AUTOMATIC TV LOGO REMOVAL USING STATISTICAL BASED LOGO DETECTION AND FREQUENCY SELECTIVE INPAINTING

Katrin Meisinger, Tobias Troeger, Marcus Zeller, and André Kaup

Chair of Multimedia Communications and Signal Processing,
University of Erlangen-Nuremberg, Cauerstr. 7, 91058 Erlangen, Germany
{meisinger,troeger,zeller,kaup}@LNT.de

ABSTRACT

This paper outlines a method for automatically removing logos characterizing a broadcast station in TV sequences. First, the logo is detected automatically based on change detection of moving videos assuming that the image content is changing over time except for the location of the logo. In order to obtain initial logo masks, difference images between frames are binarized by thresholding. The final logo mask is obtained by subsequently refining the change masks by contour relaxation based on Markov Random Fields. Then, the image signal surrounding the logo is extrapolated using a frequency selective method and placed instead of the logo. The proposed algorithm is developed for TV sequences sampled from analog television, dealing thus with real world problems as noise, sampling and real logos.

1. INTRODUCTION

Broadcast stations may reuse TV sequences like sports sequences or news originally produced by another station. They seek to remove the TV logo and to replace it by their own logo. Usually the other logo is either blurred and the own one placed in a different corner or on top or multiple logos are overlaid. We derive an algorithm for automatic logo removal requiring two steps, first the logo is detected and then compensated.

Logos exhibit certain features we want to exploit for logo detection. Usually, they are placed statically in a corner of the sequence. They have a special shape and appear brighter than the background in order to stand out. They occur as opaque or semi-transparent logos, single-colored or colored. We have disregarded animated logos since they hardly occur, at least in German television.

In [1] a simple approach for logo removal is presented. First, the logo region is chosen manually. Then the frame with the smoothest background is selected where the logo detection takes place. Thresholding on the luminance component of the selected frame provides the logo mask. Semi-transparent and colored logos thus cannot be detected. Further, many logos are surrounded by a darker shadow in order to stand out on a bright background. The logo region is filled in by interpolation from adjacent neighboring pixels. The filling in causes blurring in case of larger logos and details cannot be restored.

Logo detection is also important for detecting TV commercials [2] assuming that commercials occur in the absensce of a logo. The detection is based on finding an area with stable contours.

We remove the logos automatically from TV spots by digital post processing. The logo detection is based on change detection in moving videos [3] assuming that the video content changes over time except for the logo region. Therefore, we can also detect semi-transparent and colored logos. We use the surrounding spatial information for logo inpainting because exploiting temporal information is not possible. In case of no motion the logo is present in the previous frame at the same position and for moving contents occlusions and uncoverings occur. The algorithm for logo inpainting uses the frequency selective extrapolation technique of [4], the principle of which is already used to conceal the effects of data losses caused by transmission errors for block coded data [5]. The algorithm is able to extrapolate edges, smooth and detailed areas.

2. AUTOMATIC LOGO DETECTION BASED ON A STATISTICAL MODEL

In [3] the robust detection of changes in moving videos in presence of noise based on a statistical model is described from which our algorithm for logo detection is developed.

Assuming that the video content changes over time except for the logo, actual differences between subsequent frames at the logo position occur due to noise. In order to detect changes, the grey level difference image D is generated between two frames of the sequence F_a and F_b obtaining the local difference $d[m,n,s] = f_a[m,n] - f_b[m,n]$, with m denoting the row index, n the column index and s the number of the difference image. Based on the assumption that the content is unchanged 'u', the actual difference $d[m,n,s]$ obeys a zero mean Gaussian distribution

$$p(d[m,n,s]|u) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{d[m,n,s]^2}{2\sigma^2}\right) \tag{1}$$

assuming Gaussian distributed camera noise with variance $\frac{\sigma^2}{2}$.

Thresholding on the difference image provides a change mask Q, the labels $q[m,n]$ of which are labeled by 'u' for unchanged and 'c' for changed.

To obtain more reliable decisions, the difference value is calculated within a measurement window. Since we seek to detect the logo assuming that no changes occur over time for the logo position, we apply the windowing in temporal direction and average differences for the considered pixel position in S subsequent difference images. Thus, the decision is made on

$$\Delta_1[m,n] = \sum_{s=1}^{S} \left(\frac{d[m,n,s]}{\sigma}\right)^2 \tag{2}$$

and labeled by 'u' if $\Delta_1[m,n]$ exceeds the threshold T, otherwise by 'c'. Alternatively, the local sum of absolute differences can be used as a test statistic

$$\Delta_2[m,n] = \sum_{s=1}^{S} \frac{\sqrt{2}}{\sigma} |d[m,n,s]| \tag{3}$$

where $d[m,n,s]$ obeys a Laplacian distribution reflecting the difference process as the description of the prediction error. For detailed derivations the reader is referred to [3].

