Embedding and Retrieving Private Metadata in Electrocardiograms

Suleyman S. Kozat [†] Michail Vlachos [†] Claudio Lucchese [⋆]
Helga Van Herle [‡] Philip S. Yu [†]

[†] IBM T.J. Watson Research Center
[‡] David Geffen School of Medicine, UCLA
[⋆] University of Venice

Abstract

Due to the recent explosion of ’identity theft’ cases, the safeguarding of private data has been the focus of many scientific efforts. Medical data contain a number of sensitive attributes, whose access the rightful owner would ideally like to disclose only to authorized personnel. One way of providing limited access to sensitive data is through means of encryption. In this work we follow a different path, by proposing the fusion of the sensitive metadata within the medical data. Our work is focused on medical time-series signals and in particular on Electrocardiograms (ECG). We present techniques that allow the embedding and retrieval of sensitive numerical data, such as the patient’s social security number or birth date, within the medical signal. The proposed technique not only allows the effective hiding of the sensitive metadata within the signal itself, but it additionally provides a way of authenticating the data ownership or providing assurances about the origin of the data. Our methodology builds upon watermarking notions, and presents the following desirable characteristics: (a) it does not distort important ECG characteristics, which are essential for proper medical diagnosis, (b) it allows not only the embedding but also the efficient retrieval of the embedded data, (c) it provides resilience and fault tolerance by employing multistage watermarks (both robust and fragile). Our experiments on real ECG data indicate the viability of the proposed scheme.

I. Introduction

In the years to come, the Healthcare system is expected to experience a drastic change in its structure and organization. These changes are partly driven by changes in the human genographics, and also reinforced by the recent climatic changes and the various events and disasters throughout the world. This shift is clearly reflected on recent Healthcare reports. For example, the *Healthcare 2015 report* [1] shows that governments, health regions, hospitals, and healthcare providers are allotting billions of dollars into multiple medical initiatives.

One very important effort is the creation of electronic health records (EHR’s). As health care (and health care data) grows more complex, storage and accessibility of medical information is not only invaluable but also necessary. The long-term goal for electronic health records is to make patient data securely available to health care providers such as hospitals and emergency personnel, when and where the information is needed. Disasters, such as Hurricane Katrina, for example, have shown the practical utility of being able to store and retrieve information like prescription histories and dosages electronically in an emergency.

One of the major technological and ethical issues governing electronic records is the issue of data privacy. Protection from unauthorized access on medical history data and personal patient data, is something that can not only protect a patient’s private data hindering potential identity thefts, but can also safeguard the healthcare and insurance system from fraudulent claims. With this in mind, this work proposes techniques for hiding sensitive patient metadata within the actual medical measurements, which are stored into a patient’s medical record. In specific, we focus on electrocardiograms (ECG’s) and how to embed numerical metadata within the ECG signals.

A prerequisite of this embedding is, of course, not to destroy the data usability. We indeed show that the usefulness of the data is not affected, because of the imperceptible distortion induced through the fusion of the metadata within the actual data. For most watermarking applications this requirement can simply be stated as preserving the visual/audio quality of the signal (i.e., for image and audio processing). When dealing with medical data this means that our watermarking algorithms should not change the diagnosis of a physician. For example,

[1] `http://healthnex.typepad.com/web_log/2007/02/web_seminar_rep.html`

when dealing with ECG signals, common tasks are the detection of arrhythmia or other heart related illnesses. Therefore, the diagnosis on the watermarked signal should not deviate from the diagnosis on the original signal.

The privacy of the embedded data is assured because we do not embed directly the private metadata, but instead we embed a surrogate random sequence, that is generated by a cryptographically safe hash function using the metadata as the input and a secret key as the seed. Hence, we avoid leaking or revealing any information about the patient's sensitive information to the public. Even though the privacy of sensitive data attributes can be addressed through encryption, such an approach is inherently a blocking factor in data dissemination. Additionally, the use of encrypted fields in medical records directly suggests the existence of private data, which may be something that one would like to avoid in the first place.

The tight coupling of the metadata within the actual medical measurements presents several desirable properties: 1) Private information is effectively concealed in the signal and therefore can serve as an additional authentication seal regarding the originality of the data. 2) The fusion of the metadata within the actual data can potentially eliminate the need for recording the patient metadata separately. This could provide an additional level of security on the private information of a patient, by thwarting deliberate changes on the medical records, or even by eliminating accidental errors during a laborious replicating/typing process of a patient's record fields. 3) Finally, the techniques that are delineated here could also be applied for establishing the provenance [1] of the data. Therefore, if every recipient (or processor) of the data embed a different secret watermark, then one can trace the lineage of how the data was produced, processed and distributed in a transparent fashion.

In the experimental section we also demonstrate that the fusion of the metadata with the data is achieved in such a way, so that the data usability is not hindered or affected. The upcoming sections, will explicate in more detail the challenges and also the advantages of the proposed embedding.

II. Overview

In order to embed metadata within the medical signals, we will utilize notions from data watermarking and channel coding. The sensitive metadata (social security number (SSN), birth date, and so on) will be embedded as a hidden watermark within the medical measurements of the patient. In order to provide additional protection and data resilience we propose to embed two types of watermark on the medical signal; a robust one for storing the actual metadata and a fragile one for identifying possible tamperings on the data:

1) **The robust watermark** will encode an encrypted version of the patient's metadata, employing additional data redundancy for aiding data recovery in the case of data corruption by a malicious attacker. We show that a robust watermark cannot be easily removed without significantly distorting the actual data, i.e., without obvious attacks, which in any case will render the data useless.
2) **The fragile watermark** will be used for detecting potential data tampering. As the name suggests, simple operations can destroy the fragile watermark, but its absence on the received data is an indication that the data have been compromised or altered.

Here, we introduce novel robust and fragile watermarking approaches and apply them to medical time-series data. We strongly emphasize on the randomized nature of our algorithms. We use extensive randomization in every step of our algorithms to alleviate the vulnerability of our algorithms against malicious attackers or common alterations on the host signal. Although, it is hard or impossible to model every kind of reasonable quality preserving attacks, the utilization of randomized step significantly reduces the leakage of information to possible attackers. The robust watermark encoding the actual metadata is embedded in the frequency domain, and the data is masked effectively in certain frequencies that are selected based on a secret key. This type of embedding makes the embedded data resilient to transformations such as translations, least significant bit alternations, small noise additions, resampling and decimation. Furthermore, the regions where the private metadata are embedded are selected based on a secret key. A fraction of the hidden metadata bits will be allocated for employing error correction codes, in order to provide additional resiliency due to malicious attacks, or even due to transmission errors. In this sense, our watermarking approach not only uses ideas from spread spectrum based algorithms [2], but also has connections to watermarking techniques motivated by traditional cryptography [3].

The fragile watermark will be embedded after the robust watermark on the least significant bits at specially selected positions of the ECG signal. The fragile embedding will introduce virtually no distortion. Notice, that even though the fragile watermark is embedded on top of the robust, it cannot destroy the robust watermark which is able to withstand such minor (or even more significant) transformations. A overview of this architecture is provided in Figure 1.

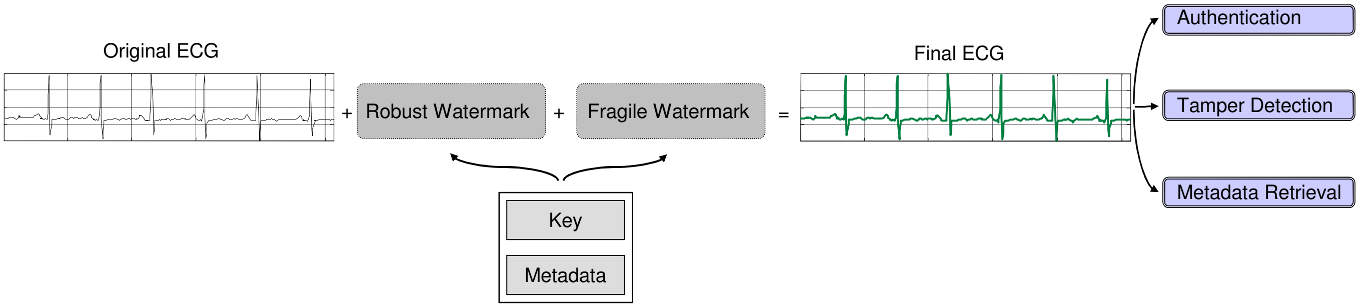


Fig. 1. Overview of our approach

Once the metadata are effectively fused within the medical signal, there are three supported modes of operation:

1) **Tamper Detection** by examining the presence of the fragile watermark.
2) **Data Authentication** through correlation with the originally embedded metadata. For example, if the SSN of a patient is embedded in an ECG signal, then using the SSN and a secret key, one can verify that the data indeed belong to the patient with a specific Social Security Number.
3) **Metadata Retrieval.** The rightful owner of the data can provide the secret key to someone else, who is now at a position to retrieve the embedded metadata from the medical signal.

A. Related Work

Watermarking research in multimedia data is a very rich field. Compared to traditional watermarking our work exhibits various differences, such as the fact that we provide the ability of *retrieving back* the embedded metadata. For this reason we also augment our watermarking technique with coding redundancy schemes, in order to achieve better data preservation and provide the ability for error correction. Additionally, because we are dealing specifically with medical ECG signals, we can exploit their regularity for tailoring more appropriately the metadata encoding scheme. Previous research work dealing with the watermarking of medical signals have appeared for ECG data [4] and for electroencephalograms (EEG's) [5]. However, to our knowledge, this is first work that considers metadata fusion within the medical signal, not only for reasons of authentication, but also for providing the ability of heterochronous metadata retrieval.

Related is also the work of [6] which watermarks numeric streams, by embedding the watermark on easily identifiable stream positions, such as the local maxima and minima. However, such an approach might not be ideally suited for ECG signals, where one would like to preserve as well as possible such areas, because of their significance in medical diagnosis. Therefore we spread the intensity of the robust watermark using a spread spectrum approach. Additionally, the Least Significant Bit (LSB) alteration that we employ in the fragile watermark is quite more advanced, in the sense that it can also pinpoint the area and type of tampering.

Watermarking work has also been used in relational databases either using direct LSB alterations [7], [8] or using hierarchical Binning approaches [9]. Finally, there is a vast literature on the topic of privacy preserving data-mining [10]–[12]. Compared to the above areas, our approach is different regarding the goals and the methodological approach that we follow.

In the upcoming sections we will describe the embedding and the retrieval of the robust and the fragile watermarks. We will also demonstrate the minimal distortion that is introduced by their embedding and in the experimental section we will empirically assess the resilience of our scheme.

III. Robust Watermarking

A. Preliminaries and Notation

Consider an ECG signal as a one dimensional time-series sequence, represented as a vector $x = \{x_1, \ldots, x_n\}$, where $x_k \in \mathbb{R}$. In such a signal, we will embed private numeric metadata by adapting on *watermarking* techniques.

However, we will also show how to retrieve back the hidden information, which is something that traditional watermarking applications do not consider. Therefore, our technique gracefully fuses watermarking and channel coding techniques. The *secret information* that will be hidden inside each ECG signal itself is encoded as a watermark $W \in \{-1, 1, 0\}^n$, which has the same length as x and can take 3 distinct values. Later we will show how we can use the sequence W to encode numeric metadata consisting of l-bits.

The embedding of the watermark consists of a composition function that, given x and W, returns a modified signal which is *similar* to x and *encloses* W. The original ECG signal should not be significantly distorted and a technique to retrieve/detect W in the watermarked signal should be provided.

We call this watermark *robust* because it is able to withstand a variety of possible data transformations. We will not embed the watermark in the original *Space-Time domain* but into the *Frequency domain*, which will guarantee better resilience against malicious attacks.

