ROBUST ALGORITHM FOR WATERMARK RECOVERY FROM CROPPED SPEECH

Aparna Gurijala and J. R. Deller, Jr.

Michigan State University
Speech Processing Laboratory
Department of Electrical & Computer Engineering / 2120 EB
East Lansing, MI 48824-1226 USA

ABSTRACT

Most digital watermarking techniques are susceptible to damage by data cropping. Although the effects of cropping might not be perceptible, watermark recovery may be rendered difficult or impossible due to the desynchronization of the recovery process. The transform encryption coding (TEC) based watermarking algorithm was presented at ICASSP 2000 [2]. The present paper investigates the performance of TEC watermarking in the presence of cropping, and presents an algorithm that identifies cropped samples and recovers watermarks from the damaged record. Implementation details and experimental results under different environmental conditions are presented.

1. INTRODUCTION

The speech watermarking technique developed by Ruiz *et al.* [2] employs transform encryption coding (TEC [3]) in conjunction with a masking algorithm for encrypting and watermarking speech. The encryption capabilities of TEC are achieved through an all-pass filtering process that has special significance in the developments to follow. For increased security the filter coefficients should exhibit a high degree of randomness. Quasi m-arrays [4] are used to achieve the desired absence of predictability in the filter [2].

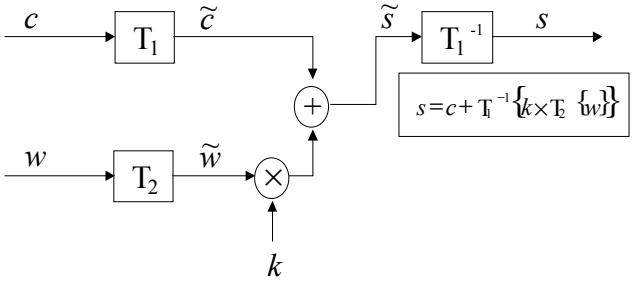

Figure 1. Watermarking process.

The watermarking process involves the application of TEC to both the *coversignal* (original speech) and the watermark. The encrypted watermark is subjected to a masking algorithm to ensure perceptual transparency based on the *cover to watermark ratio* (CWR), defined as

$$CWR_{dB} = 10\log_{10}\frac{C[m]}{k[m]\times W[m]}\ , \qquad (1)$$

where $C[m]$ and $W[m]$ are the respective short-term energy measures [8] for the cover and watermark signals, and $k[m]$ is an adaptive gain factor at time m. Alternately, a constant gain factor k can be used instead of $k[m]$. Since the encryption process involves passing the cover and watermark signals through an all-pass filter, the short-term energy measures of the encrypted and non-encrypted signals are similar. The encrypted and masked watermark is then added to the encrypted coversignal to obtain the encrypted *stegosignal*. Applying the inverse TEC operation to decrypt the stegosignal subjects the watermark to a second level of encryption.

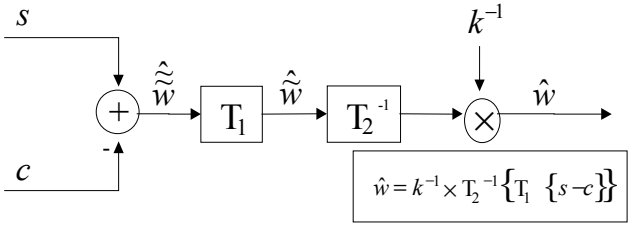

Figure 2. Watermark recovery.

For watermark recovery, an estimate of the doubly-encrypted watermark is obtained by subtracting the coversignal from the stegosignal,

$$\hat{\tilde{\tilde{w}}} = s - c\ . \qquad (2)$$

Finally the inverse TEC operations and the gain factor are applied to the estimated twice-encrypted watermark (Fig. 2):

$$\hat{w} = k^{-1}\times T_2^{-1}\left\{T_1\left\{\hat{\tilde{\tilde{w}}}\right\}\right\} = k^{-1}\times T_2^{-1}\left\{T_1\left\{s-c\right\}\right\}. \qquad (3)$$

The recovery of the watermark is only possible with knowledge of the two quasi *m*-arrays (keys) used in the process. In Ruiz's original paper [2], the entire speech record is watermarked, but only certain frames of speech may be watermarked depending upon the requirements of the application. The use of keys of

higher dimension means higher encryption security. However, a trade-off is involved between increased security and the limitations on real-time processing.