However, decision errors occur. In order to refine the change masks, maximum a posteriori (MAP) detection is applied, aiming at eliminating small spots and smoothing the logo contours. The advantage of MAP based refinement is that both surrounding decisions and the actual difference value are taken into account for a decision. If there are more unchanged than changed labels evaluating the labels along the time axis, the decision threshold T should be altered in favour of unchanged, otherwise changed.

The objective is to find that change mask which is maximizing the a posteriori density $p(Q|D)$. Due to

$$p(Q|D) \sim p(D|Q)P(Q) \tag{4}$$

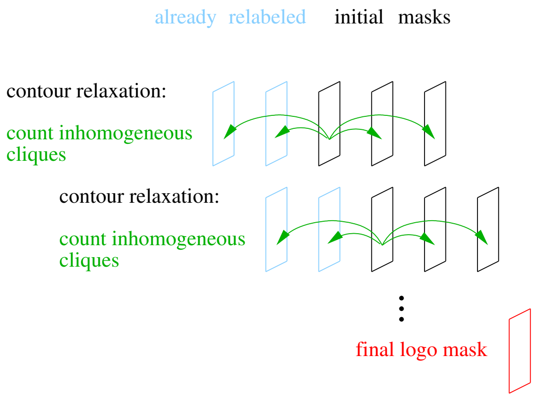

Figure 1: Refining the change masks: MAP based logo detection.

according to Bayes we can equivalently maximize the product comprised of the likelihood $p(D|Q)$ and the a priori density $P(Q)$. Modelling the differences as conditionally independent, we obtain

$$p(D|Q) = \prod_{m,n} p(d[m,n,s]|q[m,n]) \tag{5}$$

with $q[m,n] = u$ and $p(d[m,n,s]|q[m,n])$ according to (1). In the opposite case, when the label is changed $q[m,n] = c$, the difference arises from different random processes. However, for the sake of simplicity we model the process by a Gaussian process

$$p(d[m,n,s]|q[m,n] = c) = \frac{1}{\sqrt{2\pi\sigma_c^2}} \exp\left(-\frac{d[m,n,s]^2}{2\sigma_c^2}\right) \tag{6}$$

with variance $\frac{\sigma_c^2}{2}$. Obviously, the variance σ_c^2 is much larger than the variance σ^2 describing the camera noise.

The derivation of the new threshold is based on pixels pairs with different labels along the time axis, termed inhomogeneous cliques. The change mask Q is modelled as a second order Gibbs/Markov random field exhibiting the a priori density

$$p(Q) = \frac{1}{Z} \exp(-E_Q) \text{ with } E_Q = nB. \tag{7}$$

E_Q denotes the energy associated with that mask and n the number of inhomogeneous border pixels pairs in temporal direction, the cliques, with the associated potential B. The partition function Z is a normalizing constant and can be neglected therefore for the optimization. The a priori density $p(Q)$ is chosen reasonable because smoothly shaped change masks should be more likely to occur in temporal direction, i.e. if the previous and subsequent labels at position $[m,n]$ are unchanged the probability for unchanged should increase. If the surrounding labels are changed it should be more likely that the actual label is also changed.

In order to relax the contours of the change masks, the label is set to 'u' if $p(Q_u|D) > p(Q_c|D)$, otherwise 'c'. Inserting (1),(6),(7) and with help of (5) provides thus

$$p(d[m,n,s]|u) \exp(-E_{Q_u}) \underset{c}{\overset{u}{\gtrless}} p(d[m,n,s]|c) \exp(-E_{Q_c}). \tag{8}$$

We obtain the final decision rule solving for $d[m,n,s]^2$

$$d[m,n,s]^2 \underset{u}{\overset{c}{\gtrless}} 2\frac{\sigma_c^2\sigma^2}{\sigma_c^2 - \sigma^2} \left(\ln\frac{\sigma_c}{\sigma} + (n(c) - n(u))B\right) \tag{9}$$

meaning in case of $d[m,n,s]^2$ being greater than the right hand side we decide for 'c', otherwise 'u'. Hence, for the current position the label $q[m,n]$ =c is inserted and the $n(c)$ cliques are counted. Then this procedure is repeated for $q[m,n]$ =u obtaining $n(u)$ cliques. Thus, in case of more surrounding unchanged labels the decision threshold is altered in favour of unchanged, i.e. a logo pixel, otherwise for more surrounding 'c' in favour of changed.