Every ECG signal x will thus be represented with the set of its Fourier descriptors $X = \{X_1, \ldots, X_n\}$ where n is the number of points of x as well as the number of its frequency components. The mappings from one domain to the other are described by the discrete Fourier transform $\mathit{dft}(x)$:

$$X_j = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} x_k \exp\left(-i\frac{2\pi}{n}(j-1)(k-1)\right)$$

and the inverse discrete Fourier transform $\mathit{idft}(X)$:

$$x_j = \frac{1}{\sqrt{n}} \sum_{k=1}^{n} X_k \exp\left(i\frac{2\pi}{n}(j-1)(k-1)\right).$$

Every coefficient X_j can be described in terms of its *magnitude* ρ_j and *phase* ϕ_j, that is, $X_j = \rho_j e^{\phi_j i}$.

We use an additive embedding of the watermark which alters only the magnitudes but retains the original phase:

Definition 1 (Additive Fourier Embedding): For a signal $x \in \mathbb{R}^n$ and a watermark $W \in \mathbb{R}^n$, the *additive Fourier embedding* generates a watermarked signal $\widehat{x}$ by replacing the magnitudes of each Fourier descriptor of x with a watermarked magnitude $\widehat{\rho_j}$:

$$\widehat{\rho_j} \;=\; \langle \rho_j + pW_j \rangle \;\overset{\text{def}}{=}\; \max(0, \rho_j + pW_j)$$

where *power* $p > 0$ specifies the intensity of the watermark.

Notice that we use the function $\langle \cdot \rangle$, in order to ensure that we have no resulting negative magnitudes, when $W_j = -1$. We will explain later that this may introduce a power loss into the watermarking procedure. Using the modified magnitudes $\widehat{\rho_j}$ and the original phases ϕ_j, we go back from the frequency domain to the time domain and reconstruct the watermarked sequence using the inverse discrete Fourier transform.

B. Watermark Construction

Let us describe now how the private metadata are embedded into the hidden watermark. First, let's recall that the watermark W will consist of the values +1, -1 and 0. Understandably, only those W_j's that contain +1 or -1 will introduce some alteration in the respective signal frequencies. Thus, only those W_j can encode some information. Conversely, the zero values of W determine the descriptors that we do not want to modify.

The choice of which Fourier descriptors (frequencies) are most suitable to be altered, i.e., to be actually used for the embedding, can affect the goodness of the detection process. Our goal is to build an unbreakable bond between a signal and a embedded watermark. On the other hand, a potential attack cannot alter the overall shape of the ECG plot, i.e. damage its usability. Therefore we should tie the embedded metadata W with the most important frequencies. It is well established that the first descriptors hold almost all the energy of ECG signals, which means that they describe very accurately the data.

Driven by these considerations, we will focus on embedding the watermark in the lowest frequencies. However, we will not embed any portion of the watermark on the first Fourier descriptor X_1, since the DC component of the signal x ($X_1 = \sum_j x_j/\sqrt{n}$) is easily susceptible to attacks. For example, a simple translation will change the

DC level of x (that is, X_1) without affecting its shape, but it will erase this part of the watermark. Therefore we embed the watermark into the 2^{nd} and up to the $(l+1)^{th}$ Fourier descriptor, where l is the number of non zero elements of W. Then, the watermark W is formally defined as follows:

$$W_j = \begin{cases} 0 & \text{if} \quad j = 1 \quad \text{(DC component)} \\ \{-1, 1\} & \text{if} \quad 2 \leq j \leq l+1 \\ 0 & \text{if} \quad l+2 \leq j \leq n \end{cases}$$

The metadata that one wishes to embed in an ECG signal will be represented with a sufficiently long bit-string. In order to provide additional resilience to attacks, we introduce additional pre-processing before materializing the watermark W. Let $B(I)$ be the binary representation of the information I (e.g., metadata), which is randomly generated using the original information and part of the secret key κ. Details of this pre-processing would be clear later on. We prefer a randomized representation of the metadata in order to protect the private information of the patient. We next produce an error correcting code of $H_{7,4}(B(I))$ using the *Hamming(7,4)* coding. Introducing channel coding is mainly used to detect errors during the transmission of bit-streams over a noisy channel. This process introduces a controlled level of redundancy by mapping an input of 4 bits into a code of 7 bits. Due to this added redundancy, the receiver of the message will be able to *correct* 1-bit errors and *detect* 2-bit errors. In the same way, we will detect malicious attacks that may *flip* one or more bits of the embedded watermark. We adopted the Hamming(7,4) encoding for its simplicity, but more complicated and effective techniques could be utilized as well. The Reed-Solomon code, for instance, is currently used in CDs and DVDs and it provides augmented correction capabilities.

Given the above, the embedded watermark that can encode the metadata I is defined as follows:

$$W_j = \begin{cases} 0 & \text{if} \quad j = 1 \quad \text{(DC component)} \\ 1 & \text{if} \quad (j-1)\text{-th bit of } H_{7,4}(B(I)) = 1 \\ -1 & \text{if} \quad (j-1)\text{-th bit of } H_{7,4}(B(I)) = 0 \\ 0 & \text{if} \quad l+2 \leq j \leq n \end{cases}$$

where $l = |B(I)|$ is the length of the binary representation of I.

As explicative example, for the rest of the paper we will use the social security number (SSN) as the metadata to be embedded in a given ECG plot. The SSN in the United States, consists of 8 digits in the form $999-99-999$. Any number $< 10^8$, can be represented with a 27-bit long string, which for conciseness let us call *binary*(SSN). This initial representation can be as simple as the binary conversion of the decimal SSN. The binary representation *binary*(SSN) is then inputted into a cryptographically secure hash function with κ as the secret key to produce the final randomized 27 bit long string $B(\text{SSN})$ as seen in Figure 2. This representation is then divided into seven chucks of four bits each and then Hamming coding is applied independently to each chuck. The result is a $l = 49$ bits long error correcting code enclosing a given SSN.

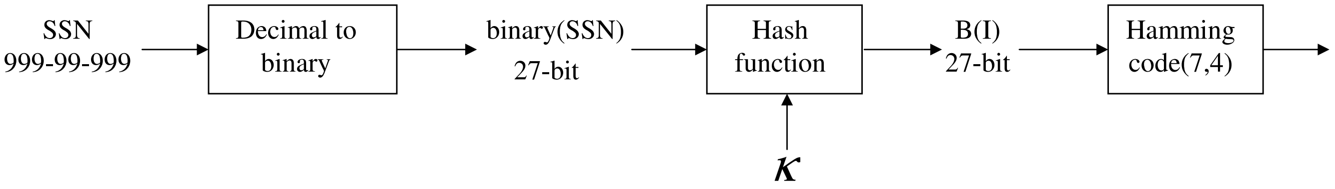


Fig. 2. Randomized binary representation of SSN metadata, through hashing and Hamming coding.

Example: Suppose we watermark an ECG signal of a patient with SSN=123456789. When we convert decimal SSN to a binary number, this yields *binary*(SSN)=111010110111100110100010101. We next select randomly $\kappa = 9823$ as the key to a hash function with *binary*(SSN) as the input, resulting $B(\text{SSN})$=110110100001101001101111110111. This pseudo-random representation of the original SSN is then divided into chunks of 4 bits (we append the last chunk with bit 0 since 27 is not divisible by 4) and each 4-bit chuck is used to generate a 7-bit chunk using Hamming(4,7) resulting a watermark of length 49 bits.

C. Embedding the metadata

After the watermark is created based on the given metadata, we use a spread spectrum approach [13] for embedding it into the host medical signal. Our technique will embed the same watermark multiple times in a single time-series sequence. A given ECG signal is partitioned into a set of subsequences $\mathcal{S}$. Then in each of these sub-ECGs the watermark is embedded. This distributes the power of the watermark across multiple frequencies of the signal subsequences, making its removal particularly difficult, while at the same time preserving the important data characteristics. In other words, we get a stronger watermark with less power, i.e. less noise introduced in the original ECG by spreading the watermark signal over the whole data.

More specific, given an ECG signal $x = \{x_1, \ldots, x_n\}$, we first select a random starting point t_k using κ as the seed of a pseudo-random number generator. We then split x into $|S| = \lfloor n/m \rfloor$ adjacent subsequences, starting from t_k. However, when we reach to the last point of x, i.e., x_n, we cyclically continue embedding the watermark W from x_1 until the remaining $n - m * s$ points of x. We ignore these last $n - m * s$ remaining points before x_{t_k}. We denote the set of these subsequences with $\mathcal{S}$ and from now on we will call them *characteristic subsequences*. We chose each characteristic subsequence to contain $m = 3 * l$ points such that each subsequence is 3 times longer than the bit-string to be hidden into the data. This simply allocates enough bandwidth in order to embed the watermark in the lowest frequencies of each subsequence, since the length should be at least 2 times the length of the watermark due to the conjugate symmetry of Fourier coefficients. The magnitudes of each subsequence are then updated according to the additive embedding scheme described before.

The embedding process returns the second part of the secret key β to be used during the detection process described later. The vector β is defined as the average values of the various ρ_j of the subsequences in S, only for those j such that $W_j \neq 0$:

$$\beta j(x) = \frac{1}{|S|} \sum_{s \in S} \rho_j(s)$$

Note that the vector β is calculated on the original ECG, i.e. before the watermarking takes place.

Unlike a non-blind watermarking approach, where in order to retrieve the watermark it is necessary to have access to the original data, in our case, we will only need the vector $\theta = [\kappa \; \beta]$. In this sense we avoid revealing the original data to the users, hence avoiding any obvious security risks.

Resilience of the Embedding: Potential transformations in a medical signal include vertical shifts, re-sampling (upsampling or downsampling) and cropping. By construction, our technique is resistant to vertical shifts, which only affect the first frequency component (the DC), where no part of the watermark is embedded. In the experimental section, we also evaluate the resilience of our scheme to other types of attacks, such as noise addition, upsampling and decimation.

D. Error introduced by the watermark.

We measure the amount of noise introduced in a watermarked signal $\widehat{x}$ as the relative error ϵ, w.r.t the original x:

$$\epsilon(x, \widehat{x}) = \frac{\|x - \widehat{x}\|}{\|x\|}$$

where, $\| \cdot \|$ signifies the L_2 norm of a vector.

If we consider a single subsequence s of x, then due to Parseval's theorem [14], and after some algebraic manipulations, it is easy to see that:

$$\begin{aligned}
\|s - \widehat{s}\|^2 &= \|S - \widehat{S}\|^2 = \ldots = \\
&= \|\rho - \widehat{\rho}\|^2 + 2\sum_j \rho_j \widehat{\rho}_j [1 - \cos(\phi_j - \widehat{\phi}_j)] \\
&= \|\rho - \widehat{\rho}\|^2 \qquad\qquad (since\ \phi_j = \widehat{\phi}_j\ ) \\
&= \|\rho - \langle \rho + pW \rangle\|^2 \\
&\leq \|pW\|^2 \ = \ l\ p^2
\end{aligned}$$

The above gives an upper bound to the error introduced in a single subsequence, assuming that $\langle \rho + pW \rangle = (\rho + pW)$. It also shows that an additive watermarking introduces an error which is proportional to the square root of key length and to the watermarking power. To get an upper bound on the error, ϵ_p, for the whole signal x, we apply the previous result for each segment, yielding

$$\epsilon_p = \frac{1}{\|x\|} \sqrt{\sum_{s \in S} lp^2} = p \frac{\sqrt{|S|l}}{\|x\|}.$$

Additionally, we define the *equivalent embedding power*, corresponding to a given maximum error ϵ_p, denoted by p_ϵ:

$$p_\epsilon = \frac{\epsilon_p}{\sqrt{|S|l}} \|x\|$$

Given this direct relationship between p and ϵ, we will use them interchangeably to address the power used in the embedding. We will use p_ϵ to indicate the power p equivalent to a given error ϵ and (similarly for ϵ_p vise-versa).

