2006 年度卒業論文

音声データにおける
墨塗り署名ツール
“SANI”の開発

研究指導　菊池浩明　教授

東海大学　電子情報学部　情報メディア学科

3ADM2127 石井利晃

目次

第１章
はじめに

1.1　背景

近年，IC レコーダが普及し，会社での議事録作成や，大学での講義などで用いられる傾向がある．平成 14 年度以降は，文部科学省が推進しているサイバーキャンパス整備事業により，インターネットを介した学習支援システムの構築も進んでいる．東海大学も教育研究システム TICU(Tokai International Cyber University)を展開しているが，講義中の個人名などのプライバシー情報をそのまま公開するわけにはいかない．一方，デジタルコンテンツの偽造を防止するためにはデジタル署名が有効である．しかし，個人名やプライバシーに関する情報を削除してしまうと，署名の検証に失敗し，元のデジタルコンテンツと同一であることを保証できない．

そこで，本研究は，一部が削除されても署名の検証が可能な墨塗り署名ツールの開発を目的とする．対象は，IC レコーダなどの録音機器で使用されている音楽形式のひとつである WAV である．

本稿では、音声データへの墨塗り署名方式の提案と開発したツールのインターフェースの説明、および性能評価を行う．

1.2　音声データの改竄

音声データへの改竄について，１つは音声データの上書きがある．図１のように，元の音声データのコピーを作成し，コピーデータを用いて元の音声データを上書きしてしまう．この方法は，２つの再生フォーマットが同じであるため比較的容易に改竄を行うことができる．

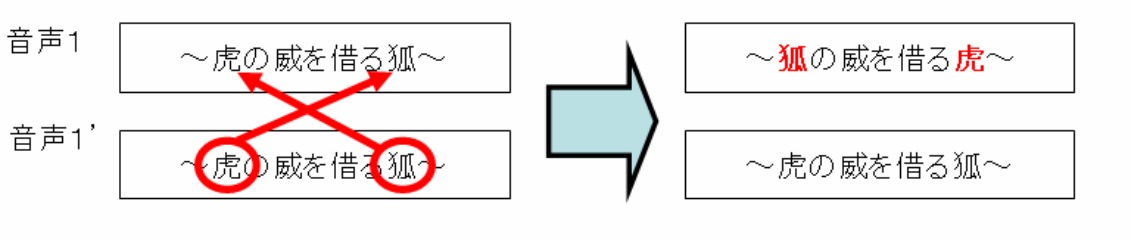

図１．音声データの改竄

もう１つは，音声データを直接書き換える方法である．こちらは任意の音声を出すということは困難であるが，無音や単調な音にすることは可能であり，他のデータを必要としない．

これら 2 種類の方法で安易に音声データの改竄を行うことができる，という仮定の下でツールの開発を行った．

第２章
関連技術

2.1　墨塗り署名

2.1.1　墨塗り署名とは

墨塗り署名は，署名者がある文書に対する署名を生成した後に，墨塗り者が文書の一部を変更(墨塗り)することを許容する署名方式である. [1]では電子文書への署名技術として提案されたが，本研究では音声データに対してこの署名技術を用いる.

[1]で提案されている墨塗り署名技術に求められるセキュリティ要件は次の４つである.

(1)開示文書に墨塗り部分が含まれていても検証が可能であり墨塗り以外の改変がなければ検証に成功すること

(2)開示文書にオリジナル文書に対する墨塗り以外の改変があったときには検証が失敗すること

(3)墨塗り部分に対応するオリジナル文書の情報が漏洩しないこと

(4)墨塗り部分の情報を推定しようとする攻撃者が，開示文書自体を，推定結果の正当性を保証する手段として利用できないこと

2.1.2　SUMI-4 とは

SUMI-4 は，[1]で提案された墨塗り署名方式の１つである.

署名生成，墨塗り，検証の手順を示す．また，モデルを図２に示す.

