognition, IEEE, Kauai, HI, volume I*, pp. 355-362, December 2001.
- [3] Yan Weiqi, "Research on Mathematical Methods and Applications in Digital Image Information Hiding", Ph.D. Thesis, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, June 2001.
- [4] Yi Zhang and Tat-Seng Chua, "Detection of Text Captions in Compressed Domain Video", *Proceedings of ACM Multimedia'2000 Workshops (Multimedia Information Retrieval)*, California, pp.201-204, November 2000.
- [5] Anil Kokaram, *Motion Picture Restoration*, Springer-Verlag London Limited, London, United Kingdom, 1998.