If $\rho_j + pW_j < 0$ for some j, it means that $W_j < 0$ and that the value of p is too large since it will produce a negative magnitude. In this case, using $\langle \rho_j + pW_j \rangle$ is actually equivalent to using a smaller amount of power for the frequency j. We refer to this implicit power reduction phenomenon as the *power loss*, denoted with $Lossq$. This fact suggests that arbitrary increasing the watermark embedding power p may not necessarily better resilience of the watermark, since not all the bits of the encoded information will be embedded with increased intensity.

E. Metadata retrieval

In order to retrieve the embedded metadata, we essentially need to retrieve the enclosed robust watermark, based on the knowledge of the secret key $\theta = [\kappa\, \beta]$. The process is illustrated in Figure 3. We want to allow only the owners of this secret key to retrieve the sensitive metadata present in the data. Note that the first part of key vector κ is randomly selected from the key space and the second part of key vector β depends only on the data and does not have any correlation with the watermark. By disclosing the secret key θ, not the watermarked data, no information can be inferred about the secret metadata.

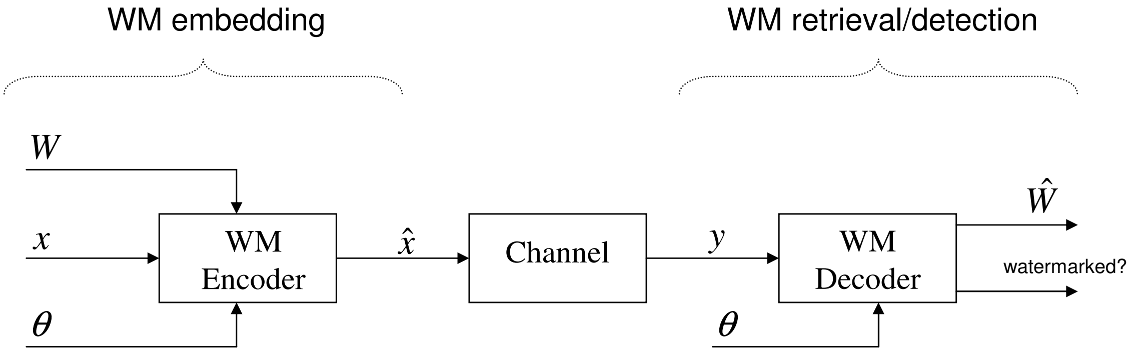


Fig. 3. Illustration of the watermark embedding and detection/retrieval process.

For retrieving the private metadata, we reverse the watermarking process by comparing the value of β that we have from the original ECG and the new value β^y that we calculate from the received ECG signal y. The received signal y is equal to the watermarked data $\widehat{x}$ if there is no distortion (attack) on the signal.

Given a received (watermarked) signal y, we split y into a new set of characteristic subsequences S_y, exactly as done during the watermark embedding process. The metadata are retrieved as follows:

Definition 2 (Metadata Retrieval): Let $\widehat{x}$ and y be watermarked and received signals, respectively. The characteristic subsequences S_y is the set derived from the received signal y, which is equal to $\widehat{S}$ if there is no distortion on the watermarked signal $\widehat{x}$. Let the calculated statistics from the received signal y be β^y, then we define the binary vector Z as

$$Z_i = \begin{cases} 1 & \text{if} \quad \beta_i^y - \beta_i \geq \tau \\ 0 & \text{if} \quad \beta_i^y - \beta_i < \tau \end{cases}$$

where the threshold is selected to control the trade-off between false alarm (FA) and false rejection (FR) rate. Then, the received $B(\text{SSN})$ is given by

$$R = H_{7,4}^{-1}\langle Z \rangle.$$

where R is equal to $B(\text{SSN})$ if there is no error in retrieval.

If $\beta_j^y - \beta_j \geq \tau$ we have a hint that the j-th element of the embedded watermark is equal to 0 ($W_j = 0$), and symmetrically equal to 1 if $\beta_j^y - \beta_j < \tau$. In order to get the actual data, we must apply the Hamming decoder $H_{7,4}^{-1}$. Using decoder we retrieve two pieces of information. First we infer whether there has been some error in the retrieval of R, and secondly we can try to remove such an error.

Example: Suppose that we are embedding an 8 digit SSN of a patient as the secret metadata. We first convert 8 digit SSN into a 27-bit long binary stream. This conversion can be as simple as using the binary representation of each digit. We next input this binary representation to a cryptographically safe hash function (with κ as the secret key) to get, again, 27-bit long random sequence. Applying Hamming(7,4) for each 4-bit blocks of this data would yield a watermark signal of 49-bit long, i.e., $\lceil 27/4 \rceil * 7 = 49$ where $\lceil \rceil$ represents rounding towards the upper integer. Given an ECG signal x, this W would be embedded for each segment of size 3×49. After decoding the watermarked signal, we get Z. If there is no attack on x, than it is easy to see that Z should be equal to W, since $\beta^y - \beta = \widehat{\beta} - \beta = pW$. In the presence of an attack one can measure the goodness of the watermarking as:

$$\text{Goodness} = 1 - \frac{\sum Z \text{ XOR } W}{49}$$

i.e. the percentage of bits correctly retrieved.

F. Watermark Detection

Given the secret key θ, one can also simply detect the presence of the watermark without retrieving back the embedded metadata. This is achieved using a generalized correlation detector which is given in the following definition:

Definition 3 (Watermark Detection): Let x, $\widehat{x}$ and y be the original, watermarked and received signals, respectively. The characteristic subsequences S_y is the set derived from the received signal y and equal to $\widehat{S}$ if there is no distortion on the watermarked signal $\widehat{x}$. Let the calculated statistics from the received signal y be β^y, then we define the generalized correlation detector as

$$\frac{\left\langle \beta^y - \beta, \widehat{\beta} - \beta \right\rangle}{\|\widehat{\beta} - \beta\|^2} \begin{array}{ccc} > & \tau & \text{watermarked} \\ \leq & \tau & \text{not-watermarked} \end{array} \qquad (1)$$

where the threshold τ is selected based on the desired false acceptance and false rejection rate, and $\langle x, y \rangle = \sum_i x_i y_i$.

The above correlation detector is decision-theoretic optimal when the disturbance on $\widehat{x}$ is white Gaussian noise [15]. However, in case of non-Gaussian disturbances, we also introduce following updated correlation detectors which work directly on the received bits instead of β values:

$$\frac{\langle Z, W \rangle}{\|W\|^2} \begin{array}{ccc} > & \tau & \text{watermarked} \\ \leq & \tau & \text{not-watermarked} \end{array} \qquad (2)$$

and

$$\frac{\langle R, B(\text{SSN}) \rangle}{\|B(\text{SSN})\|^2} \begin{array}{ccc} > & \tau & \text{watermarked} \\ \leq & \tau & \text{not-watermarked} \end{array} \qquad (3)$$

In the experimental section, we include detailed experiments regarding the performance of the above three watermark detectors.

IV. Fragile Watermark

After the robust watermark which encloses the private metadata is embedded in the ECG signal, a fragile watermark will be added on top of the resulting signal. The fragile watermark can be used to efficiently detect subsequent alterations to a marked data. Although, the robust watermark is designed to be resilient against most of the benign signal processing operations (such as compression, cropping, decimation) and/or against malicious attacks that intentionally attempt to remove the underlying watermark, fragile watermarks are designed to detect (with high probability) even the slightest changes on the underlying watermarked data. By definition, the fragile watermark should easily reveal that the data is modified or tampered. Although conceptually different, the embedding and detection of fragile watermarks is similar to that of robust watermarking framework. Given a key, i.e., SSN of a patient for our application, a fragile watermark will be generated and then embedded to the underlying ECG signal. Upon reception of the watermarked ECG signal, the recipient subsequently uses a detector to authenticate the underlying signal. This detector may use the underlying key and a side-information generated from the original data (whose generation mechanism would be clear later on) in order to determine the authenticity of the received signal. We refrain from revealing the original signal to the users and restrict their access to only side-information due to obvious security considerations. The side-information is generated using randomization in order to leak limited information about the original data to the users. We stress on the randomized aspect of our algorithms, since a randomization approach will protect the watermark against most of the intentional attackers trying to estimate the watermark.

A. *Fragile Watermark Embedding*

For our particular application, we desire our fragile watermarking to have the following properties:

1) The embedded watermark should not interfere with the underlying usage of the signal. This requirement reduces the candidate algorithms that one can use on the fragile watermark, in order to induce only minimal effects on the underlying ECG signal.
2) The fragile watermarking should be able to detect the presence of tampering on the medical signal.
3) The fragile watermarking should give localized information about tampering. To satisfy this, the fragile watermark needs to be localized. The candidate fragile watermark should also be able to quantify the nature of the underlying alterations or attacks on the corresponding signal. For some applications this property is essential, since most of benign signal processing operations such as compression or change of axis by DC addition/subtraction will destroy the fragile watermark, however, the underlying signal is still useful for all practical purposes. Hence, the fragile watermark should quantify the underlying cause of the alteration as much as possible in order to make the final judgment on the usability of the tampered signal.

Since our first motivation is to detect any alteration on the underlying ECG signal and we desire to have minimal effect on the underlying signal, we embed the watermark in the spatial domain on the *least-significant-bits* (LSB's) of the ECG signal. This type of algorithms that alter the LSB's are extremely effective for detection of random perturbations, but in their most basic form [7] are very susceptible to malicious attacks. One can easily change the underlying watermarked signal (in the extreme case completely replace with another signal) without touching the LSB's. In the literature, there are many different variations of the basic approach to reduce this kind of vulnerability to malicious attacks by including context information into the watermark [16]–[18] . In this paper, we require, the embedded watermark signal to be both context and data dependent in a randomized manner in order to avoid any possibility of an attacker to either replace the watermark partially or completely, or alter the watermarked signal. The fragile watermark embedded in the LSB's depends on randomly generated semi-global data statistics, which we believe would capture the essential features of the underlying signal [19]. We extensively use randomization in order to eliminate the possibility for an attacker to retrieve any information about the original key. Since, an attacker which has access to the original key could use this key for watermarking arbitrary data.

Next, we provide the basic fragile watermarking algorithm and the motivation of each step. The complete description of the embedding and detecting of the algorithm are given in Figure 5 and in Figure 7, respectively.

Embedding Algorithm: Given an ECG signal $x = \{x_1, \ldots x_n\}$, we first separate the underlying signal into separate blocks based on heart-beats, i.e., we use each heart-beat duration as a segment, where x^i is the portion of the ECG signal corresponding to the ith heart beat. To achieve a beat to beat signal separation we utilize an energy based filter, since the ECG signal should exhibit higher energy at the frequency indicated by the heart beat. Note, that the heart beat separation does not have to be exact, since this block processing is merely a way of providing broad localization information upon the fragile watermark. Given the fact the we work on ECG, we can exploit their inherent pattern regularity in establishing

Subsequently, we remove the LSB from each x_k^i to get $\tilde{x}_k^i$, i.e., $\tilde{x}$ is the ECG signal where all LSB's are set to zero. We use κ as the seed for a pseudo random number generator to generate p randomly located intervals with length w, where $\{t_1^i, \ldots, t_p^i\}$ are the randomly selected starting points for each interval, in Figure 4.