＜署名生成＞

Step1　オリジナル文書を構成要素毎にＮ個のブロックに分割する

Step2　Ｎ個のブロックそれぞれに対し，ブロックのデータとそのブロックに対して生成された乱数を結合したデータを生成する

Step3　各乱数つきブロックのハッシュ値を算出し，算出されたＮ個のハッシュ値を結合したデータに対し，オリジナル文書作成者の秘密鍵を用いて署名を生成する

Step4　生成された署名と，Ｎ個の乱数つきブロックとからなるデータを署名つきオリジナル文書とする

＜墨塗り＞

Step1　開示対象である署名つきオリジナル文書の中から，不開示情報を含む位置情報つきブロックを選択する

Step2　選択された各乱数つきブロックのハッシュ値と，それ以外の各乱数つきブロックと，署名とからなるデータを，開示文書とする

＜検証＞

Step1　開示文書のうち，もとのデータ自体が与えられている各乱数つきブロックのハッシュ値を算出する

Step2　算出された，または，開示文書に含まれるハッシュ値を結合したデータを，オリジナル文書作成者の公開鍵を用いて検証し，検証結果を出力する

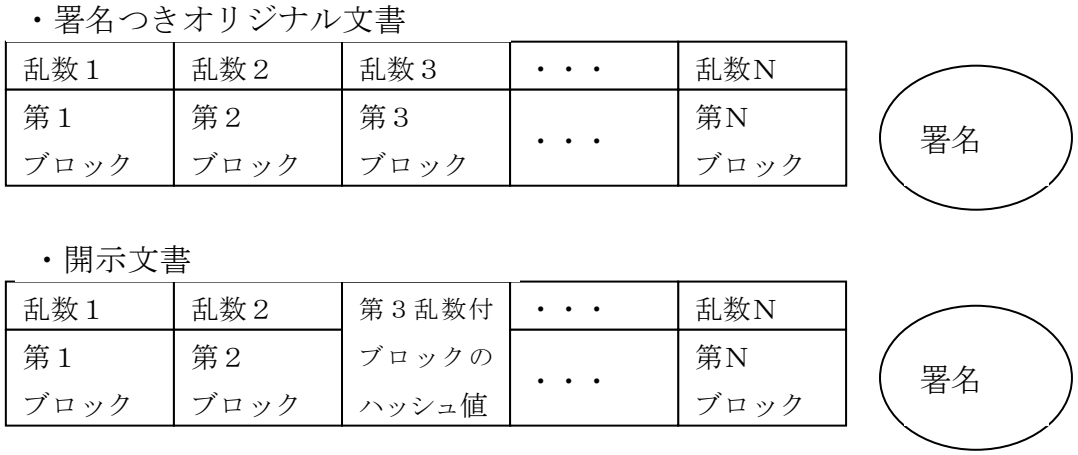

図２．署名つきオリジナル文書と開示文書

2.2　WAV フォーマット

Windows 標準の音楽，音声フォーマットである．特にウェーブ用に設計されたものではなく，マルチメディア情報を格納するために設計されたリソース交換ファイル(RIFF：Resource Interchange File Format)形式に沿っている．

リフ形式は，「チャンク」と呼ばれる単位から成っていて，ファイルは複数のチャンクを数珠つなぎになった構造をしている．チャンクを構成する要素は次のとおり．

- チャンクタイプを識別するコード(4Byte)
- チャンクのデータ部分のサイズ(4Byte)
- 実際のデータ(データサイズが奇数の場合は 1Byte の NULL 文字)

これらのチャンクは別のチャンクに挿入することも可能であり，基本的な WAV ファイルはチャンクの入れ子構造から成る．

WAV ファイルの再生に必要な最低限の構成は，RIFF チャンクの下に fmt チャンクと data チャンクがあることである．WAV ファイルの構造を図３に示す．