For the sum of absolute differences as test statistic the decision rule reduces to

$$|d[m,n,s]| \underset{u}{\overset{c}{\gtrless}} 2\frac{\sigma_c\sigma}{\sqrt{2}(\sigma_c - \sigma)} \left(\ln\frac{\sigma_c}{\sigma} + (n(c) - n(u))B\right) \tag{10}$$

An overview of the logo detection procedure is given in Fig. 1. The initial masks provided by thresholding of the difference images are marked black. We consider the contour relaxation of the mask in the middle. The inhomogeneous cliques between the two previous and following masks are counted for the current position $[m,n]$. Based on the MAP criterion, the labels are evaluated and if required relabeled. The relabeled masks are adopted and we proceed to the next mask. The last step provides the final logo mask. By proceeding along the time axis, more and more changed labels should occur except for the logo.

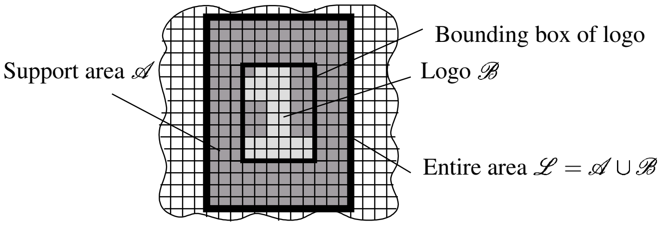

Figure 2: The principle of logo compensation: The samples belonging to the logo $\mathscr{B}$ (light gray) are extrapolated from the approximated support area $\mathscr{A}$ (dark gray).

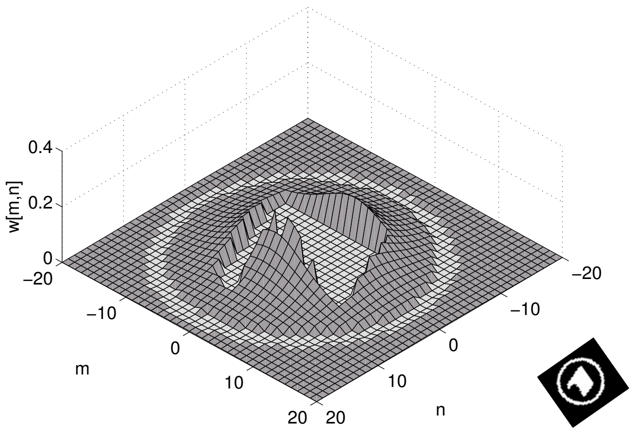

Figure 3: Weighting function using the isotropic model with $\hat{\rho} = 0.7$ for the provided logo mask.

3. LOGO INPAINTING BY FREQUENCY SELECTIVE EXTRAPOLATION

Fig. 2 shows the gray image region $\mathscr{L}$ containing a logo mask provided by the algorithm according to Sec. 2. In order to compensate the logo, we replace the pixels belonging to the logo by extrapolating the surrounding image signal. For detailed derivations and implementational issues the reader is referred to [4],[5]. Here, the principle of the spatial extrapolation technique and all innovations regarding the special case of logo inpainting are presented.

The pixels belonging to the logo are identified by the area $\mathscr{B}$. The support area $\mathscr{A}$ is formed by the pixels marked dark gray. Note

Figure 4: Two frames of original sequence 'DSF'.

Figure 5: Two frames of sequence 'DSF' with removed logo.

that the logo can be arbitrarily shaped and discontiguous. The size of the region $\mathscr{L}$ depends on the logo size. The logo is placed into a circumscribing rectangle marked black, the bounding box of the logo, and the width and the height of the logo is computed within this rectangle. In Fig. 2, $\mathscr{L}$ is formed extending the logo region by 0.5, in vertical and horizontal direction half of the logo size.

The gray values of the surrounding pixels in the support area $\mathscr{A}$ are denoted by $f[m,n]$. The parametric model $g[m,n]$ approximates the support area by a linear combination of weighted basis functions $\varphi_{k,l}[m,n]$ defined on the entire area $\mathscr{L}$

$$g[m,n] = \sum_{(k,l)\in\mathscr{K}} c_{k,l}\, \varphi_{k,l}[m,n]$$

with $c_{k,l}$ denoting the expansion coefficients and $\mathscr{K}$ the set of basis functions used. The number of available basis functions equals the number of coefficients in the entire area $\mathscr{L}$.

