個人情報保護を考慮した電子文書公開システム

秦野康生[†] 宮崎邦彦[†] 手塚悟[†]

2005 年 4 月に施行された通称 “電子文書法” によって，元々紙で作成されていた帳票などの書類を電子化した画像データも公式な書類として認められるようになった．これら電子文書に対しては，“だれによって作成されたか”，“改竄されていないか” といった事項を確認するため，電子署名を使用することなどが各省庁のガイドラインによって定められている．ところで，電子文書法と同時期に成立した個人情報保護法では，個人情報の取扱いを厳格に行うことを定めており，これら電子化された書類を第 3 者に公開する場合には，名前などの個人情報は削除（墨塗り）するなどの処理を行う必要がある．しかしながら，従来の電子署名技術では，このような墨塗りされた電子文書の検証を行うことができない．これを解決する署名方式として，電子文書墨塗り技術が提案されている．本論文では，電子文書墨塗り技術を用い，紙文書を電子化した画像データに対し，個人情報保護の観点からの墨塗りを考慮した電子文書公開システムを提案する．また，標準的な画像ファイルフォーマットである JPEG ファイルに対し，電子文書墨塗り技術適用時の問題点とその解決方法について示し，墨塗りに対応した電子署名の実装とその実験結果について報告する．

An Information Disclosure System of Digital Documents with Protecting Personal Information

Yasuo Hatano,[†] Kunihiko Miyazaki[†] and Satoru Tezuka[†]

Due to “e-Document Law”, which has been effective since April, 2005, digital documents scanned from paper documents are also accepted officially. Since digital documents can be altered without any evidence, guide lines for e-Document Law, published by government agencies, require that digital signature must be used to check “who wrote the document”, “whether the document is not forged” and so on. Also, Law Protecting Personal Information, which has become effective in the same month as e-Document Law, requires personal information to be managed carefully and therefore personal information, e.g., name and birthday, should be removed from a document published for a third party. Unfortunately, if a part of a document is removed, current digital signature technique can not verify the document. In order to solve this problem, “Digitally Signed Document Sanitizing Scheme (DSDSS)” was proposed by Miyazaki, et al. in 2002. This paper presents an information disclosure system for image files scanned from paper documents, using DSDSS. Moreover, this paper shows problems and a solution for them, when DSDSS applies a well-known image file format “JPEG”. This paper also gives an implementation and its performance of DSDSS for JPEG file format.

1. はじめに

2005 年 4 月に施行された通称 “電子文書法” によって，紙で作成された書類を電子的に取り込んだ画像データも公式な書類として認められるようになった[9]．このような電子化された書類は，従来の紙による書類の作成/保管と比較して，改変を行った場合にその痕跡が残らないため，改竄が容易になる．そのため，各省庁から提示されているガイドラインでは，電子文書が “だれによって作成されたのか”，“改竄されていないか” を確認するため，書類の電子化を行う際，電子署名の付与を求めている（文献 11) など参照）．電子署名では，本人しか知りえない秘密鍵を用いて電子文書の署名を行い，それに対応する公開鍵を用いて電子文書の検証を行うことによって，電子文書に対して改竄が行われたか否かを知ることができる．また，認証局と呼ばれる第 3 者機関が公開鍵証明書を発行することによって，秘密鍵の保有者とそれに対する公開鍵の保証を行っており，電子文書がだれによって作成（署名）されたのかを知ることが可能である．

一方で，作成/保管された書類が任意の第 3 者に公開される場合には，開示に不要な個人情報は削除したうえで公開されなければならない．たとえば，公的文

† 株式会社日立製作所システム開発研究所
Systems Development Laboratory, Hitachi, Ltd.

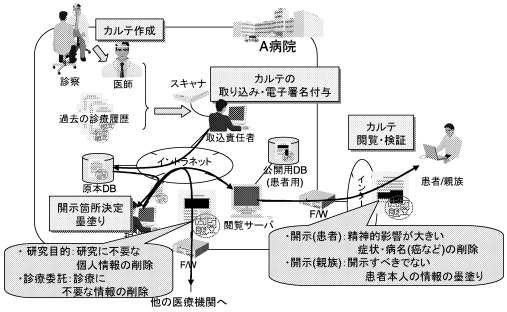


図 1 電子カルテ利用シーンにおける墨塗りの例
Fig. 1 An example removing personal information on medical record.