左側が実際のデータ(“と”で書かれているデータは固定)，右側がチャンクの入れ子構造のイメージである．

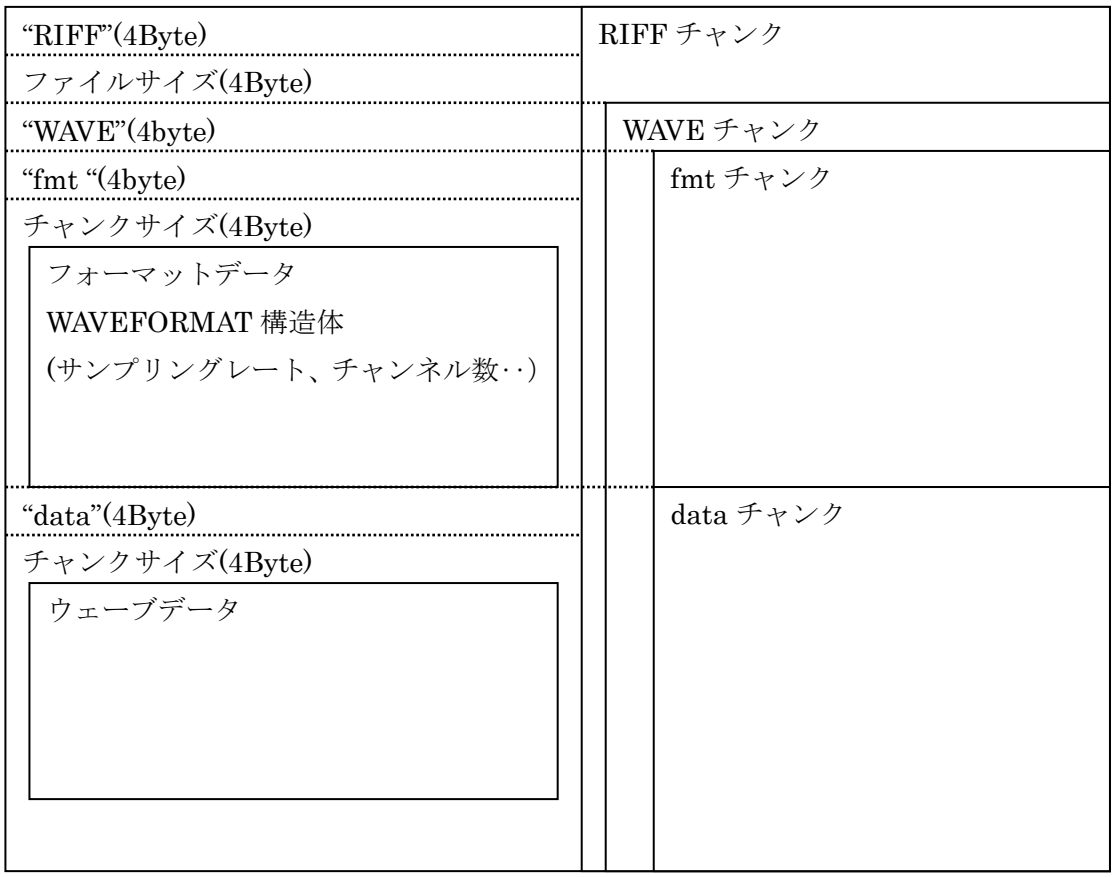

図３．WAV ファイル構成図

第３章
墨塗り署名ツール“SANI”の実装

3.1 音声データへの墨塗り署名方式の適用

第２章で SUMI-4 と WAV フォーマットについて触れたが，図３のファイル構造では乱数を付加することができない. そこで, リフ形式に基づいて乱数を格納するチャンク(仮に rand チャンクとする)を追加する. この方法によって WAV ファイルの音声データには影響されない，またチャンクの特徴から付加するデータの制限がないためファイルサイズの大きいデータでも扱うことができる．墨塗り署名を可能とする WAV ファイルの構造を図４に示す.