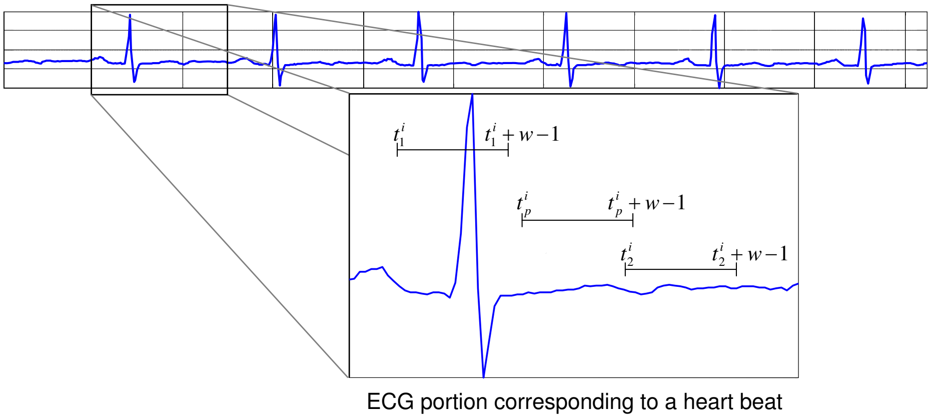


Fig. 4. Localization of the fragile watermark is achieved through data 'blocking' into heart-beats. Subsequent selection of randomly generated windows within the heart-beat for embedding the fragile watermark.

Naturally each t_j^i is selected to avoid any interference with the next segment, i.e., $t_j^i + w - 1$ should be less than the starting point of the next segment. The length of these windows w is a design parameter. Obviously, there is a trade off in selecting w, since a large w would capture the essential (or global) characteristics of the signal better, but a small w would capture the local characteristics of the signal better [20]. Given a randomly selected location and a window of length w, we generate semi-global statistics from this portion of the data. These statistics can also have random components in their generation, however in this work we do not use any randomization, except their locations. Our algorithms are generic such that this kind of alterations can readily be incorporated. The windows can be overlapping so that we avoid constraining the selection of locations to reveal limited information to an attacker. These local and randomly generated features are essential and would be called *hash values*. We use these hash values (after appropriate quantization) and the patient metadata, as the seed of a random number generator to generate the final fragile watermark which is comprised of zeros and ones of length equal to heart-beat duration. The resulting fragile watermark is embedded to the LSB's of the corresponding heart-beat. We repeat the same process for each heart-beat to create the watermarked ECG signal.

We generate several different statistics (or hash values) per window to capture different features of the data in that window. Per window, we generate three different hash values $g_{1,j}^i, g_{2,j}^i, g_{3,j}^i$, $j = \{1, \ldots, p\}$, by calculating: the power of the corresponding signal filtered by a low pass filter, a band pass filter and a high pass filter as seen in Figure 6. Hence for each heart-beat segment

$$\begin{aligned} g_{1,j}^i &= T_1(\{\tilde{x}_{t_j}^i, \ldots, \tilde{x}_{t_j+w-1}^i\}),\ j \in \{1, \ldots, p\} \\ g_{2,j}^i &= T_2(\{\tilde{x}_{t_j}^i, \ldots, \tilde{x}_{t_j+w-1}^i\}),\ j \in \{1, \ldots, p\} \\ g_{3,j}^i &= T_3(\{\tilde{x}_{t_j}^i, \ldots, \tilde{x}_{t_j+w-1}^i\}),\ j \in \{1, \ldots, p\} \end{aligned}$$

where $T_1(x)$ ($T_2(x)$,$T_3(x)$) represents the composite operation of first lowpassing (bandpassing, highpassing) the signal x and then calculating the power of lowpass ((bandpass, highpass) filtered signal. We collect all hash values corresponding to all segments and windows in $g = \{g_{l,j}^i\}$. Apparently, these three different hash values would capture the different features of the data. For example a local shift of the heart-beat data, i.e., a DC addition or subtraction, will not effect the hash values generated by the high pass or band-pass filters, hence revealing and localizing the corresponding tampering. The amount of tampering could also be determined as the amount

```
Embedding:

Step 1: Let $x \in \mathbb{R}^n$ be an ECG signal of size $n \times 1$.
Step 2: For each sample of $x$, remove the LSB to get $\tilde{x}$.
Step 3: Split $\tilde{x}$ into disjoint segments $\tilde{x}^i$ where each $\tilde{x}_i$ corresponds to a single heart-beat and $\tilde{x}$ is the union of $\tilde{x}^i$ $i = \in \{1, \ldots, N\}$
Step 4: For each $i = \{1, \ldots, N\}$
Step 4.1: Given $\tilde{x}^i$, generate $p$ possibly overlapping intervals (each with size $w \times 1$) with time stamps $\{t_1^i, \ldots, t_p^i\}$,
Step 4.2: For each interval generate three semi-global features: $g_{1,i} = T_1(\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\})$, $g_{2,i} = T_2(\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\})$ and $g_{3,i} = T_3(\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\})$ where
    $T_1(.)$ is the power of low-passed filtered $\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\}$ with pass band $[0, \pi/3]$,
    $T_2(.)$ is the power of band-passed filtered $\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\}$ with pass band $[\pi/3, 2\pi/3]$,
    $T_3(.)$ is the power of high-passed filtered $\{\tilde{x}^i_{t_j}, \ldots, \tilde{x}^i_{t_j+w-1}\}$ with pass band $[2\pi/3, \pi]$
Step 4.3: Construct $\kappa_i$ by appending $\kappa$ with appropriately quantized version of $g^i_{l,j}$, $\kappa_i = \text{CONCAT}(\kappa\,\{g^i_{l,j}\})$.
Step 4.4: Generate a random vector of the same size of $x^i$ comprised of zeros and ones, $W^i_{fra}$ using $\kappa_i$ as the seed of a random number generator.
Step 4.5: Replace LSB's of $\tilde{x}^i$ with this random vector.
```

Fig. 5. Embedding of fragile watermark.

of change in the corresponding hash values. Even a local tempering could be pinpointed since we use several overlapping windows for each heart-beat segment. Although we use simple outputs of straightforward DSP filters, more sophisticated filters or algorithms that are tuned for a particular application or a signal database can be easily introduced in the algorithm. Each new addition will introduce further localization or capture different features of the data. After collecting the hash values for each interval for each segment of a heart-beat, we append the patient metadata with appropriately quantized values of these hash values as the seed of a random number generator:

$$\kappa_i = \text{CONCAT}(\kappa\,\{g^i_{l,j}\})$$

to generate the fragile watermark for this segment W^i_{fra}. The fragile watermark, W^i_{fra}, is the same length as the ith segment and comprised solely of zeros and ones. This randomly generated WM will be the LSB's of this particular segment. We replicate this procedure for each heart-beat segment to get the final fragile watermarked signal.

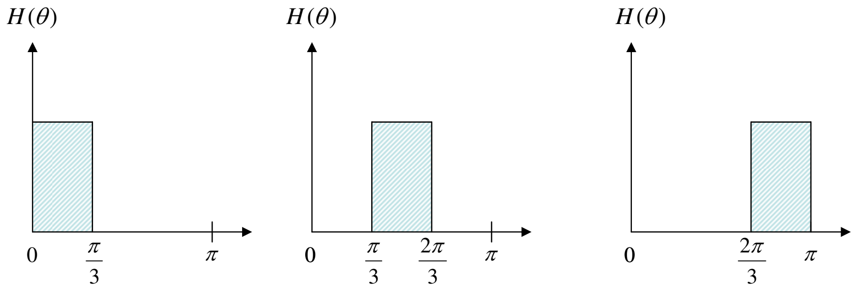


Fig. 6. Filters used for extracting the various window statistics

B. Fragile Watermark Detection

For detecting of the fragile watermark, we follow similar steps as the embedding. Given a watermarked ECG signal $\widehat{x}$ and hash values g of the original data as the side information, we first remove and store the LSB's for each $\widehat{x}_k$. The hash values are generated for each heart-beat segment using the same random number generator with

κ as the seed. After getting the time stamps, $\{t^i_1, \dots, t^i_p\}$, we calculate the following hash values,

$$\widehat{g}^i_{1,j} = T_1(\{\widehat{x}^i_{t_j}, \dots, \widehat{x}^i_{t_j+w-1}\}),\ j \in \{1, \dots, p\}$$
$$\widehat{g}^i_{2,j} = T_2(\{\widehat{x}^i_{t_j}, \dots, \widehat{x}^i_{t_j+w-1}\}),\ j \in \{1, \dots, p\}$$
$$\widehat{g}^i_{3,j} = T_3(\{\widehat{x}^i_{t_j}, \dots, \widehat{x}^i_{t_j+w-1}\}),\ j \in \{1, \dots, p\}$$

where with an abuse of notation we used $\widehat{x}$ to represent the watermarked signal with LSB's removed. We then generate the final random signal using concatenated κ and the quantized hash values as the seed of a random number generator,

$$\widehat{\kappa}_i = \text{CONCAT}(\kappa\ \{\widehat{g}^i_{l,j}\}).$$

We next compare this random sequence $\widehat{W}^i_{fra}$ with the stored LSB's to reveal any alteration. If these two sequences differ, than we announce a possible tampering. One can check to see whether this tampering can be localized through the use of the hash values by calculating

$$\text{Tampering}(i, l, j) = \frac{|\widehat{g}^i_{l,j} - g^i_{l,j}|}{|g^i_{l,j}|}, j = \{1, \dots, p\}, l = \{1, 2, 3\}$$

for each segment i. The absolute relative change in the hash values would reveal the possible tampering in the respective region. Although most of the tampering should be localized by the hash values, small changes on the data (intentional or not intentional) may not be caught by the hash values (although they will be caught by our fragile watermarking).

Detection:

Step 1: Let $\widehat{x} \in \mathbb{R}^n$ be a watermarked ECG signal of size $n \times 1$
and $g \in R^m$ be a vector of side informations.
Step 2: For each sample of $\widehat{x}$, remove the LSB and store it.
Step 3: Generate the side information sequence $\widehat{g}$ following the exact same lines of WM embedding
Step 4: Generate fragile WM using $\widehat{g}$ and SSN and compare it with the stored LSB's
Step 5: If they are different than the signal is tampered
Step 6: If tampering is present, check Tampering(i, l, j).

Fig. 7. Detection of fragile watermark

V. Experiments

We evaluate empirically the robustness of the proposed metadata embedding technique. We demonstrate that the methods introduce only imperceptible variations that do not distort important ECG features and, as as sequence do not alter the diagnosis of a cardiologist or physician. Additionally, we show that the embedding techniques are able to withstand various attacks. We utilize ECG signals extracted from the MIT arrhythmia database [21] which include normal signals as well as arrhythmic signals annotated as malignant ventricular or supra ventricular arrhythmias. The datasets used are available by emailing the contact author.

A. Determining the embedding power

In order to determine the proper embedding power of the metadata, we solicited the expertise of co-author Helga van Herle, who is a cardiologist. She examined a random subset of over 100 normal and abnormal ECG's, on which various random SSN's were embedded using increasing embedding powers on the robust watermark. A subset of such ECG's is demonstrated in Fig. 8. The result of this user study with a topic expert, indicated that for SNR=20 the diagnosis might change for certain ECG's, because of various distortions that were introduced near the P-wave region. However, for SNR's of 30 or 40 the diagnosis would not be affected for any of the examined ECG's. Therefore, for our experiments we use embedding powers p that would lead to $SNR > 30$ for each ECG signal.