2. WATERMARK RECOVERY FROM CROPPED SPEECH

2.1. Cropping

Cropping is an attack on the content wherein samples of the host signal are randomly deleted. About 1 in 50 speech samples may be cropped without introducing any perceptible difference [6]. Cropping is one of the most destructive attacks on any watermarking scheme because it desynchronizes the decoding process, making watermark recovery difficult or impossible. Hence, there is a need for an algorithm to identify the cropped samples and to undo the damage caused by cropping so as to recover the watermark.

2.2. Watermark recovery after cropping

A recovery algorithm is presented which is based on the concept of dynamic programming [5]. Consider the *i-j* plane (as shown in Fig. 3) with the cropped stegosignal (test string) along the *i*-axis and the stegosignal (reference string) along the *j*-axis. Determination of the cropped samples is treated as the problem of finding the minimum distance path through the grid. A path is a collection of nodes of the form $(t(i), s(j))$ connecting the original and terminal nodes. Distances or costs are assigned to paths in the form of nodal costs. The cost associated with the node $(t(i), s(j))$ is defined as,

$$D_n(i, j) = (t(i) - s(j))^2. \quad (4)$$

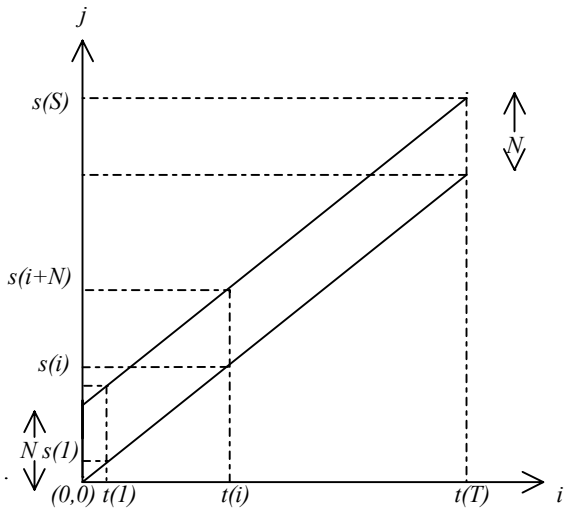

Figure 3. Dynamic programming approach for recovering cropped samples.

Let S be the length of the stegosignal and T be the length of the cropped stegosignal. Assuming no additional or duplicate samples are added to the stegosignal, the number of samples cropped is

$$N = S - T \quad (5)$$

***Search constraints**:* Constraints on the search region are imposed to limit the amount of computation and to ensure appropriate matching between the test and reference strings.

Monotonicity. For the path to be monotonic it must advance in the upward direction, i.e., it should not go "south" or "west" in the grid. Further, movement of the path in the horizontal or the vertical direction is prohibited as a single test sample cannot be associated with more than one reference sample and vice versa.

Global path constraints. Since N samples are cropped and the path can only move in the upward direction, element $t(i)$ of the cropped stegosignal can be matched only with the $(N+1)$ elements $s(i)$ *to* $s(i+N)$ of the stegosignal. The same constraint is also applied at the endpoints. The search region is the diagonal strip shown in Fig.1.

Local path constraints. As every sample of the cropped stegosignal is contained in the *original* stegosignal, the optimal path should include all the test string elements. That is, no skips are permitted along the *i*-axis. At most, N reference string samples may be skipped in the process of finding the optimal path, as N samples were cropped.

Thus, for node $(t(i), s(j))$ in the search region, the possible immediate predecessor nodes include $(t(i-1), s(k))$ where k ranges from $(i-1)$ to $(j-1)$.