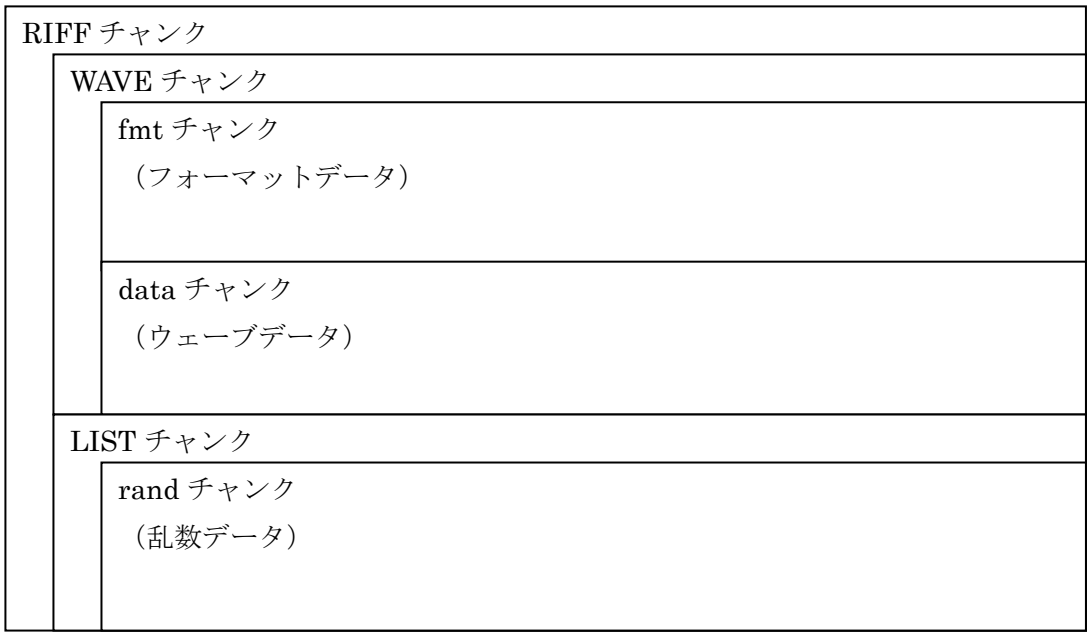

図４．墨塗り署名を可能とする WAV ファイル構造

これにより音声データに墨塗り署名を行うことができる．しかし，実際にウェーブデータが格納されている data チャンクをハッシュ値で上書きしてしまうと, 音声データに雑音・不快音が混ざってしまう可能性がある．そこで，音声データの自由度が高い方法で墨塗り署名を行う.

文書の場合は墨塗りブロック単位が 1 文字以上で行うことができるが，音声の場合は墨塗り指定範囲を 0.1 秒～1.0 秒単位が適切であると考える.（墨塗り範囲を指定する際にビット単位で指定することはないため，人間の知覚可能範囲で考える.）これにより，文書に比べて音声のほうが１ブロックのサイズが大きいことが問題にならない．これに対する乱数のサイズを十分大きくとることにより，ハッシュ値の埋め込みを rand チャンクだけに済ませることができ，data チャンクへの埋め込みの自由度が高くなる．またこの際に，墨塗り署名

方式のセキュリティ要件として，元のデータから音声データを作成すると情報が漏洩する可能性があるため，一定のデータを書き込むことに注意する．今回のツールでは無音データを埋め込むことで解決した．ファイル構成を図５に示す.