書が情報公開制度に則って公開される場合には，公開される書類に含まれる個人情報や国家機密情報などは削除（墨塗り）したうえで公開することが定められている[7]．2005 年 4 月に施行された個人情報保護法[8] では，民間企業に対しても個人情報の取扱いを厳格に行うことを定めており，個人情報保護の観点からの墨塗りは，公的機関だけでなく民間企業でも重要な問題となってきている．その一例として，電子カルテ公開における墨塗りを**図 1** に示す．

しかしながら，公開される電子文書に対して墨塗りが行われた場合には，電子署名による検証ができなくなる．これは，墨塗りを行うことが電子文書に対する改竄と見なされるためであり，従来の電子署名技術では，個人情報保護の観点からの墨塗りと電子署名による電子文書の完全性保証の双方を達成することはできない．このような問題に対する署名方式として，電子文書墨塗り技術[14),15)] が提案されている．

本論文では，紙で作成された書類を電子化した画像データを対象とした電子文書公開システムについて提案する．また，画像データとして標準的なデータフォーマットの 1 つである JPEG ファイルに対して，電子文書墨塗り技術を適用することによって，墨塗りに対応した電子署名の生成方法について述べる☆．

1.1 関 連 研 究

個人情報の削除（墨塗り）によって電子署名の検証ができなくなる問題は，Bull らによっても指摘されており[4]，この問題を考慮した完全性保証についての提案が行われている[3),4)]．また，吉岡らは，電子文書の訂正を考慮に入れた電子署名による完全性保証技術について提案を行っており，訂正の一形態として墨塗りについても言及している[16),17)]．これらの関連研究では，電子文書の流通における墨塗りの可能性について述べられているが，本論文で述べるような電子文書公開システムそのものの構成については述べられていない．

また，上記の関連研究では，XML を用いた実装例についても示されている[3),17)] が，紙の書類を取り込んだ画像データに対しては実装例が示されていない．XML を利用した場合には，XML 要素を墨塗りの単位とすることで電子文書墨塗り技術の実装が可能であるが，JPEG ファイルのような画像データに対しては電子文書墨塗り技術を直接適用できない．本論文では，JPEG ファイルへ電子文書墨塗り技術を適用した場合の問題点とその適用方法について示す．

本論文の構成は次のとおりである．まず，次章において，電子文書公開システムの提案を行う．次に，3 章において電子文書墨塗り技術と JPEG ファイルの概略について述べ，4 章において，JPEG ファイルへの電子文書墨塗り技術適用時の問題点とその解決方法について示す．5 章において，JPEG ファイルへの墨塗りに対応した電子署名の実装と作成したプログラムの概略を示す．また，6 章に実装したプログラムの実験結果について報告し，その考察を行う．最後に 7 章にまとめを述べる．

2. 電子文書公開システム

2.1 セキュリティ要件

紙で作成された文書の電子化にあたっては，見読性，完全性，機密性，検索性の 4 つの要件が求められる（文献 10) などを参照）．これらのうち，セキュリティの観点からは次の 2 点が重要である．

完全性：電子文書の消失や改竄の防止

機密性：情報への不正なアクセスの防止

完全性確保のためには，電子署名の付与や電子書類に対する改変記録の管理などの措置が必要となる．また，機密性確保のためには，アクセス制御や暗号化などの措置が必要となる．これらの機密性確保に関する措置は，個人情報保護の観点からの不正な個人情報へのアクセス防止なども含まれるが，電子文書公開システムでは，さらに開示に不要な個人情報の削除（墨塗り）を行うことが必要となる．

上記の措置のうちアクセス制御やデータの暗号化などについては，従来技術やシステムの適切な管理によって解決することができる．しかしながら，電子文

☆ JPEG ファイルのような非可逆圧縮のデータフォーマットを用いることで，保管される電子文書のデータサイズを削減することが可能である．ただし，非可逆圧縮のデータフォーマットを使用する場合には，“業務などに支障がない精度であること”，“対象となる紙文書の汚れなどの状態まで判定できること” などの要件が求められる（たとえば，文献 11) など）．

書の公開に際して墨塗りが行われた場合には，従来の電子署名技術では検証を行うことができなくなる．そのため，個人情報保護のための機密性確保と電子署名による完全性保証のいずれか一方を達成することができなくなる．本論文では，電子文書墨塗り技術を用いることで，上記の 2 要件を満たす電子文書公開システムについて提案する．

2.2 システム構成

図 2 に，電子署名による完全性保証と個人情報保護の観点からの墨塗りを考慮に入れた電子文書公開システムの構成を示す．以下では，電子文書として，紙で作成された書類を取り込んだ画像データを対象とする．本システムでは，署名者装置と墨塗り者装置，検証者装置の 3 つの装置から構成される．各装置で行われる処理は次のとおりである．

署名者装置

署名者装置では，スキャナなどから入力された画像ファイルに対し，署名者の署名用秘密鍵を用いて署名を行う．

墨塗り者装置

墨塗り者装置では，開示対象となる署名済みの画像ファイルから，個人名など，開示に際して不適切な個人情報の墨塗りを行う．

検証者装置

検証者装置は，公開された墨塗り済み画像ファイルの閲覧を行い，開示に際して墨塗り以外の不正な改変が行われていないかの検証を行う．

図 2 において，原本 DB は検証者装置からはアクセスできないようにファイアウォールなどを用いてアクセス制御を行う必要がある．また，請求者からの要求に対して，そのつど電子文書を墨塗りして請求者に渡す場合には，公開用 DB は不要である．なお，提案システムでは，署名，墨塗り，検証の各処理は，すべて電子文書墨塗り技術を用いて行われる（概略は次章，または付録参照）．