As a consequence of *Bellman optimality principle* [5] the optimal path to the node $(t(i), s(j))$ can be found by considering the best paths associated with all the possible predecessor nodes and choosing the one with the minimum cost,

$$D_{\min}(i, j) = \min_{(i-1,k)} \{D_{\min}(i-1, k) + d_n(i,j)\}, \quad k = (i-1),\ldots,(j-1). \quad (6)$$

When all the nodes in the search region are considered, a set of N optimal paths is obtained and the global optimal path is the one associated with least cost among them. At every node $(t(i), s(j))$ of a particular optimal path, it is necessary to record the immediate predecessor node from which the path was extended. This way the path may be reconstructed by backtracking beginning at the terminal node.

The overall algorithm involves the following steps.

1. *Initialization:* The original node is $(0,0)$ and the nodal cost associated with it is zero.
   $D_{\min}(1, j) = d_n(0,0), \quad j = 1,\ldots,(1+N)$
   $\psi(1, j) = (0,0), \quad j = 1,\ldots,(1+N)$
   $\psi(1, j)$ = the index of the predecessor node to $(1, j)$.
   $\delta_l(j) = D_{\min}(1, j), \quad j = 1,\ldots,(1+N)$
2. *Recursion*:
   For $I = 2,\ldots,T$
   For $j = i,\ldots,(i+N)$
   Compute $D_{\min}(i, j)$ using (6).
   ($\psi(1, j)$ is recorded for every (i, j)).
   $\delta_l(j) = D_{\min}(i, j)$
   Next j
   Next I
3. *Termination:* The best path is the one associated with the

least cost,

$$\min \{D_{\min}(T, j)\}, \quad j = T, \ldots, (T+N)$$

4. *Reconstruction:* The best path accurately identifies samples of the cropped stegosignal that are present in the stegosignal. The cropped samples are the ones, which are not present in the stegosignal. The reconstructed stegosignal can be obtained easily by reinserting the cropped samples at the appropriate places.
5. *Watermark recovery:* The watermark recovery process is now applied to the reconstructed stegosignal.

Memory and computational requirements*:* The algorithm requires about $(N+1)T$ nodal costs or distance measures to be computed and $((N+1)(N+2)T)/2$ implementations of equation (6). Considering the memory requirements, a matrix of size $O(TS)$ must be allocated for backtracking. To compute $D_{\min}(i, j)$ at every (i, j) within the search region, it is necessary to have just the past $D_{\min}(i-1, j)$ values for $j = (i-1), \ldots, (j-1)$. Therefore, at the most an array of dimension $1 \times (N+1)$ is required assuming that the computation can be done in-place.

3. EXPERIMENTAL RESULTS

The software implementation of the algorithm was done in Matlab. The stegosignal was subjected to cropping using the robustness testing engine for speech watermarking developed by Ruiz *et al*.

As an example, the dynamic programming based watermark recovery algorithm was applied to a cropped version of speech, watermarked using the TEC-based watermarking technique. The coversignal was obtained from the TIMIT speech database [8] and has a male voice saying: "She had your dark suit in greasy wash water all year." It is a 1-second signal, sampled at 16kHz with 16-bit quantization. "Lena" image was used as the watermark. Alternatively, speech can also be used. Experiments have shown that the results are comparable for image and speech watermarks. The stegosignal thus obtained was fed into the robustness-testing engine. About 13 samples were randomly cropped from the stegosignal, which consisted of 16129 samples.

The watermark recovery algorithm accurately identified the cropped samples. The watermarks were then recovered (refer Fig. 4) from the reconstructed stegosignal. The algorithm was tested for different CWRs and for a different number of cropped samples. In all the cases the cropped samples were accurately determined.

3.1. Cropping in the presence of noise

The TEC-based speech watermarking technique is satisfactorily robust to uncorrelated random noise. However if cropping is present in addition to random noise, then it would be impossible to obtain the watermark recovery signal (refer equation (2)). The difference between the coversignal and stegosignal is no longer pertinent due to their misalignment.