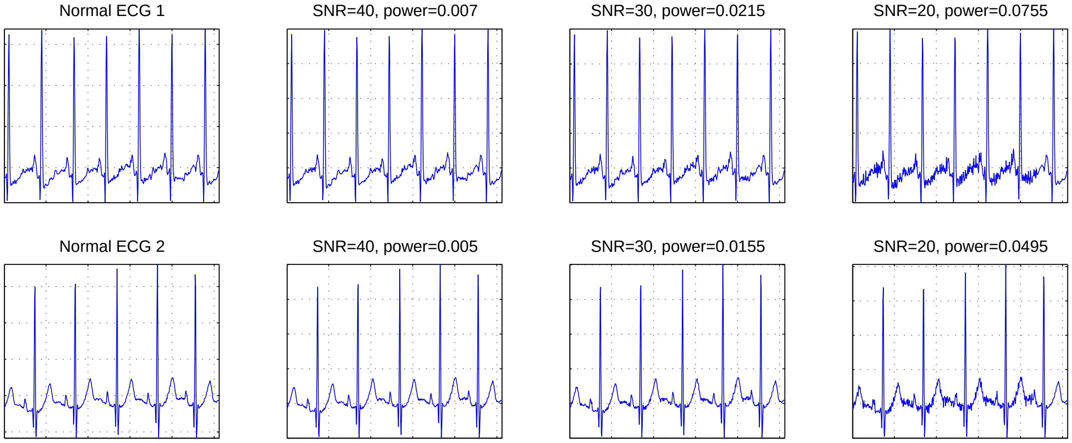


Fig. 8. Distortion of ECGs for various embedding powers and the resulting signal-to-noise ratio

B. Class-Label Preservation

One the major features that a cardiologist examines on ECG data is the presence of arrhythmias which can be an indication of various heart pathologies. Atrial fibrillation is the most common cardiac arrhythmia [22] which can be a strong indication for the possibility of a stroke. Spectral [23] and bispectral [24] techniques have reported success in detecting arrhythmias in medical data. Here, we utilize the spectral distance measure of [23] for quantifying the similarity between 10 normal and 10 arrhythmic ECG's into which we have embedded random SSN's. After the pairwise distances between the 20 ECG's are evaluated we create the resulting dendrogram, which is illustrated in Fig. 9. With the darker color are shown the abnormal ones and with lighter color the normal ECG's. One can observe that even on the ECG's with the embedded metadata there is a clear separation between the two classes of data. Similar results we obtain for the remaining portion of ECG datasets. This example, serves as a simple demonstration that the metadata embedding does not distort significant ECG features, which are important for a proper medical diagnosis.

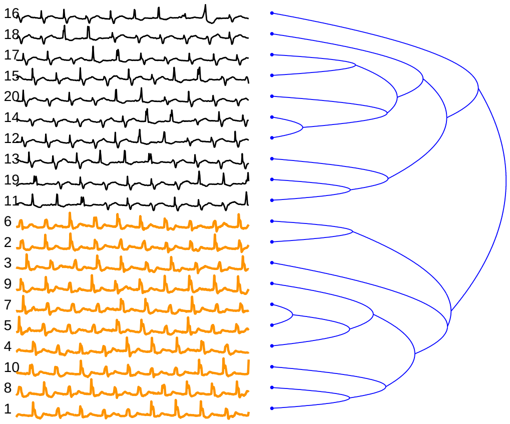


Fig. 9. Dendrogram of ECG's with embedded metadata. We observe that class labels are not distorted. One can still discriminate clearly between arrhythmic (dark color) and normal (light color) ECGs.

C. Resilience Under Attacks for Robust Watermarking

We test the efficacy of metadata retrieval and watermark detection under various data transformations (or potential attacks). In this section we quantify the performance of the robust watermark that carries the metadata, but both robust and fragile watermarks are embedded on the ECG's. The fragile watermark can detect the presence and location of the transformation, and its efficiency we quantify in the upcoming section. For the robust watermark, we examine effect of the following transformations:

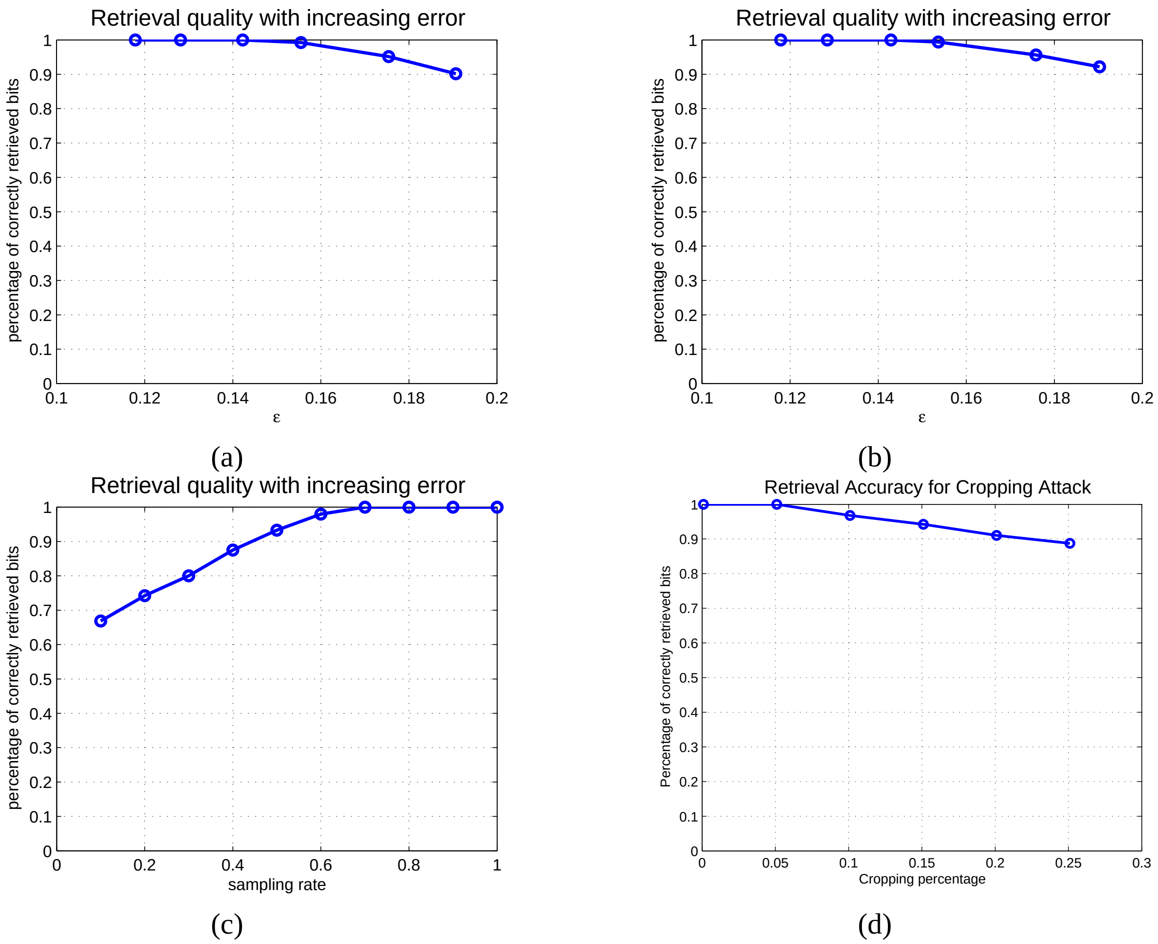


Fig. 10. (a) Noise addition in space domain, (b) Noise addition in Frequency Domain (c) ECG downsampling (d) ECG cropping

■ **Noise addition in the space domain:** This is a critical attack because it can potentially destroy the embedded metadata. We first test metadata retrieval when we translate randomly the baseline of the ECG signal (which doesn't destroy the ECG usability) and we add up to 20% relative noise on the original ECG signal. In Figure V-C(a) we plot metadata retrieval (as the percentage of correctly recovered bits) versus noise level. From the figure, we observe that up to 14% of distortion (which would anyway destroy the ECG usability) one can retrieve the whole amount of the embedded metadata. This is possible due to the redundancy schemes that we employ in the encoding of the hidden metadata.

■ **Noise addition in the frequency domain:** An adversary may also add Gaussian noise in the frequency domain, which is where the metadata are embedded. The results for this attack are depicted in Fig. V-C(b). We observe similar results for this attack as well, which again validate the robustness of our approach.

■ **Decimation:** On this attack an ECG is represented by smaller set of points that best approximate the original ECG signal. A shorter sequence is obtained by sampling equidistant points from the spline associated with the original ECG sequence. Decimation is a significant attack, because even though it does not change significantly the shape of the ECG signal, it allows the adversary to generate a new sequence which has no points in common to the original sequence. In our tests (see Figure V-C(c)), even when the ECG signals are represented using only 70% of the original number of points, all of the metadata bits are retrieved correctly.

■ **Cropping:** This is another severe attack on ECG signals. In cropping attack, the ECG signal is shortened by a fixed amount by eliminating a part of the ECG signal. Since, the size of the cropped ECG signal is shorter than the expected length, we perform a local search based on the correlation between the recieved β^y and original β over a window. The point where this correlation is maximized is used for watermark retrieval and detection. As seen in

Figure V-C(d), we plot the watermark retrieval with respect to amount of cropping performed on the ECG signal. We observe that the retrieval performance gracefully degrades as the cropping amount increases. We observe no distortion up to 5% percent and minimal distortion up to 20% croppings.

Therefore, the above experiments have shown that the effective coding scheme which also carries redundancy, can effectively retrieve the embedded metadata even under the presence of significant transformations. Additionally, a malicious adversary would have to destroy the usability of the signal (distort the shape significantly) in an effort to erase the hidden data.

D. Robust Watermark Detection

In addition to retrieving the metadata, one can also simply detect the presence of the watermark using one of the three watermark detectors presented in section III-F. We evaluate the performance of these detectors under the same data transformations as in the previous experiments, using false acceptance/false rejection curves (FA-FR curves).

■ **Noise addition in the space domain:** Here, the ECG signals are normalized to have maximum amplitude equal to 1 with zero DC and the average power of an ECG signal is 0.05. The attack consists of additive Gaussian noise with standard deviations: $\sigma = 0.001$, $\sigma = 0.01$, $\sigma = 0.05$, $\sigma = 0.1$. Hence some of these attacks can be considered as severe. In Figure V-D(a), we plot the FA-FR curves for four different noise powers for the correlation detector introduced in Equation (1). As seen, for noise powers 0.001 and 0.01 the FA-FR curves are on the x-y axes, i.e., the algorithm perfectly separates detection regions (hence there are no errors in detection). As expected, the detection performance gracefully degrades as the noise power increases. Similar performance results are observed in the other algorithms introduced in Equation (2) and (3), respectively. To compare the performance of these three different detectors, we also plot corresponding FA-FR curves for $\sigma = 0.05$. We observe that for additive Gaussian noise (even though the additive noise is in space domain) the first watermark detection algorithm based on correlation of β's outperforms the other two.

■ **Noise addition in the frequency domain:** The FA-FR curves for frequency domain attacks are presented in Figure V-D, using the same four additive noise levels as before. The results directly follows the results of noise addition in space domain. These results further corroborate the robustness of our watermarking algorithm.

■ **Decimation attack:** We next present the FA-FR curves for decimation attack in Figure V-D, for decimation up to 80%. We observe that our watermark detection algorithm is effective up to 75% decimation, which is a quite severe distortion. We attribute this robustness due to using lower part of the frequency spectrum for mark embedding, since in decimation type of attacks, the higher frequencies are more effected due to lowpass filtering to avoid aliasing.

■ **Cropping attack:** We finally present the FA-FR curves for cropping attacks in Figure V-D. We try several different amount of croppings from 10% up to 50%. We observe the same robustness properties for this attack also.

E. Fragile Watermarking

This section studies the performance of the fragile watermark and in specific the behavior of hash values under several different attacks on the ECG signals. For fragile watermarking, we choose a window size of 50 samples and for each region we collect hash values from 4 different subintervals. We observe that the hash values are not that sensitive to window length but 50 samples provide a fair trade-off between localized information and capturing of semi-global robust statistics [20]. For generation of hash values, we use 8th order low pass, band pass and high pass filters where each filter is designed using a Butterworth algorithm. We choose an 8th order filter to decrease the effect of initial transients due to the use 50 sample sub-intervals. We observe that the effect of this initial transients are unavoidable but acceptable.