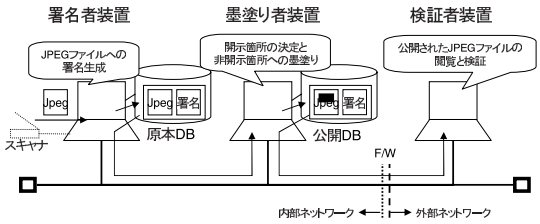


図 2 電子文書公開システムの概略
Fig. 2 Overview of Disclosure System of Information.

3. 提案システムにおける利用技術

3.1 電子文書墨塗り技術[14),15)]

電子文書墨塗り技術は，墨塗りを行った場合でも開示箇所に対する完全性保証が可能な，電子署名の応用技術である．電子文書墨塗り技術では，署名者が電子文書を複数の墨塗り可能な最小単位（以下，墨塗りブロックと呼ぶ）に分割し，墨塗りに対応した電子署名の生成を行う．これによって，署名が付与された電子文書であっても，墨塗りブロックごとでの削除（墨塗り）を行うことが可能であり，それ以外のいかなる改変に対しても電子文書の改竄を検知することが可能となる．なお，墨塗りブロックには，セキュリティ上の理由から乱数などの補助データが付与される（電子文書墨塗り技術における署名生成・墨塗り・検証手順の詳細については，付録参照）．

3.2 JPEG ファイル[2)]

図 3 に JPEG ファイルへの変換プロセス，およびファイルの概略を示す☆．図 3 に示したように，JPEG ファイルは MCU（Minimum Coded Unit）と呼ばれる単位によって構成されている．入力された画像データは，まず YCbCr などの色成分に分割され，色成分ごとに MCU に分割される．MCU に分割されたイメージデータは，8×8 画素のブロック（以下，JPEG 要素と呼ぶ）に分割され，DCT，量子化，エントロピー符号化の処理を行い，MCU 単位でファイルに保存される．なお，DCT 変換では，入力された JPEG 要素が周波数成分に分割され，直流成分と 63 個の交流成分に分割されるが，直流成分については，エントロピー符号化において，直前の JPEG 要素との差分値に対して符号化を行う．

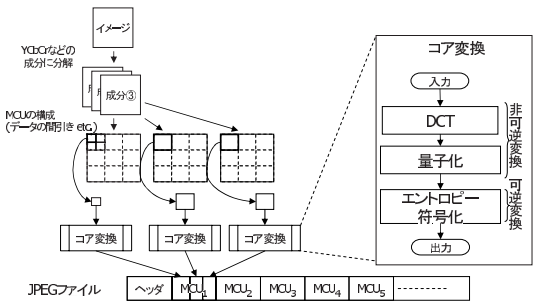


図 3 JPEG ファイルフォーマット
Fig. 3 JPEG file format.

☆ JPEG には，可逆圧縮方式と，非可逆圧縮方式の 2 つの方式があるが，ここでは非可逆圧縮方式のみを示す．なお，本論文では非可逆圧縮の方式を単に JPEG と呼ぶ．

4. JPEG ファイルへの電子文書墨塗り技術の適用

本章では，JPEG ファイルへの電子文書墨塗り技術適用時の問題点とその解決方法について示す．

4.1 適用時の問題点

JPEG ファイルに電子文書墨塗り技術を適用する場合，以下に述べる 3 点が問題となる．まず，JPEG ファイルが画像データであるため，次の問題点があげられる．

(1) 墨塗りによってデータの欠落が生じること
電子文書墨塗り技術では，対象となる墨塗りブロックのデータを削除することによって墨塗りを行う．そのため，JPEG ファイル内にデータの欠落が生じ，通常の閲覧ソフトでは墨塗り後の JPEG ファイルを表示できなくなる．

上記の問題を解決するため，墨塗りされた領域を何らかの代替画像によって置き換えることが必要となる．単純な代替画像への置き換え方法としては，次の 2 つの方法が考えられる．

- JPEG ファイルから得られる表示イメージの墨塗り対象領域を直接黒のイメージデータなどに置き換える．
- JPEG ファイルのブロック構造（MCU）を利用し，墨塗りの対象となる MCU を黒のイメージデータなどに置き換える．

しかしながら，このような単純な代替画像への置き換え方法では，次のような問題が生じる．

(2) 変換プロセスが非可逆変換であること
JPEG ファイルへの変換には非可逆変換プロセスが含まれている．そのため，表示イメージに対して直接墨塗りを行うと，墨塗り後の JPEG ファイルと元の JPEG ファイルの間に相違が生じ，墨塗り後の JPEG ファイルを検証できなくなる．