In order to determine the expansion coefficients, we minimize a weighted error criterion between the original signal and the parametric model evaluated with respect to the support area. The weighting function $w[m,n]$ has only non-zero amplitudes $\rho[m,n]$ in the support area

$$w[m,n] = \begin{cases} \rho[m,n] & , \quad (m,n)\in\mathscr{A} \\ 0 & , \quad (m,n)\in\mathscr{B}. \end{cases} \tag{11}$$

Hence, the following weighted error criterion is minimized during the approximation with respect to the support area

$$E_{\mathscr{A}} = \sum_{(m,n)\in\mathscr{L}} w[m,n]\,(f[m,n]-g[m,n])^2\,. \tag{12}$$

Due to the weighting function we can emphasize regions which are closer to the logo. We choose an isotropic model for $\rho[m,n]$

$$\rho[m,n] = \hat{\rho}^{\sqrt{(m-\frac{M}{2})^2+(n-\frac{N}{2})^2}};\ 0<\hat{\rho}<1 \tag{13}$$

with $\hat{\rho}$ being a prespecified constant. An example for a weighting function using the isotropic model is given on the left hand side of Fig. 3 using the provided logo mask of the right hand side.

The support area is approximated successively by computing one expansion coefficient $c_{k,l}$ per iteration requiring the following

Figure 6: Left: Detected logo 'DSF'. Right: Dilated mask.

two steps. First, that basis function $\varphi_{u,v}[m,n]$ is selected which maximizes the decrease of the residual error criterion. Then the respective coefficient $c_{u,v}$ is computed by minimizing the residual error criterion. Subsequently, the residual error signal in the support area is computed and further approximated by the next coefficient. The approximation of the support area stops when the decrease of the residual error energy drops below a prespecified threshold. The parametric model is then given in the entire area and the logo area is obtained by an inherent extrapolation.

We use 2D Discrete Fourier Transform (DFT) basis functions for the approximation which are especially suited in order to conceal monotone areas, edges and noise-like regions. The DFT allows an efficient realization in the frequency domain using FFT algorithms. The block $\mathscr{L}$ as shown in Fig. 2 is transformed into the DFT domain where the approximation takes place. No transforms are required in between. We compute one DFT coefficient per iteration until the algorithm terminates. The parametric model $g[m,n]$ is then given by the inverse DFT and the logo is cut out of $g[m,n]$. In summary, the support area is described by a few dominant features using spectral estimation and the area of the logo is given by an extrapolation.

In case of large logos, the logo is partitioned into blocks and the different blocks are processed subsequently. The prespecified target size of a block determines the number of blocks to be processed. For inpainting the logo, we want to exploit as much surrounding data as possible and align the processing order with. Hence, first the corner blocks and then inner blocks are inpainted, where for the inner blocks already inpainted blocks are used but they are weighted in order to limit the influence.

In scenes with no or slow motion, the extrapolation algorithm selects slightly different basis functions for the parametric model for each frame which causes a flicker effect. Therefore, if there are no significant changes in the scene the parametric model is used also for the next frame.

4. RESULTS

The logo removal algorithm was developed for sequences sampled from analog German television which is broadcasted in PAL format corresponding to 768×576 pixels at 25 Hz.

Assuming that there are no significant changes between two consecutive frames, the luminance component of every fifth frame of a 30 s sequence is read in for logo detection. First of all, we select the corner where the logo is placed based on the assumption that the logo pixels exhibit a certain luminance value during the entire sequence. Opaque white logos have the highest luminance values, whereas colored and semi-transparent logos have lower amplitudes. However, they all exceed a certain threshold in all frames. Hence, a temporary logo mask is generated with pixels exceeding the threshold $T_Y = 120$ in each frame and each iteration.

In order to exploit the temporal variations for logo detection, the sequence is divided into segments with constant contents because it is important that the images for generating the difference images really differ. The segmentation is based on the change of the mean of the luminance signal. A frame of another segment is subtracted from all frames belonging to one segment. All difference images of a segment are averaged over time in order to make the decisions for the initial logo masks more reliable.

Figure 7: Two frames of original sequence 'KIKA'.

Figure 8: Two frames of sequence 'KIKA' with removed logo.