As the first attack, we try a DC shift on the ECG signals. Naturally, a DC shift does not alter the usability of ECG signal, unless there is also additional clipping involved. In Figure 15, we plot the histogram of percentage change in three different hash values for a DC shift of 1. These are hash values corresponding to low pass filter, high pass filter and bandpass filter outputs. Naturally for a DC shift in space domain, high pass and low pass features are not effected. The obvious change in hash values corresponding to low pass filter is due to the impulse like change in frequency domain at frequency 0 due to DC addition. We clearly distinguish the particular change in DC value due to hash values generated by low pass filter which is the main motivation for hash usage.

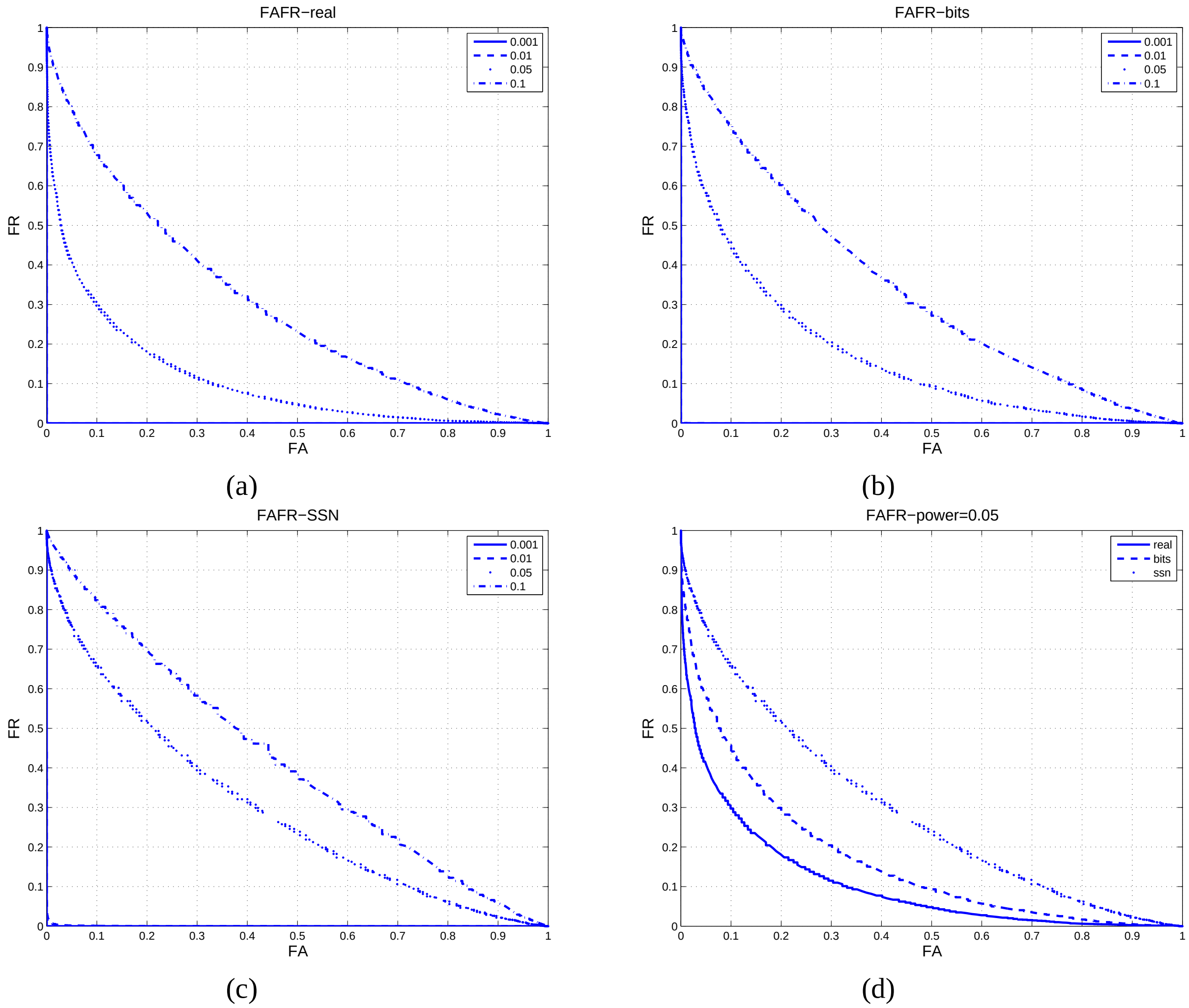


Fig. 11. FA-FR curves for three different watermark detectors for four different noise levels added in space domain. (a) Watermark detector from Equation (1) (b) Watermark detector from Equation (2) (c) Watermark detector from Equation (3) (d) FA-FR for all watermark detection algorithms together for $\sigma = 0.05$..

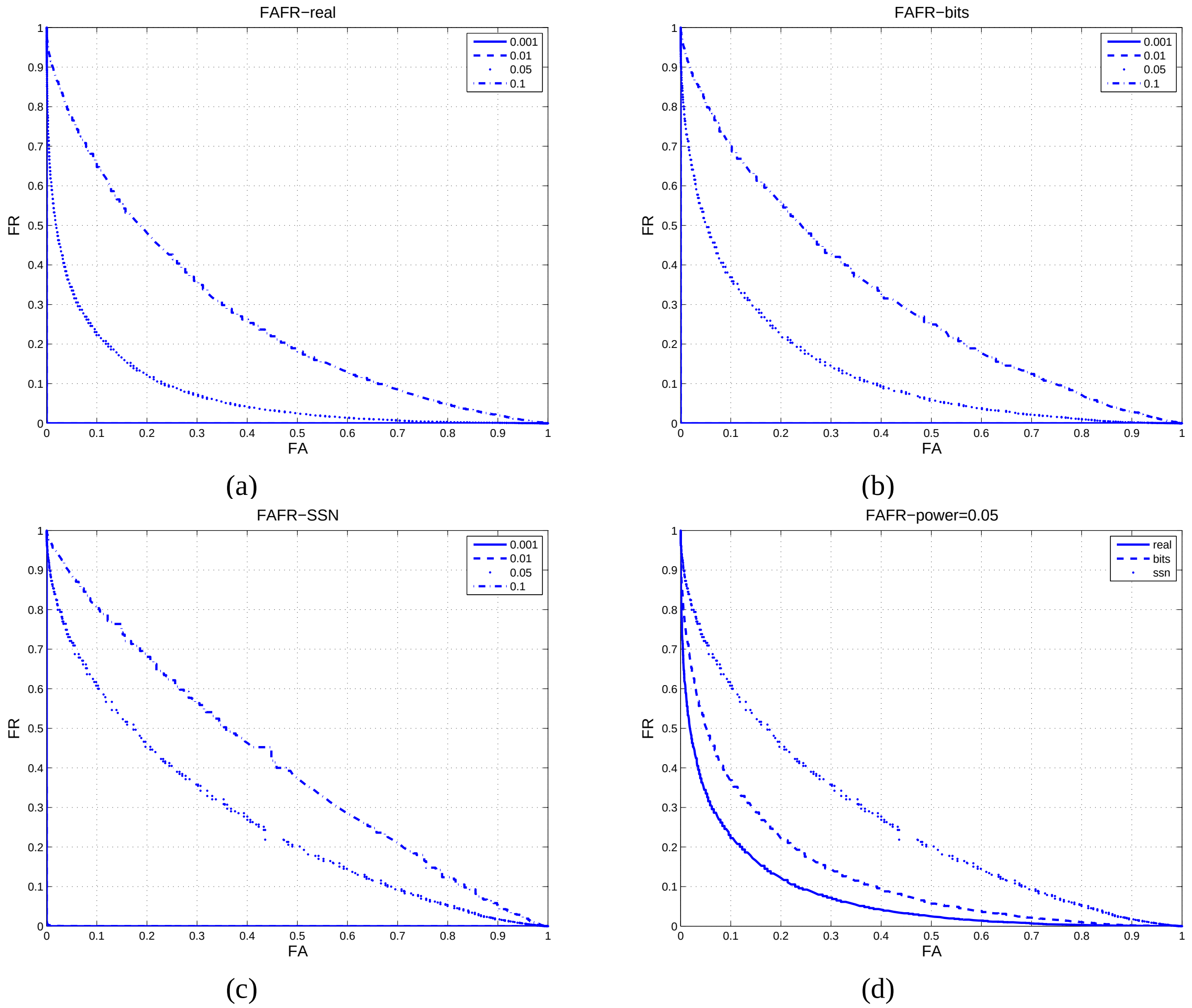
We next try additive Gaussian noise (as done in the robust watermarking experiments) since this kind of attack (or disturbance) is common due to both intentional or unintentional changes, e.g., data compression. In Figure 16, we plot histogram of percentage changes in hash values for two different noise levels: $\sigma_n = 0.1$ representing a severe attack and $\sigma_n = 0.001$ reprenting a less-severe attack. We observe that the hash values corresponding to all three filters are effected by this attack. The changes in hash values reflect the degree of the attack since the percentage change in the severe attack is an order of magnitude larger than the less-severe case. A change in all hash values shows a broadband attack on ECG signals since all the frequency components are effected.

As the next set of experiments, we repeat the previous attack in the frequency domain with the same noise levels and plot the results in Figure 17. We observe the same kind of behavior illustrating the effectiveness of hash values to assess the severity of attacks on ECG signals.

For decimation attacks we plot histogram of percentage changes in hash values corresponding to decimation amounts 50% and 75%, in Figure 18. Naturally, since the decimation of a signal effects mainly the higher frequencies due to lowpass filtering to avoid aliasing (and if lowpass filtering is not present, due to aliasing), we observe large changes in hash values generated from bandpass and highpass filters. The hash values generated from lowpass filters are relatively unchanged.

We next present the results for cropping experiments and plot the changes in hash values in Figure 19 for cropping amounts 20% and 40% providing the sensitivity of hash values under cropping.

We point out that in all cases, the fragile watermarking is destroyed, hence showing presence of an alteration on



(a) (b) (c) (d)

Fig. 12. FA-FR curves for three different watermark detectors for four different noise levels added in frequency domain. (a) Watermark detector from Equation (1) (b) Watermark detector from Equation (2) (c) Watermark detector from Equation (3) (d) FA-FR for all watermark detection algorithms together for $\sigma = 0.05$.