(3) 各 JPEG 要素間に依存関係があること
エントロピー符号化において，直流成分は直前の JPEG 要素との差分値に対して符号化が行われる．そのため，JPEG ファイルのブロック構造（MCU）を利用して墨塗りを行った場合には，墨塗り後の JPEG ファイルから意図した表示結果を得ることができなくなる．

これらの問題によって，直接電子文書墨塗り技術を適用することができない．

4.2 適 用 方 法

前節で述べた問題点 (1)，(3) は墨塗り後の JPEG ファイルの表示に，問題点 (2) は，署名生成，検証にかかわる問題である．上述のように，問題点 (1) を解決するためには，墨塗りされた墨塗りブロックを代替画像（たとえば黒のイメージデータ）によって置き換えることが必要となる．しかしながら，単純な置き換え方法では，問題点 (2)，(3) から，電子文書墨塗り技術を JPEG ファイルに適用することができない．以下では，まず問題点 (2)，(3) に対する解決方法について述べ，次に代替画像への置き換え方法について述べる．

問題点 (3) は，変換プロセスにおけるエントロピー符号化において発生する．また，問題点 (2) から，非可逆変換前のデータに対して電子文書墨塗り技術の各処理を適用することはできない．そこで，エントロピー符号化前の中間データに対して，署名・墨塗り・検証の各処理を行う．エントロピー符号化は可逆変換であり，エントロピー符号化前の中間データは他の JPEG 要素からの影響を受けていない．したがって，エントロピー符号化前の中間データに対して電子文書墨塗り技術を適用することによって，JPEG ファイルに対して墨塗りに対応した電子署名の付与が可能となる．なお，JPEG ファイルへの変換プロセスが MCU を単位として行われることから，墨塗りブロックとして指定できる最小単位も MCU となる．

次に，代替画像への置き換え方法について述べる．前述のように墨塗りブロックの最小単位が MCU となるため，代替画像への置き換えも MCU を単位として行う．ここで，問題点 (3) から，単純に墨塗りの対象となる MCU のデータを黒色のイメージデータに置き換えただけでは，墨塗り後の JPEG ファイルから意図した表示イメージを得ることができなくなる．そのため，墨塗りを行う際には，JPEG ファイルをいったんエントロピー符号化前の中間データに復号してから代替画像への置き換えを行い，その後再びエントロピー符号化を行う必要がある．エントロピー符号化は可逆変換であり，JPEG ファイルへの変換プロセスから，エントロピー符号化前の中間データは他の JPEG 要素からの影響を受けていないため，問題点 (3) によって生じる問題を回避することが可能となる．なお，代替画像については，任意の画像データを用いることが可能であるが，以下では，代替画像は黒のイメージデータによって置き換えられることを前提とする．検証の際には，電子文書墨塗り技術における検証処理のほか，墨塗りされた領域が黒のイメージデータによって置き換わっていることを確認することが必要となる．

以上に述べた方法によって，JPEG ファイルに対し

て電子文書墨塗り技術の実装が可能になる．すなわち，JPEG ファイルに対し，墨塗りに対応した電子署名の付与，および，墨塗り後の JPEG ファイルを通常の閲覧ソフトを用いて閲覧することが可能となる．

5. JPEG 対応電子文書墨塗り技術の実装

電子文書墨塗り技術では，墨塗りブロックの対象領域や付与される乱数などの情報（以下，墨塗り用補助データと呼ぶ）を管理する必要がある．本章では，墨塗り用補助データの XML を利用した管理と，これを用いた署名，墨塗り，検証の実装について述べ，作成したプログラムについて報告する．

5.1 XML を用いた実装

図 4 に，XML を利用した墨塗り用補助データと署名フォーマットの概略を示し，以下に墨塗り用補助データと署名フォーマットの概略を示す．

(1) 墨塗り用補助データ

電子文書墨塗り技術では，対象となる電子データを複数の墨塗りブロックに分割し，各墨塗りブロックに対して，開示・非開示が決定される．さらに，これらの墨塗りブロックにはセキュリティ上の理由から，乱数などの補助データが添付される場合がある．

そこで，これらの墨塗り用補助データを管理するための XML 要素を定義し，これを用いて画像データ内の墨塗り領域への参照を行う（図 4 参照）．ここで，前章で述べたように，JPEG ファイルに対して電子文書墨塗り技術を適用する場合には，MCU が墨塗りブロックの最小単位となる．そのため，上記の XML ファイルで管理される墨塗りブロックの対象領域は MCU 単位で管理を行う．

また，ヘッダ領域では，署名対象となる JPEG ファイルの識別情報の管理を行う．電子文書墨塗り技術では，署名・墨塗り・検証の各処理において，墨塗りブロックのハッシュ値などの計算が必要となる．そのためヘッダでは，JPEG ファイルの識別情報のほか，使用されるハッシュ関数など，電子文書墨塗り技術固有の情報の管理を行う．

