Computer Security Symposium 2021
26 - 29 October 2021

QRコード化した身分証明書顔写真による本人認証システムの開発

中嶋　祥吾[1,a)]　瀧田　愼[2]　白石　善明[1]　森井　昌克[1]

概要：多くの身分証明書では顔写真と所属がクレジットカード大の板（用紙）に印刷され, 目視によって本人確認が行われている. 最近ではICカード自体に身分証明書を印刷することによって従来の証明書としての機能を踏襲するとともに, 機械認証による入退室管理等に用いられ, 利用者にとって敷居が低く, ユーザーフレンドリー（user-friendly）かつユーザビリティ (usability) に優れた認証, 身分照合方式となっている. しかしながらICカード自体の価格は低くなく, その認証装置自体も決して安価ではない. 本稿ではICカードによる非接触型無線通信によるデータ認証ではなく, 紙やプラスチック板に印刷されたQRコードによる認証方式を提案する. 固定化されたQRコードによる認証方式はすでに提案されているが, 従来からの身分証明書と併用されることを前提としている. 提案方式は顔写真による顔認証方式から生成されるPINへのデジタル署名を顔写真と一体化したQRコード自体に埋め込むところに新規性がある. 認証装置自体もスマートフォンのアプリで実現可能であり, また身分証明書での顔写真を張り替えてデジタル署名を回避することは困難であることから, 大きな抑止効果も期待できる.

キーワード：QRコード, デジタル署名, 身分証明書, 顔認証, 誤り訂正符号, 非組織符号化

Development of an identity authentication system using QR-coded ID cards and facial photographs

Shogo Nakajima[1,a)]　Makoto Takita[2]　Yoshiaki Shiraishi[1]　Masakatu Morii[1]

Abstract: In most ID cards, the photo are printed on a credit-card-sized board, and the identification is done visually. Recently, IC cards is printed with an identification card, which realize same function. It is a user-friendly and usable authentication and identity verification method. However, the price of the IC card itself isn't low, and the authentication device isn't inexpensive. Then, we propose an authentication method using QR codes printed on paper, instead of contactless wireless communication. Although QR code authentication method has already been proposed, it needs to be used with identification cards. The proposed method is novel in that the digital signature is embedded in the QR code, which is integrated with face. Since the authentication device can be realized by a application, and it is difficult to avoid the digital signature by replacing the face, a significant deterrent effect can be expected.

Keywords: QR code, digital signature, identification card, face recognition, error correction code, non-organizational coding

1. はじめに

現代において, 多くの企業は, 社員であるという証明を行う目的で社員証を発行している. これは, 本人の名前や所属等の個人情報と顔写真が付加されており, 顔写真と持ち

[1] 神戸大学
Kobe University
[2] 兵庫県立大学
University of Hyogo
[a)] nnakajima@stu.kobe-u.ac.jp

主が一致することで, 持ち主が社員であると証明が可能である. また, セキュリティー強化の観点から, 多くの社員証は IC チップを内蔵している. この IC チップはデジタル署名 [3] を格納しており, データが正規の作成者により作成されたもので, 改竄されていないことを証明することが可能である. これにより, 社員証の偽造や改竄が不可能であり, 安全性の高い社員証となる.

しかし, IC チップ入りの社員証には以下のようなデメリットが存在する.

- 社員証の作成に費用がかかる
- 情報の書き換えに専用の IC ライタが必要
- 顔写真は上から他人の写真に張り替えることが可能

また, 入退室管理だけに着目すれば, 顔認証システムの導入も考えられるが, そのコストだけでなく, 認証精度や認証に必要な時間も問題となり, 直接的な生体情報の利用もあって必ずしも, その障壁は低くない.

本稿では IC カードのように非接触型接触型無線通信によるデータ認証ではなく, 紙やプラスティック板に印刷された QR コードによる認証方式を提案する. 静的な QR コードから読み取ることができる ID 番号などによる認証方式は従来から存在するものの, 以下の問題点を有している.