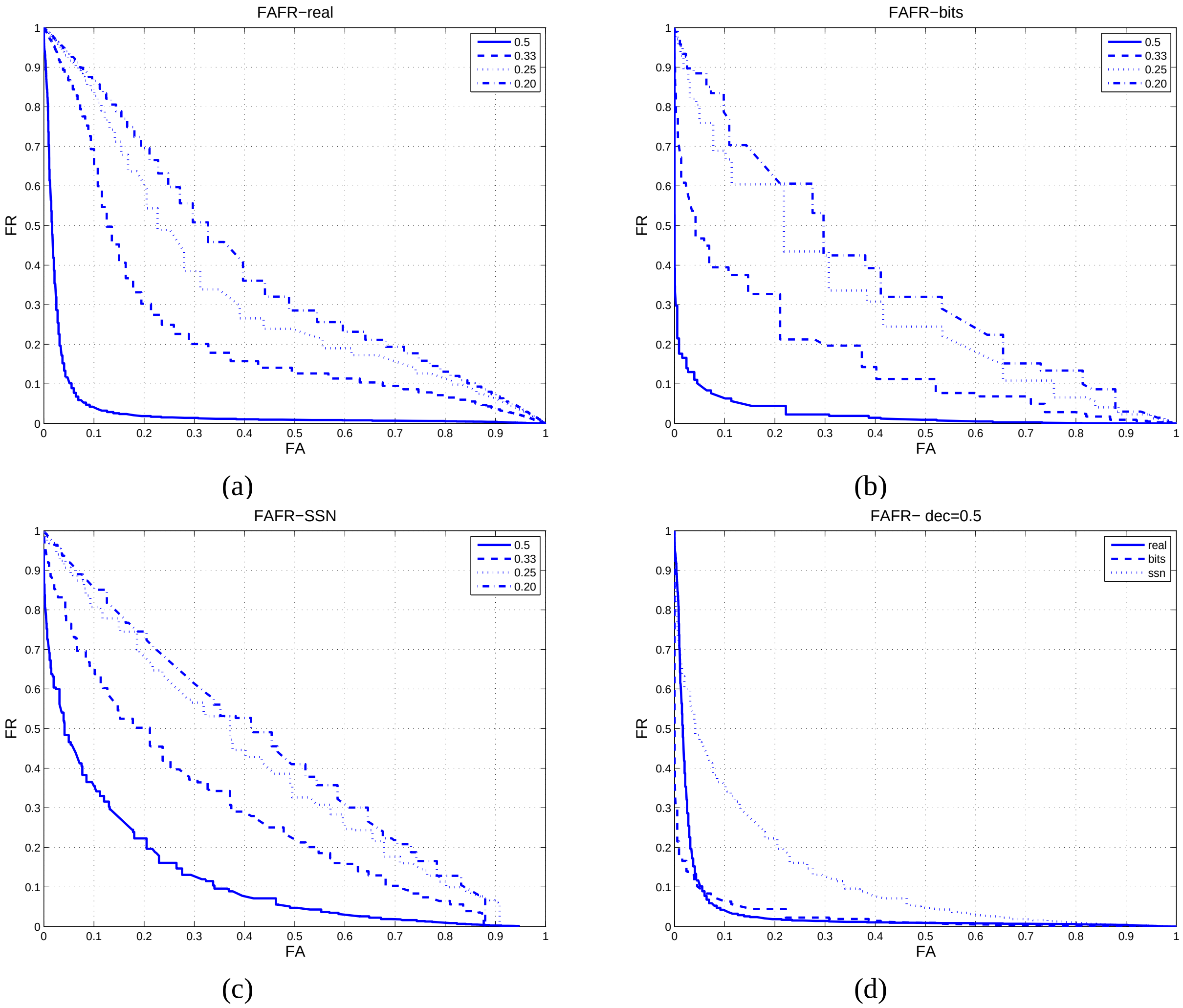
ECG signals. In all cases the hash values give relavant information about the nature of the underlying attack.

VI. Conclusion

In this paper we introduced the topic metadata fusion within medical time-series data. To our knowledge, this is the first work that examined this problem. We show that this embedding does not distort the visual appearance of the medical signal and it also does not induce any changes in the diagnosis. On a technical level we offer the following contributions:

- We effectively combine watermarking and channel coding schemes for providing the sufficient resilience on the metadata retrieval
- We augment the above robust technique with localized fragile watermarks that can pinpoint the type and location of a potential tampering
- Finally, we evaluate the robustness of the proposed schemes under various transformations and attacks using publicly available ECG datasets.

Even though we presented our techniques on statically stored ECG signals, due to the inherent windowing of our technique, our method is very easily extendible on streaming medical data. Such types of data, are even more prevalent nowadays, with the advent of economic sensor devices that can transmit various measurements of interest. Streaming medical measurements are, for example, transmitted during aeronautical exercises for measuring the stress level of a pilot or an astronaut. Also, telemedical applications are not uncommon for patients that need



(a)

(b)

(c)

(d)

Fig. 13. FA-FR curves for three different watermark detectors for four different decimation amounts: 50%, 66%, 75% and 80%. (a) Watermark detector from Equation (1) (b) Watermark detector from Equation (2) (c) Watermark detector from Equation (3) (d) FA-FR for all watermark detection algorithms together for 50% decimation.

continual monitoring but are not required to reside in a hospital. The various methodologies proposed in this work, can function as an additional authentication step, regarding the originality of the transmitted streaming medical measurements.

REFERENCES

[1] Paulo Pinheiro da Silva, Deborah L. McGuinness, and Rob McCool, "Knowledge Provenance Infrastructure," in *Data Engineering Bulletin Vol.26 No.4, pages 26-32,*, 2003.

[2] I. J. Cox, J. Kilian, F. T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," in *IEEE Trans. on Image Processing*, 1997, pp. 1673–1687.

[3] M. K. Mihcak, R. Venkatesan, and T. Liu, "Watermarking via optimization algorithms for quantizing randomized semi-global image statistics.," in *ACM Journal on Multimedia Systems, Volume 11*, 2005, pp. 185–200.

[4] M. Engin, O. Cidam, and E.Z. Engin, "Wavelet Transformation Based Watermarking Technique for Human Electrocardiogram," in *Journal of Medical Systems, Vol. 29, No. 6: 589-594*, 2005.

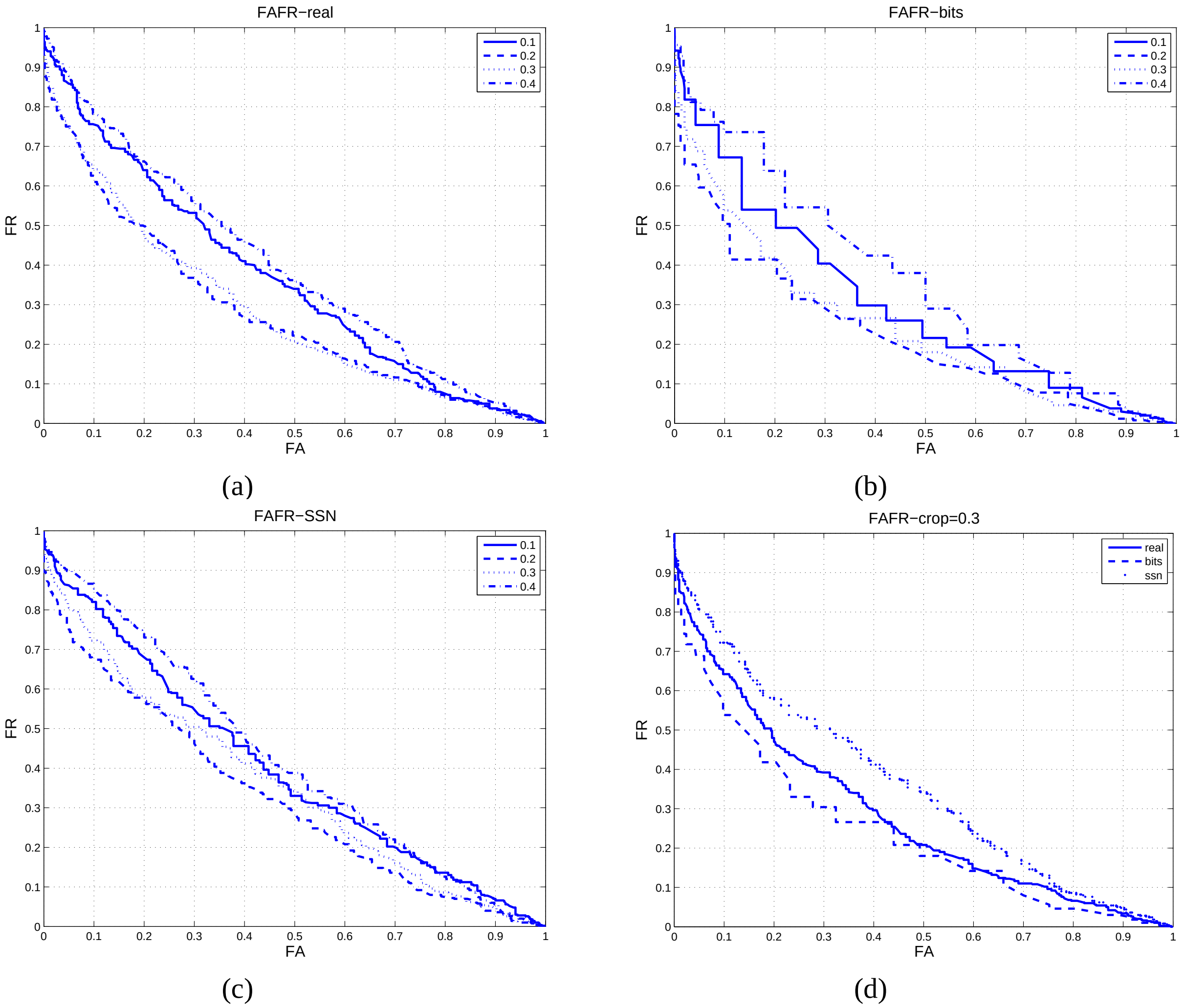
[5] Watermarking Medical Signals for Telemedicine, "Xuan kong," in *IEEE Transactions On Information Technology in Biomedicine, Vol. 5, No. 3*, 2001.

[6] Radu Sion, Mikhail J. Atallah, and Sunil Prabhakar, "Rights Protection for Discrete Numeric Streams," in *IEEE Trans. Knowl. Data Eng. 18(5): 699-714*, 2006.

[7] Rashesh Agrawal and Jerry Kiernan, "Watermarking Relational Databases," in *Proc. of VLDB*, 2002.

[8] R. Sion, M. Atallah, and S. Prabhakar, "Rights Protection for Relational Data," in *Proc. SIGMOD 2003*, 2003.

[9] Elisa Bertino, Beng-Chin Ooi, Yanjiang Yang, and Robert H. Deng, "Privacy and Ownership Preserving of Outsourced Medical Data," in *Proc. of ICDE*, 2005, pp. 521–532.



(a)

(b)

(c)

(d)

Fig. 14. FA-FR curves for three different watermark detectors for four different cropping amounts: 10%, 20%, 30% and 40% (a) Watermark detector from Equation (1) (b) Watermark detector from Equation (2) (c) Watermark detector from Equation (3) (d) FA-FR for all watermark detection algorithms together for 30% cropping.