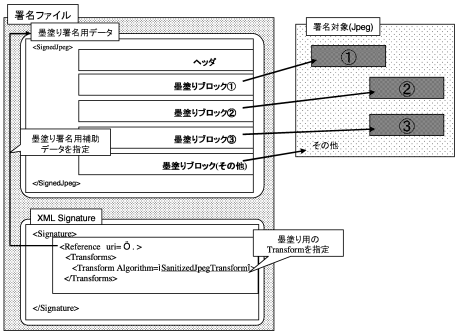


図 4 XML による墨塗り用補助データの管理
Fig. 4 Auxiliary data for DSDSS using XML.

(2) 署名の生成と墨塗り，検証

署名，墨塗り，検証の各処理を行う場合には，すべて墨塗り用補助データを用いて行う．まず，墨塗りを行う場合には，墨塗り用補助データ内の情報を参照して，JPEG ファイル内の墨塗りの対象となる墨塗りブロックの領域参照を行い，電子文書墨塗り技術の処理手順に従って，墨塗り用補助データと JPEG ファイルの更新を行う☆．

署名の生成・検証には，本提案システムでは XML Signature[1] を用いる．XML Signature では署名対象要素を Reference 要素を用いて指定し，各署名対象ごとにハッシュ値の算出を行い，署名値の計算を行う（詳細は付録参照）．このとき，Refrence 要素で指定された署名対象要素は Transform 要素を用いて操作を行うことができる．提案システムでは，この処理手順に着目し，XML Signature を利用して電子文書墨塗り技術の実装を行う．すなわち，Transform 要素に JPEG ファイルに電子文書墨塗り技術のための新たな変換を定義し，これを用いて署名を行う☆☆．

5.2 プログラムの概略

以上に述べた墨塗り用補助データを用いて電子文書公開システムを構成する各プログラムの実装を行った．実装したプログラムの概略は次のとおりである．なお，各プログラムは，( ) 内に示した装置内に実装される（装置の概略については 2.2 節，および図 2 参照）．

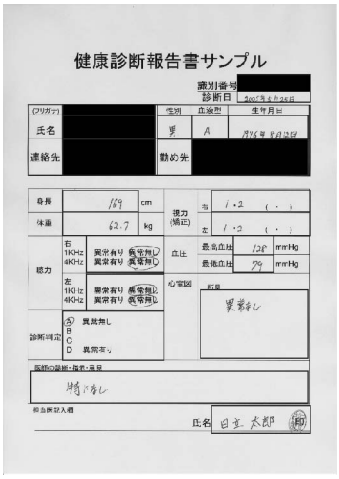
墨塗り領域設定プログラム（署名者装置）
署名対象となる JPEG ファイルに対し墨塗りブロックを設定し，墨塗り領域設定ファイルを出力する．

署名生成プログラム（署名者装置）
署名対象となる JPEG ファイルと墨塗り領域設定ファイル，署名者の署名用秘密鍵を用いて，JPEG ファイルに対して墨塗りに対応した電子署名の生成を行い，署名ファイルと署名済み JPEG ファイルを出力する．

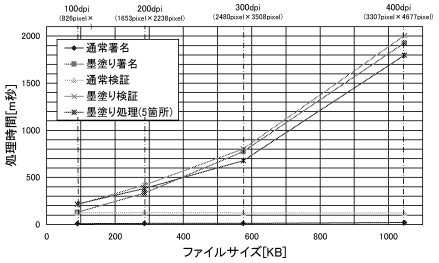
墨塗りプログラム（墨塗り者装置）
JPEG ファイルとそれに対応する署名ファイルを用いて，JPEG ファイルに対して墨塗りを行い，墨塗り後の JPEG ファイルと署名ファイルを出

☆ 前章で述べたように，実際に操作の対象となるのは，JPEG ファイルのエントロピー符号を復号した中間データである．

☆☆ SUMI-4 を用いる場合には，付録に示した検証処理の 1〜7 を実行する．



(a) 使用したサンプル画像（墨塗り後）



(b) 実験結果

● 実験環境☆
.NET Framework on Windows®XP (SP2)
Pentium®4 2.8 GHz, 512 MB RAM
JPEG ファイルの操作には，Independent JPEG Group のライブラリを利用
● 画像情報
対象書類：A4，カラー
墨塗りブロック数：22 カ所
（図中の太枠 ＋ その他）
*実験結果は，100 回の平均値を示した．また，墨塗り処理は，サンプル画像と同じ箇所を墨塗りした．

図 5 実験結果
Fig. 5 Experimental result.

