指紋を鍵とするファイル暗号化システムの開発

File encryption system using fingerprint as key

田島 英朗*, 鈴木 裕之**, 小尾 高史*, 山口 雅浩**, 大山 永昭**
Hideaki Tashima*, Hiroyuki Suzuki**, Takashi Obi*, Masahiro Yamaguchi** and Nagaaki Ohyama**
*東京工業大学大学院総合理工学研究科, **東京工業大学大学院像情報工学研究施設
*Interdisciplinary Grad. School of Sci. and Eng. Tokyo Tech., **Imag. Sci. and Eng. Lab. Tokyo Tech.

概要

従来の指紋を用いたファイル保護システムでは、認証によってファイル本体、もしくはファイルを暗号化する鍵へのアクセス制御を行っていた。しかしこの方法では、認証のための指紋情報がローカルに保存されるため、別の PC へファイルを移動して使用する場合などでは安全性を保つことが困難であった。一方我々の提案するシステムでは、指紋を暗号の鍵として用いた暗号化、復号化を行うことにより、認証情報をローカルに残さずにファイルを保護することが可能となった。

1. はじめに

近年、個人情報や機密情報などの重要な情報が、PC 本体や USB メモリのようなストレージの盗難、紛失など、管理上のミスにより漏洩してしまうという事件が頻繁に起こっている。対策としてファイルを暗号化していれば、ファイルを盗まれた際に中身を見られてしまう危険性はほとんど無くなるが、その際暗号化鍵をどのように管理するかがセキュリティ上重要な問題となる。ここで本人と直結した生体情報である指紋を暗号化、復号化の際の鍵として用いることが出来れば、鍵が本人と直結しているので管理の必要がなく安全かつ手軽に利用できると考える（図 1)。

しかし、指紋を暗号化鍵として用いようとした場合、たとえ本人の指紋であっても指紋センサーから指紋画像を取得するごとに、位置ずれや回転、ノイズなどの影響により少しずつ異なってしまうため、そのままでは 1 ビットの違いも許されない通常の暗号化処理には用いることが出来ない。鍵として用いる指紋画像がある程度変化しても同じ指紋からは同じ鍵を得られる手法が必要となる。

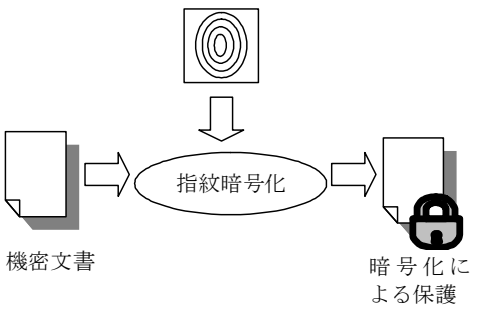

図 1 指紋暗号化による機密文書の保護

2. 基本原理

これまでに、Double Random Phase Encoding[1]と呼ばれる光学的暗号化手法（図 2）を利用して、指紋を暗号の鍵とする画像暗号化手法が提案されている[2]。この手法では、暗号化に用いる指紋画像と復号化に用いる指紋画像の類似度が、復号化画像に復元される原画像の復元精度を決定する。つまり、2 つの指紋画像の類似度が高いと原画像が正しく復元され、低いとランダムパターンが生成される。従って、暗号化したいデータのビットパターンを 2 次元画像にコーディングし、この画像を原画像として上記手法による暗号化、復号化を行えば、指紋画像がある程度変化しても正しい復号化結果が得られる暗号化手法が実現できる。

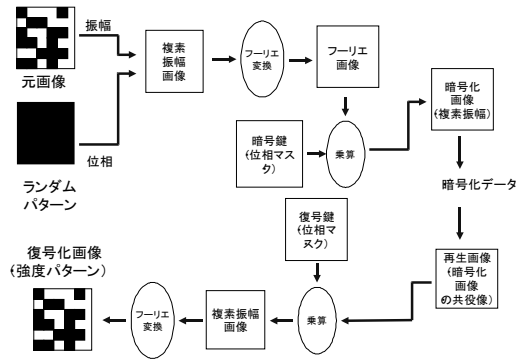

図 2 光学的暗号化手法

2.1. ビットパターン画像コーディング

バイナリデータを画像に変換する際、変換された画像の総濃度がバイナリデータの種類に依存して変化してしまうと、解読者へのヒントとなる可能性がある。そこで、ビットパターンを画像に変換する方式として、バイナリデータのすべてのビットを２つのドットで表現し、右が黒ならば“１”左が黒ならば“0”を表すとした（図 3）。この方式は、変換するビットパターンの長さが同じならば変換された画像の濃度が一定となる。例として“1234ABCD”という文字列データをビットパターンにしたものを示す（図 4）。

復号化画像からバイナリデータに復元するときには、まず復号化画像の中からビットパターンの部分を抽出し、各ビットを表す部分の左右のドットそれぞれの平均画素値を比較し、左が大きい場合には“１”、右が大きい場合には“0”としてバイナリ値を決定する。