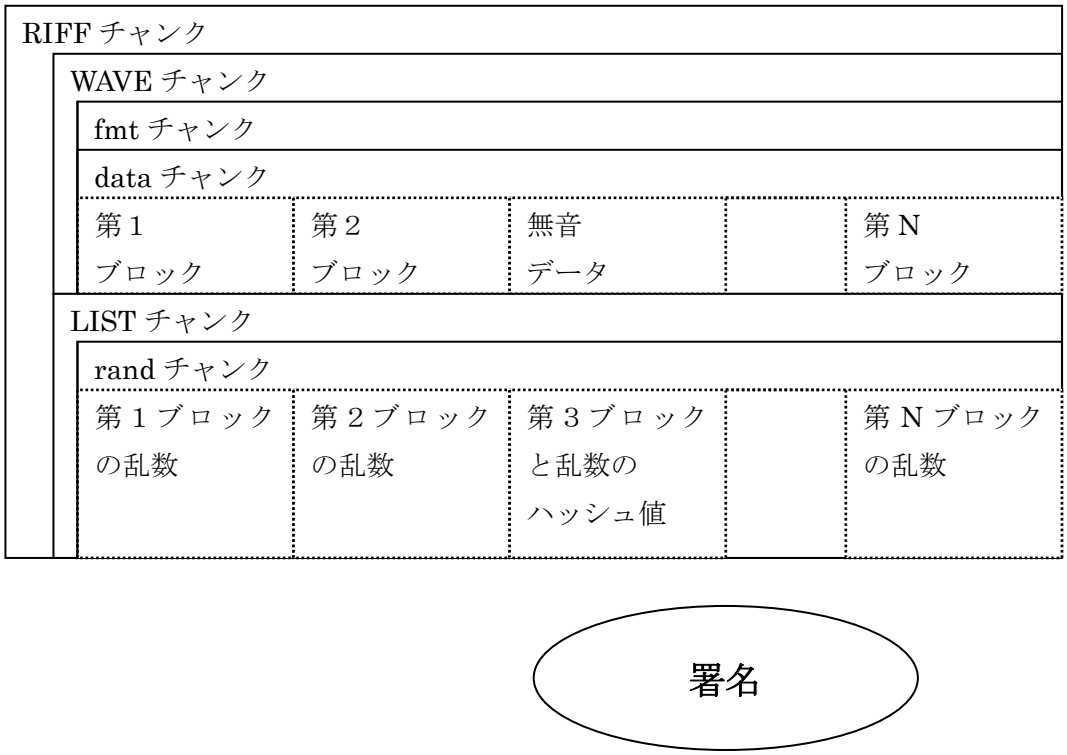

図５．音声データにおける墨塗り署名の開示データ

3.2　ツール構成

本ツールは以下で構成されている.

- SANIK.exe
  秘密鍵と公開鍵を生成するプログラム
- SANI1.exe
  秘密鍵と音声ファイルから署名済み音声ファイルを生成するプログラム
- SANI2.exe
  署名済み音声ファイルに墨塗りを行うプログラム
- SANI3.exe
  公開鍵と墨塗り後の音声ファイルを用いて検証を行うプログラム

3.3　準備

署名者はあらかじめ SANIK.exe を用いて秘密鍵と公開鍵を生成し，検証者が公開鍵を取得できるように準備する.

3.4 使用方法

3.4.1 署名生成ツールの使用方法

(1) SANI1.exe を起動すると Swing 画面が表示される.

(2) “WAV OPEN”ボタン[図６.（ア）]を押し，署名をする音声ファイルを選択する.
（右に選択したファイル名が表示される）

(3) “SecretKeyFile OPEN”ボタン[図６.（イ）]を押し，秘密鍵ファイルを選択する.
（右に選択したファイル名が表示される）

(4) テキストボックス[図６.（ウ）]に墨塗りを行うブロック単位を秒(0.1～1.0)で指定し，”Sign Up”ボタン[図６. (エ)]を押してしばらく待つ.

(5) 処理が終わったら”WAV SAVE”ボタン[図６.（オ）]を押して，署名済み音声ファイルの保存をする.

(6) 次に”Sign Save”ボタン[図６.（カ）]を押して，署名ファイルを保存する.

(2)と(3)，(5)と(6)は逆でも可

図６. SANI1.exe 実行画面

3.4.2 墨塗りツールの使用方法

(1) SANI2.exe を起動すると Swing 画面が表示される.

(2) “WAV OPEN”ボタン[図７.（キ）]を押し，墨塗りする音声ファイルを選択する.
（右に選択したファイル名が表示される）

(3) テキストボックス[図７.（ク）（ケ）]に墨塗り範囲を入力する.

(4) “Sanitizing”ボタン[図７.（コ）]を押して，しばらく待つ.

(5) 処理が終わったら”WAV SAVE”ボタン[図７.（サ）]で墨塗りした音声ファイルを保存する.