力する．

検証プログラム（検証者装置）

公開（墨塗り）された JPEG ファイルとそれに対応する署名ファイルを用いて，JPEG ファイルの検証を行い，その結果（検証成功/失敗）を出力する．

ここで，署名者装置内に，墨塗り領域設定プログラムと署名生成プログラムが存在する理由は，JPEG ファイルを墨塗り領域に分割するためだけでなく，署名対象となる JPEG ファイルが定型的な書式を持っている場合に，テンプレートとなる墨塗り領域設定ファイルを作成しておくためである．

これらのプログラムとスキャナなどの読み取り装置，データベースなどを用いることで，電子文書公開システム（図 2）を構成することができる．

6. 評　　価

本章では，前章で述べた電子文書公開システムを構成する各プログラムの実験結果について示し，その考察を行う．

6.1 実 験 結 果

前章で示したプログラムについて，署名，墨塗り，検証の各処理性能について実験を行った．**図 5** (a) に実験に使用したサンプルを示し，実験結果を図 5 (b) に示す．

図 5 において，“通常署名”，“通常検証” は，署名対象（Reference 要素）に JPEG ファイルを指定した場合の XML Siganture による署名，検証を表し，“墨塗り署名”，“墨塗り検証” は，本論文で述べた墨塗りに対応した電子署名を利用した場合の署名，検証を表す．使用したサンプルは A4 版カラーの個人情報を含む書類であり，墨塗りは，氏名，フリガナ，住所，勤務先，および，書類を管理するための識別番号の 5 カ所に対して行った．なお，実験結果はすべて 100 回の実行時間の平均値である．

図 5 に示したように，墨塗りに対応した電子署名を生成した場合，通常署名の場合と比較し，処理速度が遅くなる．これは，墨塗りに対応した電子署名を生成した場合には，JPEG ファイルのエントロピー符号の復号や墨塗り用補助データの解析などが行われるためである．たとえば，300 dpi で取り込んだ場合には，通常の電子署名の場合には，署名，検証は，およそ 12 m 秒，120 m 秒の処理時間であるのに対し，墨塗りに対応した電子署名の場合には，署名，検証に，それぞれ，780 m 秒，800 m 秒を要している（このときの墨塗りの処理時間は約 680 m 秒）．また，墨塗りに対応した電子署名について，ファイルサイズの増加に対する，署名，墨塗り，検証の各処理時間の増加はほぼ同じであった．

6.2 考　　察

電子文書公開システムでは，すべての電子文書が公開され，検証されるわけではない．一般には，大量に電子化された書類の一部が，公開請求に応じて墨塗り

☆ Windows は，米国およびその他の国における米国 Microsoft Corp. の登録商標です．
Pentium は，Intel Corporation のアメリカ合衆国およびその他の国における登録商標です．
Independent JPEG Group URL: http://www.ijg.org/

され，開示先においてその検証が行われる．そのため，最も利用頻度が高い処理は署名生成処理となる．

そこで，署名処理について，本実験結果の考察を行う．なお，文献 11)，12) などでは，紙で作成された書類の電子化に際して，150 dpi〜300 dpi の読み取り解像度を推奨している．そこで，以下の考察では 300 dpi における処理速度をもとに考察を行う．

文献 12) には，大量の契約書を電子化する例として，次の 3 つが示されている．

(1) 書類 A4 版 6.5 億枚/年（拠点 500 カ所）
(2) 書類 A4 版 1.75 億枚/年（拠点 100 カ所）
(3) 書類 A4 版 6.5 億枚/年（拠点 1 カ所）

(1) の場合には，拠点 1 カ所あたりで平均 13 万枚/年の書類を処理する必要があり，1 日あたりに換算すれば，拠点ごとに処理する書類数は約 400 枚となる．これをもとに換算すると，1 枚の処理は 24 秒程度で行われる必要がある．また，(2) について同様の見積りを行うと，1 枚あたり 18 秒程度で処理する必要がある．実験結果（図 5）より，読み取り解像度 300 dpi における署名生成の処理時間は 800 m 秒程度であることから，1 拠点あたり PC1 台で処理し，書類 1 枚ごとに墨塗りに対応した電子署名を付与した場合でも，上記の例 (1)，(2) に対しては，十分処理可能であると考えられる．なお，(3) について同様の見積りを行うと，1 枚あたりの処理を 50 m 秒で行う必要があるため，1 台の PC で墨塗りに対応した電子署名を付与することは困難である．そのため，PC を並列化し，署名処理を分散させるなどの対策が必要となる．

前述したように，一般に電子文書公開システムでは電子化された書類すべてに電子署名が付与され，その一部に対して，墨塗り，検証の処理が行われる．実験結果より，墨塗り，検証に要する時間は，署名生成処理とほぼ同じであるため，提案システムを実現するために十分な処理能力を有していると考えられる．

7. ま と め

本論文では，紙で作成された書類を電子化した画像データに対して，電子文書墨塗り技術を適用し，これを用いた電子文書公開システムについて提案した．本システムを用いることによって，個人情報保護の観点からの墨塗りと電子署名による完全性保証を考慮した電子文書の公開が可能となる．