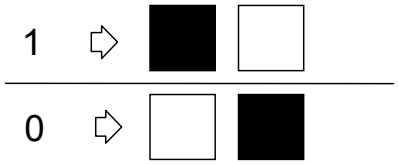

図 3 画像コーディング手法

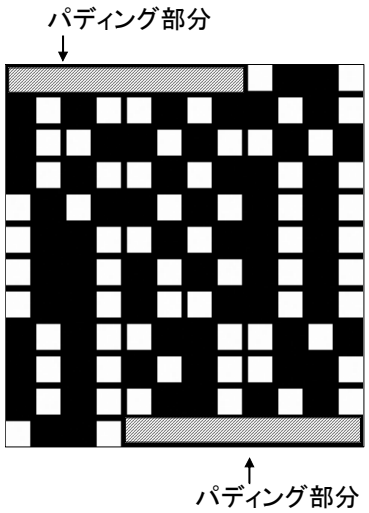

図 4 ビットパターン画像の例

2.2. ビットパターンの位置検出

暗号化時と復号化時の指紋に位置ずれがあると復元される画像にも位置ずれが現れる。そこで、ビットパターン画像に位置ずれ検出用タグを設置し（図 5)、復号化された画像中のタグを、タグのみを表示した画像との相関演算によって検出することにより、ビットパターンの正確な位置を検出する（図 6)。

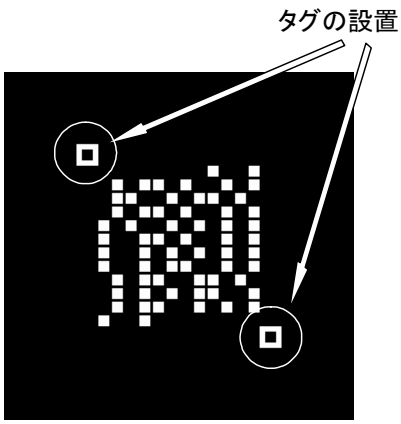

図 5 ピットパターンの位置検出用タグ

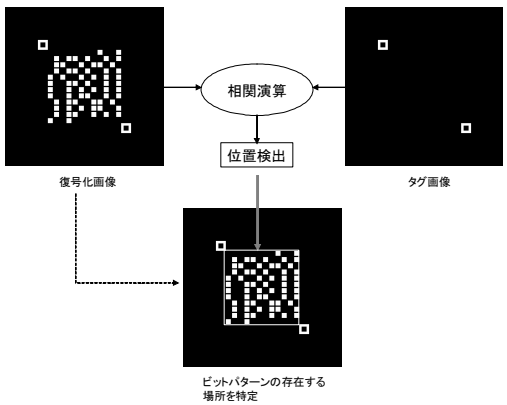

図 6 ビットパターンの位置検出

3. ファイル暗号化システム

前節で述べた暗号化手法は、画像の相関演算に基づく手法であるため、サイズの大きなファイルを処理する場合には遅延が大きい。よって本システムでは、指紋を鍵とする暗号化手法を直接ファイルの暗号化には適用せず、高速な暗号化処理が可能な共通鍵暗号方式（Triple DES）を用いてファイル本体を暗号化し、そのときに用いた Triple DES の鍵を、指紋を鍵とする暗号化手法により暗号化する 2 段階の暗号化方式を用いる（図 7)。また、指紋で暗号化された Triple DES の鍵は、Triple DES で暗号化されたファイルのデータにヘッダとして付加され、指紋によって暗号化されたファイル（指紋暗号ファイル）が作成される。

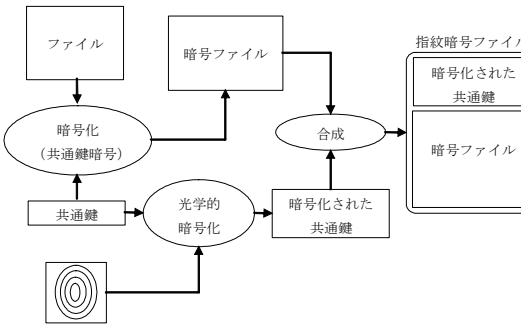

図 7 指紋を鍵とするファイル暗号化

復号化時には、まず指紋暗号ファイルから指紋によって暗号化された Triple DES の鍵と Triple DES によって暗号化されたファイルデータとを分離する（図 8)。そして暗号化されている Triple DES 鍵を、復号化用の指紋を用いて復号化し、Triple DES の鍵を取り出す。次に取り出した Triple DES の鍵を用いて暗号化されているファイルデータを復号化することにより元のファイルを得る。

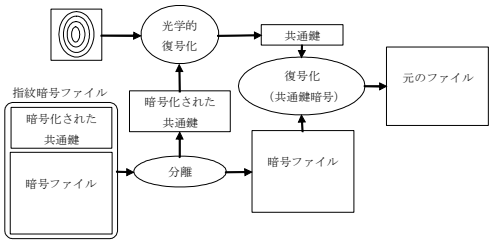

図 8 指紋暗号の復号化

4. 実装

今回、指紋リーダとして「DigitalPersona」社の「U.are.U」を用い、Windows 上のアプリケーションとして実装して、ファイルの暗号化および復号化が可能なことを確認した（図 9）。指紋を鍵とする暗号化手法では、指紋画像取得の状況により共通鍵の復元精度が劣化してしまうことがあるため、指紋の位置ずれ検出など復元精度向上のための機能も実装した。

図 9 暗号化ダイアログボックスのスクリーンショット

5. まとめ

本研究では、指紋を鍵として利用することが可能な暗号化手法を利用し、指紋でファイルを暗号化・復号化するシステムの開発を行った。本システムにより、暗号化鍵の管理を必要としないファイル保護が可能となった。

6. 参考文献

[1] P.Refregier and B.Javidi, “Optical encryption based on input plane Fourier plane random encoding,” Opt.Lett., **20**, 767-769 (1995).

[2] H. Suzuki, et al., “Fingerprint verification for smart-card holders based on optical image encryption scheme,” Proc. SPIE, **5202**, 88-96 (2003)

当開発作業は IPA の「未踏ソフトウェア創造事業（未踏ユース）」の採択案件として行い、IPA から開発補助を受けている。