(2)と(3)は逆でも可

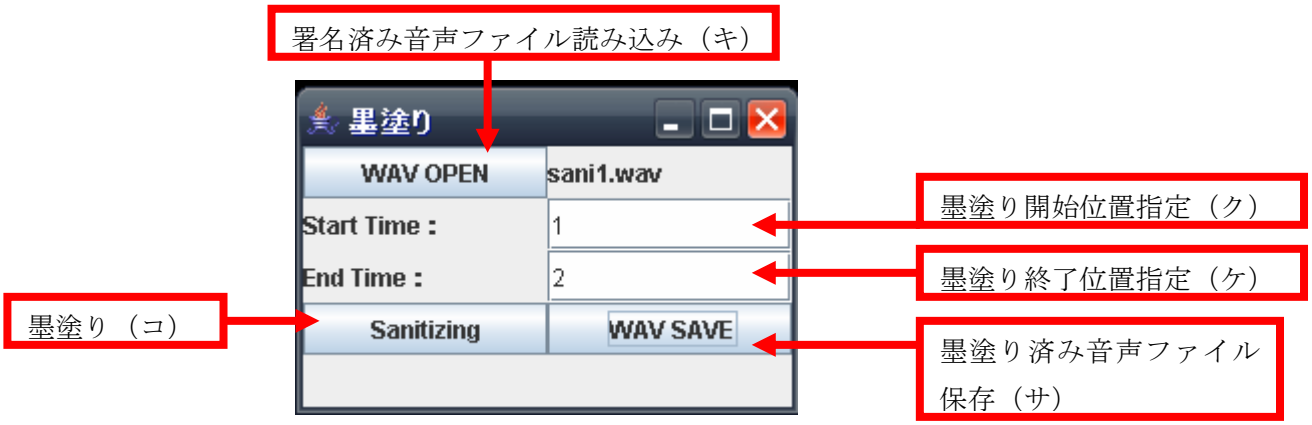

図７． SANI2.exe 実行画面

3.4.3　署名検証ツールの使用方法

(1)　SANI3.exe を起動すると Swing 画面が表示される．

(2)　“WAV OPEN”ボタン[図８．（シ）]を押し，検証する音声ファイルを選択する．
（右に選択したファイル名が表示される）

(3)　“PublicKeyFile OPEN”ボタン[図８．（ス）]を押し，公開鍵ファイルを選択する．
（右に選択したファイル名が表示される）

(4)　“Sign Open”ボタン[図８．（セ）]を押し，署名ファイルを選択する．
（右に選択したファイル名が表示される）

(5)　“Check!!”ボタン[図８．（ソ）]を押し，検証を行う．
検証に成功すれば「Valid」，失敗すれば「InValid」と表示される．

(2)と(3)，(4)の順番は自由

図８． SANI3.exe 実行画面

第４章
評価

実際にこれらのツールを用いて音声データへの墨塗りを行う際に，どの程度の時間がかかるのかを検証する．実験に用いた音声データのフォーマットは以下のとおり，

ビットレート：256kbps
オーディオサンプルサイズ：16bit
チャンネル：モノラル
オーディオサンプルレート：16KHz
オーディオ形式：PCM

である．これらを墨塗りブロック単位 1.0 秒で指定した際の，データサイズにおける実行時間を示したのが図９である．今回のフォーマットでは，約 2MByte が 1 分の再生時間になる．