The watermark recovery algorithm accurately reconstructed the cropped stegosignal under fairly noisy conditions. Table 1 shows the performance of the recovery algorithm under different conditions. In all the cases, one second of the speech "Theodore Roosevelt talks about Wilson and Taft" [7], taken from Vincent

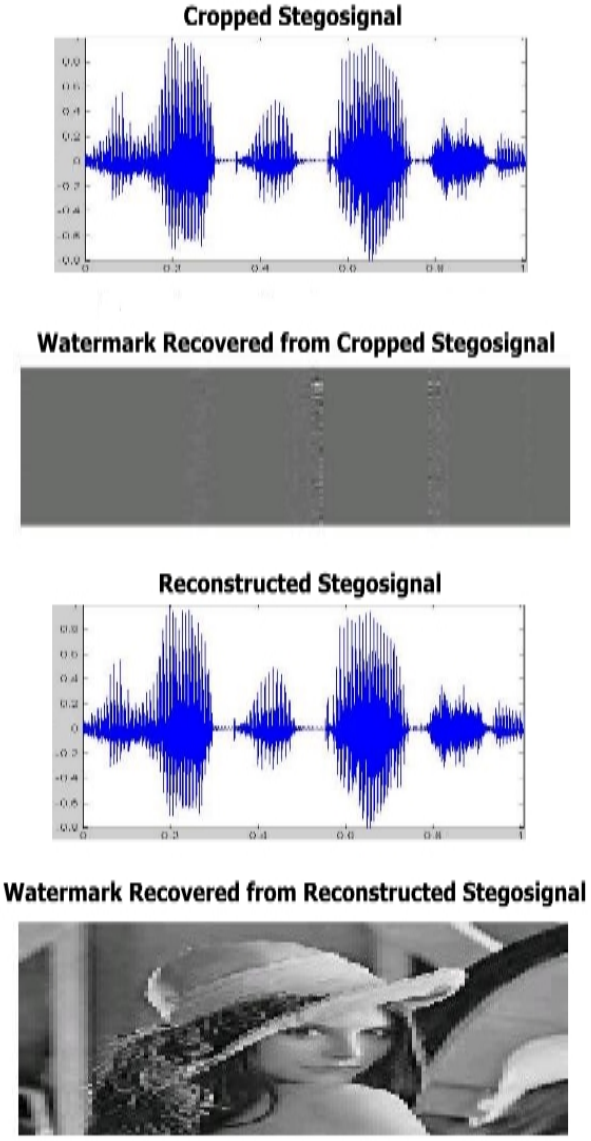

Figure 4

Voice Library at Michigan State University was used as the coversignal. The signal is monaural, sampled at 16kHz with 16-bit quantization. The "Mandrill" image was used as the watermark. The CWRs (greater than or equal to 20dB, using a constant gain factor) were chosen so as to ensure the imperceptibility of the watermark in the cover signal. Gaussian random noise was used in all the cases. One sample out of every 1200 samples was cropped. Similar results were obtained when the number of cropped samples was increased or decreased.

Table 1 also makes use of normalized correlation between the original and distorted watermark recovery signals, which are obtained by taking the difference between the respective

stegosignals and the cover signal. If s' is the distorted stegosignal then

$$s' = c + \hat{\tilde{w}} + d_n = c + w' . \quad (7)$$

The normalized correlation between $\hat{\tilde{w}}$ and w' is defined as,

$$Ds = \hat{\tilde{w}}.w' \Big/ \left|\hat{\tilde{w}}\right| |w'| \quad (8)$$

A high value of Ds indicates the presence of the watermark. Due to the misalignment between the watermark recovery signals when cropping is present, Ds fails to detect the presence of the watermark. If the cropped stegosignal is appropriately reconstructed using the watermark recovery algorithm, then Ds returns a high value whenever the watermark is present.