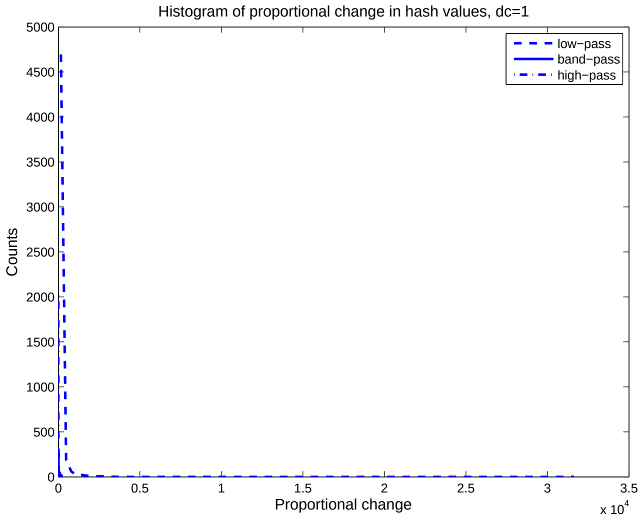
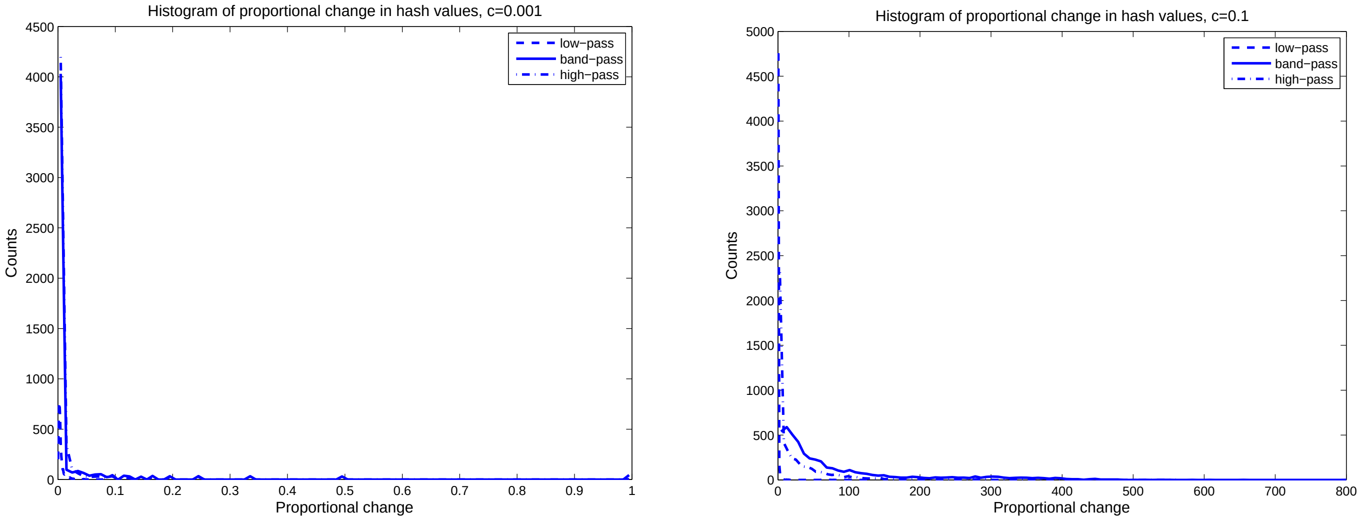
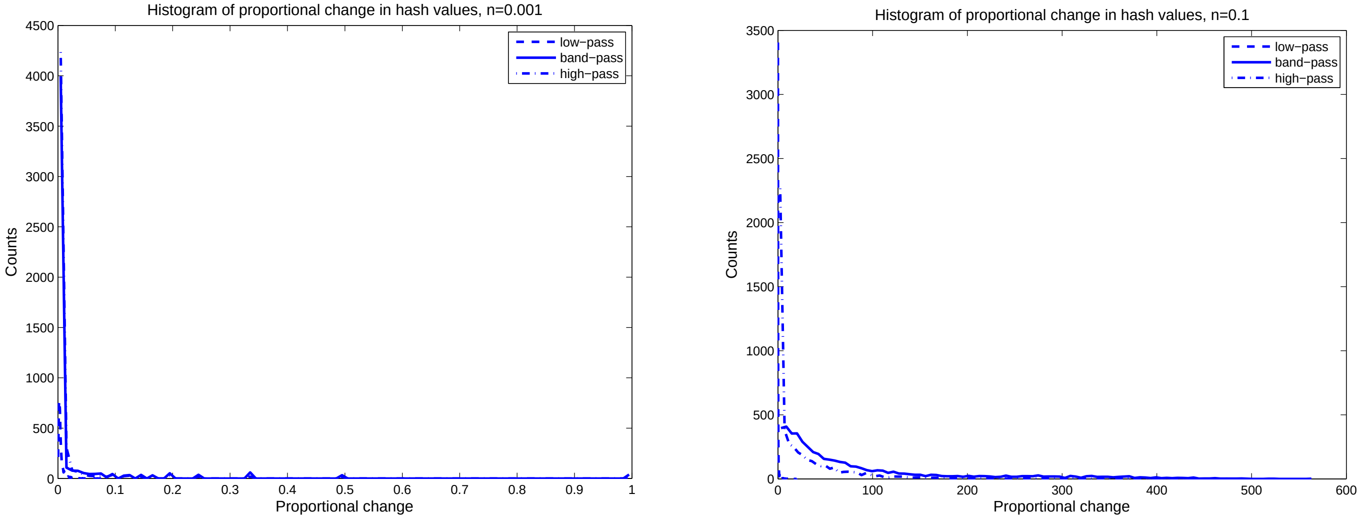


Fig. 15. Fragile watermarking. Percentage change in hash values. DC shift of 1.



(a) (b)

Fig. 16. Fragile watermarking. Percentage change in hash values. Additive noise in space domain. (a) Noise standard deviation $\sigma = 0.001$. (b) Noise standard deviation $\sigma = 0.1$.



(a) (b)

Fig. 17. Fragile watermarking. Percentage change in hash values. Additive noise in frequency domain. (a) Noise standard deviation $\sigma = 0.001$. (b) Noise standard deviation $\sigma = 0.1$.

[10] H. Kargupta, S. Datta, Q. Wang, and K. Sivakumar, “On the Privacy Preserving Properties of Random Data Perturbation Techniques,” in *ICDM*, 2003.

[11] L. Liu, M. Kantarcioglu, and B. Thuraisingham, “The applicability of the perturbation model-based privacy preserving data mining for real-world data,” in *Intl Workshop on Privacy Aspects of Data-Mining*, 2006.

[12] S. Oliveira and O. Zaiane, “Privacy preserving clustering by data transformation,” in *SBBD*, 2003.

[13] Ingemar J. Cox, Joe Kilian, Tom Leighton, and Talal Shamoon, “Secure Spread Spectrum Watermarking for Multimedia,” in *IEEE Transactions on Image Processing*, 1995.

[14] A.V. Oppenheim, A.S Willsky, and S.H. Nawab, *Signals and Systems, 2nd Edition*, Prentice Hall, 1997.

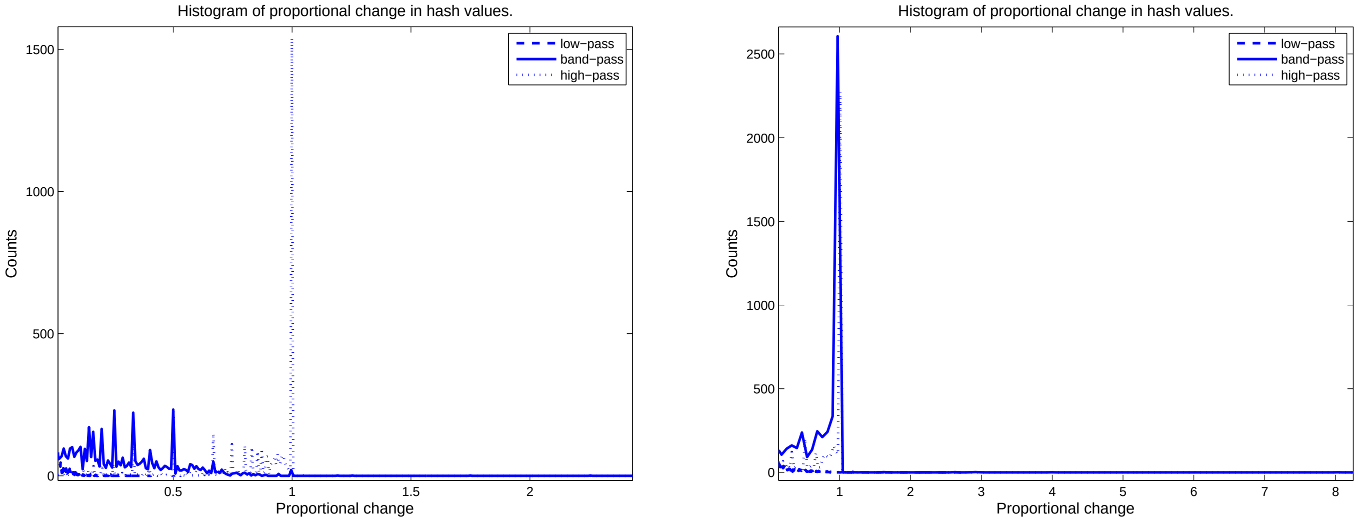
[15] H. Vincent Poor, *An introduction to signal detection and estimation (2nd ed.)*, Springer-Verlag New York, Inc., New York, NY, USA, 1994.

[16] J. Fridrich, “Image watermarking for tamper detection,” in *Proc. ICIP*, 1998, pp. 404–408.

[17] P. Wong, “A public key watermark for image verification and authentication,” in *Proc. Int. Conf. Im. Proc*, 1998, pp. 455–459.

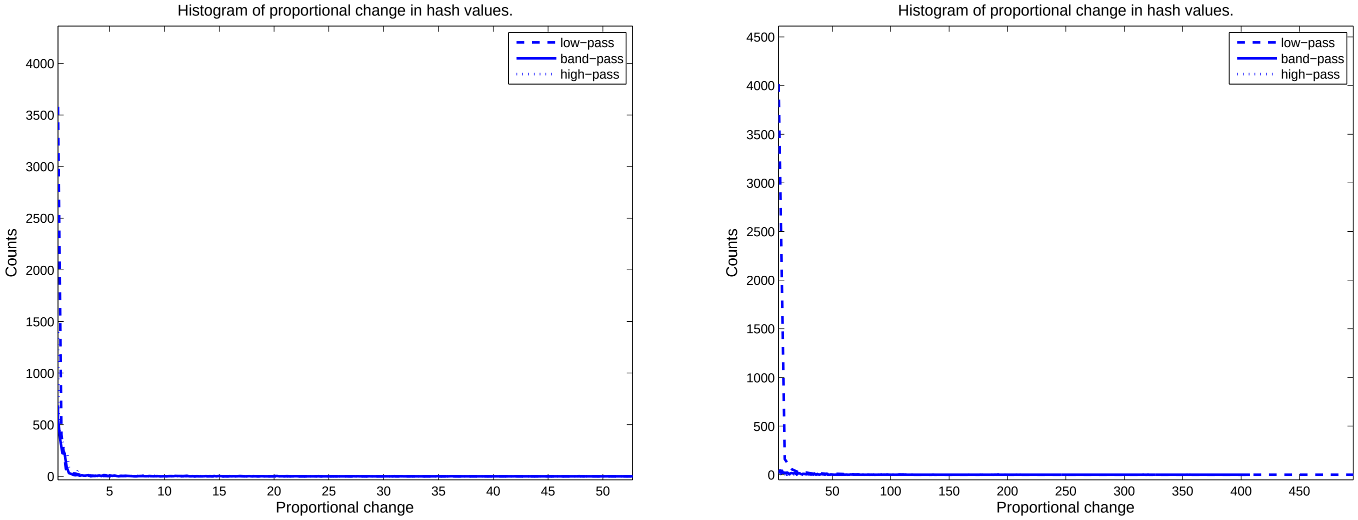
[18] Raymond B. Wolfgang and Edward J. Delp, “Fragile watermarking using the VW2D watermark,” in *Proc. SPIE/IS T Inter. Conf. Security and Watermarking of multimedia Contents*, 1999, pp. 204–213.

[19] P. Moulin, M.E. Mihcak, and G.-I. Lin, “An Information-Theoretic Model For Image Watermarking And Data Hiding,” in *IEEE Int. Conf. on Image Processing*, 2000.



(a) (b)

Fig. 18. Fragile watermarking. Percentage change in hash values. Decimation attack. (a) 50% decimation. (b) 75% decimation.



(a) (b)

Fig. 19. Fragile watermarking. Percentage change in hash values. Cropping attack. (a) 20% cropping. (b) 40% cropping.

[20] S. S. Kozat, R. Venkatesan, and M. K. Mihcak, "Robust hashing via matrix-invariances," in *Proc. of IEEE Conf. on Image Processing* , 2004.

[21] MIT-BIH Arrhythmia Database, "`http://www.physionet.org/physiobank/database/mitdb/`," .

[22] E. Petrucci, V. Balian, G. Filippini, and L.T. Mainardi, "Atrial fibrillation detection algorithms for very long term ECG monitoring," in *Computers in Cardiology*, 2005.

[23] M. Vlachos, P.S. Yu, and V. Castelli, "On Periodicity Detection and Structural Periodic Similarity," in *Proc. of SDM*, 2005.

[24] G. Chiranjivi, V.K. Madasu, M. Hanmandlu, and B.C. Lovell, "Arrhythmia Detection in Human Electrocardiogram," in *APRS Workshop on Digital Image Computing, 1(1), pages 189-192*, 2005.