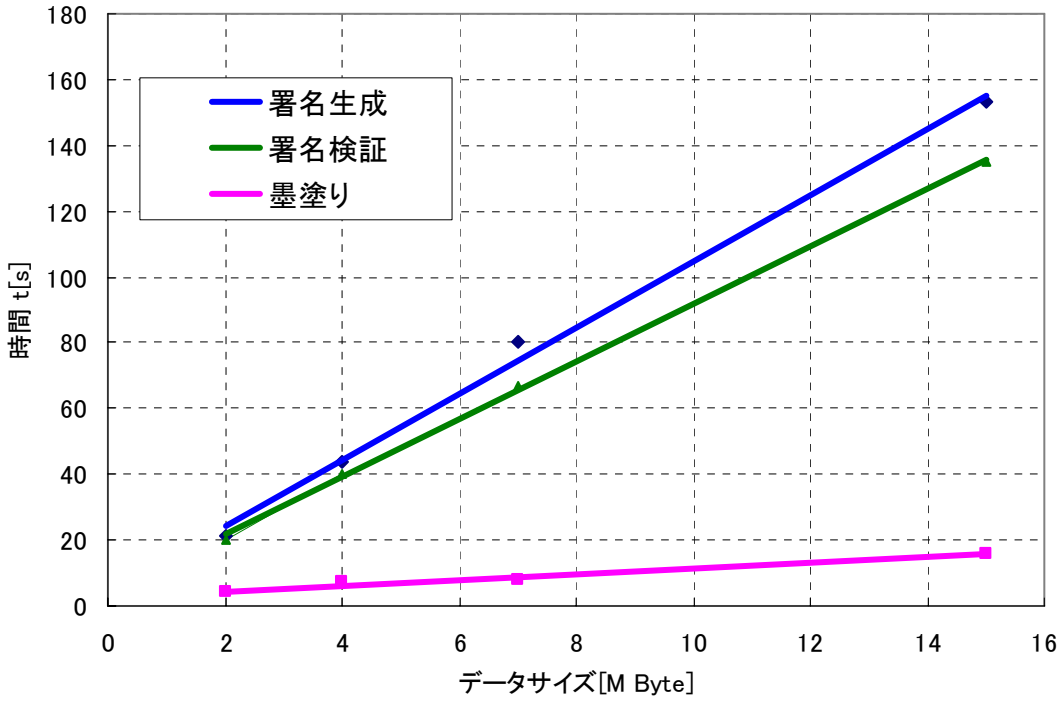

図９．データサイズに対する署名生成・墨塗り・署名検証実行時間

これにより実際の処理時間は，1MByte あたり 10 秒かかることがわかり，比例関係にあることからデータ量の多いファイルの処理時間も推測可能である．

また，同じ音声ファイルに対して墨塗りブロック単位 0.1 秒と 1.0 秒で指定した際の，ファイルサイズ増加量を調べる．表 1 にその結果を示す．

元サイズ	2.09931	4.37048	7.38735	15.27343	31.29726
署名後(0.1)	2.11312	4.39919	7.43587	15.37369	31.50339
増加分(0.1)	0.01381	0.02871	0.04852	0.10026	0.20613
署名後(1.0)	2.10071	4.37338	7.39223	15.28349	31.31845
増加分(1.0)	0.0014	0.0029	0.00488	0.01006	0.02119

表１．署名後の音声ファイル増加量

赤のラインが増加分であるが，1.0 秒から 0.1 秒への増加分の変化は約 10 倍であることから，墨塗りブロック単位の変化によりファイルサイズ増加は比例関係であることがわかる．

この２つの評価から，背景で述べた大学授業を考える．大学授業は 90 分であるので，

90×2[MByte]＝180[MByte]

から１つの授業に必要な音声データは 180MByte であることがわかる．また実際に署名をする際に必要な処理時間は，

180[MByte]×10[秒/MByte]＝1800[秒]＝30[分]

パソコンの性能にも左右されるが，実用に耐えうる数値ではないかと思われる．

第５章
結論

Java を用いて，音声データにおける墨塗り署名ツールを開発した．署名対象が音声データでためファイルサイズが大きく，自然と署名時間が長くなってしまったのは問題であるかもしれない. 原因として, 扱った音声ファイル形式が比較的ファイルサイズが大きい WAV であることも考えられる．MP3 などのもともとのファイルサイズが小さい音声データに対して墨塗り署名を行うことで処理時間の短縮が可能であると考えられる．

この提案方式は，WAV ファイルに限らず RIFF 形式のものに適用することが可能であるので，ファイル形式の調査を進めればツールへの対応ファイルが大きく広がる可能性が高い．また，墨塗り署名方式がもともとは文書，画像に対する署名技術であることを考慮すれば，動画データへの応用も考えられる．

これらのことから，より多くのファイル形式に対応する必要性があると考えられる．

参考文献

[1] 宮崎 他: “電子文書墨塗り問題”, 電子文書通信学会(ISEC2003) , pp. 61-67,(2003).

謝辞

本研究を完遂するにあたり御指導をいただきました東海大学電子情報学部情報メディア学科菊池浩明教授，また菊池研究室の大学院生の方々に心より感謝申し上げます.

特に，常日頃からの細かな御指導をしていただいた上山真梨さんには深くお礼申し上げます.

この場をかりて感謝の意を述べるとともに，謝辞とさせていただきます.