本論文では，電子文書公開システムとして，紙をスキャンした画像データについて取り上げた．このような JPEG ファイルに対する墨塗りは，ディジタルカメラで撮影された画像データなどにも適用可能である．これを用いるならば，たとえば，Web 上のニュースを掲載する際に，写真から不適切な部分の削除などを行った場合でも，掲載された写真の電子署名による検証が可能となる．これにより，掲載された写真の撮影者の確認が可能となり，著作権の確認が可能となる．このように，電子文書墨塗り技術を JPEG ファイルに適用することによって，個人情報保護のほか，様々な分野への応用が期待できる．

また，今回の実装では，対象とする JPEG ファイルのサイズの増加にともない，電子文書墨塗り技術適用時の処理速度が通常の署名生成処理の時間に比べ大きく低下している．より高速な電子文書墨塗り技術の適用方法，実装については，今後の課題である．

参 考 文 献

1) XML-Signature Syntax and Processing, W3C Recommendation (Feb. 12, 2002). URL: http://www.w3.org/TR/xmldsig-core/
2) ISO/IEC 10918-1: Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines, 1994.
3) Bull, L., Stanski, P. and McG. Squire, D.: Content extraction signatures using XML digital signatures and custom transforms on-demand, International World Wide Web Conference, *Proc. 12th international conference on World Wide Web*, Budapest, Hungary, pp.170–177 (2003).
4) Bull, L., McG. Squire, D., Newmarch, J. and Zheng, Y.: Grouping Verifiable Content for Selective Disclosure, *8th Australasian Conference on Information Security and Privacy* (*ACISP 2003*), LNCS 2727, pp.1–12 (2003).
5) Steinfeld, R., Bull, L. and Zheng, Y.: Content Extraction Signatures, *International Conference on Information Security and Cryptography* (*ICISC 2001*), LNCS 2288, pp.285–304, Springer-Verlag, Berlin (2001).
6) Pennebaker, W.B. and Mitchell J.L.: *JPEG STILL IMAGE DATA COMPRESSION STANDARD*, Van Nostrand Reinhold (1992).
7) 行政機関の保有する情報の公開に関する法律. URL: http://www.soumu.go.jp/gyoukan/kanri/jyohokokai/
8) 個人情報の保護に関する法律. URL: http://www5.cao.go.jp/seikatsu/kojin/houritsu/index.html
9) 民間事業者等が行う書面の保存等における情報通信の技術の利用に関する法律. URL: http://www.cas.go.jp/jp/houan/

10) 経済産業省：文書の電磁的保存等に関する検討委員会 (2005).
11) 厚生労働省：医療情報システムの安全管理に関するガイドライン. URL: http://www.mhlw.go.jp/shingi/2005/03/s0331-8.html
12)（社）日本経済団体連合会情報通信委員会：税務書類の電子保存に関する報告書 (2004).
13) 武仲正彦，吉岡孝司，金谷延幸：検証者が署名者と墨塗り者を識別可能な電子文書の墨塗り方式，2004 年コンピュータセキュリティシンポジウム予稿集 (Oct. 2004).
14) 宮崎邦彦，洲崎誠一，岩村 充，松本 勉，佐々木良一，吉浦 裕：電子文書墨塗り問題，信学技法，ISEC2003-20, pp.61–67, 電子情報通信学会 (2003).
15) 宮崎邦彦，岩村 充，松本 勉，佐々木良一，吉浦 裕，手塚 悟，今井秀樹：開示条件を制御可能な電子文書墨塗り技術，2004 年暗号と情報セキュリティシンポジウム（SCIS 2004）予稿集，pp.515–520 (Jan. 2004).
16) 吉岡孝司，武仲正彦：電子文書の訂正・流通を考慮した部分完全性保証技術の提案，第 3 回情報科学技術フォーラム（FIT 2004），M-066 (Oct. 2004).
17) 吉岡孝司，武仲正彦：電子文書の訂正・流通を考慮した部分完全性保証方式の改良，2005 年暗号と情報セキュリティシンポジウム（SCIS 2005）予稿集 (Jan. 2005).