CWR (dB)	Noise $N(\mu,\sigma)$	SNR (dB)	Cropped samples accurately determined Yes/No	Watermark detection based on Normalized correlation.
20	$N(0, .12)$	3.569	No	0.5733
20	$N(0, .05)$	11.37	Yes	0.8528
20	$N(0.003, 0.0303)$	15.47	Yes	0.9389
20	$N(0, .03)$	15.54	Yes	0.9386
25	$N(0, .05)$	11.07	Yes	0.8531
20	$N(0, .03)$	15.54	Yes	0.9389

Table 1: Cropping in the presence of noise. Correct determination of cropped samples depends upon the SNR.

The accurate determination of the cropped samples and the reconstruction of the distorted stegosignal are dependent on the signal to noise ratio (SNR). Higher SNRs guarantee watermark detection and recovery. However, the recognizability of the recovered watermark in the presence of noise is dependent on the energy of the watermark signal. For a given SNR, if the watermarking process involved the use of a lower CWR, then the recovered watermark will be easily recognizable than watermarks recovered from a process making use of a higher CWR. The CWR value is bounded below by the need to ensure the perceptual transparency of the watermark in speech. For a given CWR, it is better to embed the watermark in regions where the speech has high energy. In these regions, the inserted watermark will also have a higher energy. Thus the watermark would be more robust to attacks.

When the SNR is very low (refer Table 1), the recovery algorithm fails to detect the cropped samples properly. The fidelity of the stegosignal is very low for these SNR values and its commercial value might be lowered.

The watermark recovery algorithm for cropped speech is associated with a zero false positive rate. This is because even if one cropped sample is not properly determined, the recovery process is affected due to the desynchronization effect.

4. CONCLUSIONS

A robust algorithm for watermark recovery from cropped speech has been described. The algorithm was tested under different environmental conditions. SNR and CWR were two important parameters used in the performance evaluation of the algorithm. Higher SNR and lower CWR contribute towards better performance and increased robustness respectively. However, the CWR is limited by the necessity to ensure the imperceptibility of the watermark.

With some modifications, the algorithm can easily be extended for watermark recovery after resampling.

Further work will involve the study of the robustness of the TEC-based watermarking scheme to other possible attacks. More elaborate robustness-testing engine for speech watermarking schemes needs to be developed. Future work will also comprise utilization of the compression properties of TEC in conjunction with watermarking.

REFERENCES

[1] http://www.ngsw.msu.edu/

[2] Fco. J. Ruiz and J.R. Deller, Jr., "Digital watermarking of speech signals for the national gallery of the spoken word," *IEEE International Conference on Acoustics, Speech and Signal Processing 2000,* Istanbul, Turkey, May 2000 (published on CD).

[3] C.J. Kuo, J.R. Deller, Jr. and A.K. Jain. "Pre/post-filter performance improvement of transform coding," *Signal Processing: Image Communication,* vol. 8, 1996, pp. 229-239.

[4] C.J. Kuo and H.B. Rigas, "2-D quasi m-arrays and Gold code arrays," *IEEE Trans. Information Theory*, vol. 37, Mar. 1991, pp. 385-388.

[5] J.R. Deller, Jr., J.H.L. Hansen, and J.G. Proakis, *Discrete Time Processing of Speech Signals* (2d ed.), New York: IEEE Press, 2000.

[6] R.J. Anderson and F.A.P. Petitcolas, "On the limits of steganography," *IEEE Journal of Selected Areas in Communications,* May 1998, pp. 474-481.

[7] "Theodore Roosevelt talks about Wilson and Taft" audio file, Vincent Voice Library, Michigan State University Libraries, http://www.lib.msu.edu/vincent/t_roosevelt.ram.

[8] W.M. Fisher, G.R. Doddington, and K.M. Goudie-Marshall, "The DARPA speech recognition research database: Specifications and status," *Proceedings of the DARPA Speech Recognition Workshop*, pp. 93-99, 1986.

[9] L. Boney, A.H. Tewfik and K.N. Hamdy, "Digital watermarks for audio signals," *IEEE International Conference on Multimedia Computing and Systems,* Hiroshima, June 1996, pp. 473-480.