The variance σ^2 of the camera noise was estimated according to [6] and the variance σ_c^2 was set to $\sigma_c^2 = 100\sigma^2$ [3]. For semi-transparent logos our assumption that changes at the logo position occur due to noise only does not hold anymore because the content within the logo region has changed. We thus cannot provide a global threshold for generating initial logo masks as input for the contour relaxation. Therefore, we exploit the temporary logo mask and in order to reach all logo pixels, we use the maximum difference value at the temporary logo position as the threshold T. For contour relaxation, the two previous and two proceeding change masks are used for evaluating (9) and (10), respectively. The procedure is repeated until no labels are changed anymore.

If the logo is not detected sufficiently, the entire process for logo detection can be repeated by reading in different frames, e.g. starting with the second frame. In order to generate difference images, the frame which is subtracted from all frames in a segment can be now from another segment as before. As already mentioned, it is important to use frames with differing contents.

In order to inpaint the logo, it is partitioned first. The target size for one processing block is 16×16 which is comparable to the usual size applied in block based video coding. 80% of the block-size is taken in each direction from the surrounding in order to form the block which is transformed by a FFT of size 64×64. We use the isotropic model with $\hat{\rho} = 0.8$ and already inpainted blocks are weighted by 0.1 in order to include them with limited influence. The extrapolation terminates when either the residual energy decrease drops below a threshold ($\Delta_{\min} = 5$, see [5]) or a maximum number of 25 iterations is reached. If the scene is changing slowly, the parametric model is used for the next frame. After maximally adopting the inpainted logo area five times, a new parametric model is computed. By this, the flickering is reduced significantly.

Fig. 4 shows two parts of 180×240 pixels of sampled frames from a soccer game containing the opaque and white logo 'DSF'. The detected logo mask using the squared sum as test statistic (9) with $\sigma^2 = 5.3$ and $B = 90$ is depicted in Fig. 6 on the left hand side. However, the mask is dilated once, since the white logo exhibits a darker shadow around the letters in order to stand out on a bright background but which is not always present to the same extent. The logo consists of 20×87 pixels leading to a 2×6 partitioning. In Fig. 5 the frames with the removed logo are presented. The algorithm can inpaint both periodic structures like the seats of the tribune at the left hand side as well as smooth areas on the right hand side.

Figure 9: Left: Detected logo 'KIKA'. Right: Dilated logo mask accounting for smearing effects of colored logo.

Figure 10: Left: Orginal frame with semi-transparent logo 'RTL'. Right: Dilated logo mask 'RTL'.

Further, we present the removal of a colored logo. In Fig. 7 parts of two frames containing the colored logo 'KIKA' are shown. The detected logo mask is depicted on the left hand side of Fig. 9 obtained with the squared sum as test statistic (9) using $\sigma^2 = 2.0$ and $B = 90$. Looking closely to the left hand side of Fig. 7 shows that the colors of the logo are spread to the surrounding of the logo. However, in the frame on the right hand side this effect cannot be observed. Therefore the logo mask is dilated in order to account for these smearing effects, because otherwise the colors would appear in the inpainted logo causing artefacts. The logo is partitioned into 3×7 processing blocks due to its size of 34×111 pixels.

In Fig. 10 we present the detection of a semi-transparent logo which is only possible due to change based detection. On the left hand side a frame of the sequence is shown and on the right hand side the detected and dilated final logo mask. For logo detection we use the sum of absolutes as test statistic according to (10) with $B = 30$ and $\sigma^2 = 6.1$.

REFERENCES

[1] W.-Q. Yan and M. S. Kankanhalli, "Erasing video logos based on image inpainting," in *Proc. Int. Conf. on Multimedia and Expo (ICME)*, Switzerland, Aug. 2002, vol. 2, pp. 521–524.

[2] A. Albial, M. J. C. Fullà, A. Albial, and L. Torres, "Detection of TV commercials," in *Proc. Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, Canada, May 2004.

[3] T. Aach, A. Kaup, and R. Mester, "Statistical model-based change detection in moving video," *Signal Processing*, vol. 31, pp. 165–180, March 1993.

[4] A. Kaup, K. Meisinger, and T. Aach, "Frequency selective signal extrapolation with applications to error concealment in image communication," *to appear in Int. J. Electron. Commun. (AEÜ)*, June 2005.

[5] K. Meisinger and A. Kaup, "Minimizing a weighted error criterion for spatial error concealment of missing image data," in *Proc. Int. Conf. on Image Processing (ICIP)*, Singapore, Oct. 2004, pp. 813–816.

[6] J. Immerkaer, "Fast noise variance estimation," *Computer Vision and Image Understanding*, vol. 64, no. 2, pp. 300–302, Sept. 1996.