付 録

A.1 電子文書墨塗り技術

電子文書墨塗り技術には複数の方式が提案されているが，これらのうち基本的な方式の 1 つである “SUMI-4” の署名・墨塗り・検証処理は次のとおりである．

[SUMI-4]

- 電子文書 $M = (m_1, \ldots, m_n)$
  $(m_i \in \{0,1\}^*$:墨塗りブロック$)$
- 乱数 $r_i \in \{0,1\}^m \quad (i = 1, \ldots, n)$
- ハッシュ関数 $H : \{0,1\}^* \to \{0,1\}^n$
- 署名関数 $Sign() : \{0,1\}^* \to \{0,1\}^l$
- 検証関数 $Verify()$:
  $\{0,1\}^* \times \{0,1\}^l \to$ `Accept/Reject`
- 署名値 $s \in \{0,1\}^l$
- 墨塗り用署名 $S = (r_1, \ldots, r_n, s)$

署名・検証関数 $Sign(), Verify()$ は，それぞれ署名者の秘密鍵とそれに対応する公開鍵を用いた署名・検証を表す．また，$Sign(), Verify()$ には，DSA など既存の署名関数を使用することができる．

署 名

`input:` $M = (m_1, \ldots, m_n)$
`output:` $S = (r_1, \ldots, r_n, s)$

`for` i=1, ..., n
  $r_i \leftarrow \{0,1\}^m$, $d_i \leftarrow H(m_i||r_i)$
`end for`
$s \leftarrow Sign(d_0||\ldots||d_n)$
`Output` S

墨 塗 り

`input:` $M = (m_1, \ldots, m_n)$,
  $S = (r_1, \ldots, r_n, s)$
`output:` $M' = (m'_1, \ldots, m'_n)$,
  $S' = (r'_1, \ldots, r'_n, s)$

`for` i=1, ..., n
  `if` i^{th} `block = disclose`
    $r'_i \leftarrow r_i$, $m'_i \leftarrow m_i$
  `else if` i^{th} `block = close`
    $r'_i \leftarrow H(m_i||r_i)$, $m'_i \leftarrow$ `NULL`
  `end if`
`end for`
`Output` M' `and` S'

検 証

`input:` $M' = (m'_1, \ldots, m'_n)$,
  $S' = (r'_1, \ldots, r'_n, s)$
`output:` `Accept/Reject`

`for` i=1, ..., n
  `if` i^{th} `block = disclose`
    $d_i \leftarrow H(m'_i||r'_i)$
  `else if` i^{th} `block = close`
    $d_i \leftarrow r'_i$
  `end if`
`end for`
`Output result of` $Verify(d_1||\ldots||d_n, s)$

A.2 XML Signature [1)]

XML Signature は，W3C によって規格化されている XML を利用した電子署名のフォーマットである．XML Signature のフォーマットを**図 6** に示す．XML Signature の生成では，署名対象を URI などを用いて指定し，各署名対象ごとにハッシュ値の算出を行い，Reference 要素の生成を行い（Reference 生成），生成された Reference 要素に対して署名値の計算を行い，Signature 要素を生成する（Signature 生成）．なお，必要があれば各署名対象に対して Transform 要素を用いて，正規化などの変換を行うことができる．

XML Signature の検証では，各 Reference 要素の記述に従い，ハッシュ値の計算を行い，DigestValue

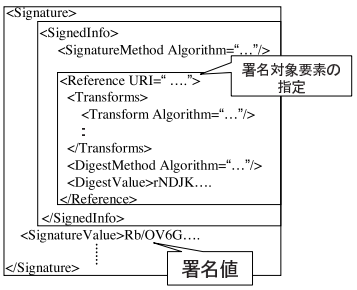


図 6 XML Signature
Fig. 6 XML Signature.

要素の値との比較を行い（Reference 検証），さらに SignedInfo 要素に対して SignatureValue 要素を用いて検証を行う（Signature 検証）．ここで，Reference 検証，Signature 検証のいずれかに失敗した場合には，検証失敗とする．

(平成 17 年 6 月 16 日受付)
(平成 18 年 1 月 6 日採録)

秦野　康生

1979 年埼玉県生．2004 年東京理科大学大学院理工学研究科修士課程修了．同年（株）日立製作所入社．現在に至るまで，同社システム開発研究所にて，暗号・情報セキュリティの研究に従事．2002 年暗号と情報セキュリティシンポジウム（SCIS2002）論文賞受賞．電子情報通信学会会員．

宮崎　邦彦（正会員）

1973 年神奈川県生．1998 年東京大学大学院数理科学研究科修士課程修了．同年（株）日立製作所入社．現在に至るまで，同社システム開発研究所にて，暗号・情報セキュリティの研究に従事．2004 年暗号と情報セキュリティシンポジウム（SCIS2004）論文賞受賞．電子情報通信学会会員．

手塚　　悟（正会員）

1984 年慶應義塾大学工学部数理工学科卒業．同年（株）日立製作所入社．マイクロエレクトロニクス機器開発研究所に勤務し，パーソナルコンピュータのオペレーティング・システム，デバイス・ドライバ，LAN システム等の研究開発に従事．その後，システム開発研究所に勤務．以来，パーソナルコンピュータを中心とした LAN システムの構築・運用管理の研究開発，さらにセキュリティシステムの研究開発に従事し，現在，システム開発研究所第七部部長．著書に『Inside CORBA』アスキー出版（共訳）（1998 年），『インターネットコマース—新動向と技術』（共立出版，共著）（2000 年）．