- 静的な QR コードゆえに複製が可能である
- 身分証明書を仮定した場合, 改めて QR コードの印刷場所を確保しなければならない

これらの問題点を解決するために, 顔写真と一体化された QR コードを作成し, 格納情報と顔写真の不正を不可能にする. すなわち, 個々の顔写真での特徴点から抽出される, 理想的には終生不変, 万人不同の PIN に対してデジタル署名を施し, QR コードのデータとして格納する. 本提案の特徴の第一は, 身分証明書用写真に QR コードを重畳させることにある. 現実的には身分証明書の大きさはクレジットカード大（53.98x85.60mm サイズ標準）であり, QR コードの形状は正方形であることから, 30mm 四方から 40mm 四方となり, その中に写真を重畳させることになる. 氏名所属等の属性情報, 特徴量が確保できる鮮明な写真, およびデジタル署名を誤りなく読み取れる QR コードの作成を提案する. この QR コードにおいて, 写真を他人にすり替えてデジタル署名を無効にすることは困難であり, 複製しても本人の写真と利用する敵対者の顔が異なることから不正利用の困難性を確保できる. 何よりも従来からの身分証明書と仕様がほぼ同じであり, かつ顔認証に関わる生体情報を認証側で保存する必要がないため, 生体情報を扱うセキュリティ上の意識の不安を低減することが出来る.

なお, 本稿では身分証明書用の顔写真から特徴量を抽出する簡易的な方法を前提とし, その方法に基づく, 顔写真と一体化したデジタル署名付き QR コードを作成する. 本稿で与えた QR コードは通常のデコーダであれば属性情報を読み込むことが可能である. また, スマートフォン等のアプリとして開発可能な認証用の特殊なデコーダを用いて, デジタル署名を読み込み, 簡易的な顔認証による PIN と照合することによって本人確認が可能となる.

2. QR コードについて

QR コード (Quick Response Code)[1] とは二次元コードの 1 つであり, 従来の一次元バーコードに対して大きなデータ容量を持ち, 格納情報を符号化することで誤り訂正能力を有するという特徴を持つ. QR コードは開発された当初は製造や配送といった分野での仕様が想定されていたが, 現在は当初の分野に留まらない幅広い用途で活用されている. 特に, 近年の情報化社会の発展によって携帯電話やスマートフォンといった携帯情報端末の普及が進んでおり, これらの機器での読み取りを想定して雑誌や広告等に QR コードが記載される例も非常に多い. 本章では, QR コードの用語および構成の説明と, QR コードの訂正能力に利用されているリード・ソロモン符号 [2] に関して示す.

2.1 QR コードの関連用語

以下に QR コードにおける用語の説明を行う.

(1) 型番 (バージョン)
QR コード全体の大きさを示し, 1 から 40 までの値で設定される. 型番が大きくなると格納できる情報量も大きくなる.

(2) 誤り訂正レベル
QR コードは汚損が生じてモジュールの明暗が正しく読み取れなかった場合も, 一定の数だけ, 誤りを訂正してすることが出来る. これを誤り訂正能力と呼び, L, M, Q, H の 4 段階から設定できる.

(3) モジュール
QR コードのシンボルを構成する最小の単位セル. 1 モジュール当たり明暗いずれかの状態を示しており, これは 1 ビットの情報に相当する.

(4) データコード語
QR コードでの情報はコード語という単位で扱われる. 1 つのコード語は 8 つのモジュールから成り, これは RS 符号におけるシンボルに相当する. データコード語とは, QR コードに格納する情報が収められるコード語であり, RS 符号における情報点に相当する.

(5) 埋め草コード語
QR コードに格納する情報がデータコード語数に満たない場合, 余った空のコード語は一定のパターンの情報で埋められる. これを埋め草コード語と呼ぶ. 埋め草コードは単に容量を埋めるために用いられ, 有意な情報は持たない.

(6) 誤り訂正コード語
RS 符号の検査点に相当する情報が収められたコード語を示す. データコード語から一意に決定される. 符号

理論の検査点に相当する.

(7) 機能パターン
QR コードを光学的に読み込む際に必要な情報が記される領域であり, QR コードの位置検出や画像の歪み補正に用いられる.

(8) マスク処理
QR コードを読み取り易くするために行う処理. マスク処理パターンは 8 種類用意されており, その中で明モジュールと暗のモジュール数を均一化し, 画像の高速処理の障害となるパターンの発生が最も抑えられるマスクを採用する. マスク処理は, 符号化領域のビットパターンとマスク処理パターンを XOR する.

2.2 リード・ソロモン符号

リード・ソロモン符号 (以下, RS 符号) [2] とは, QR コードの格納情報を符号化する際に用いられる誤り訂正符号の一種である. 誤り訂正符号とは, 本来の情報に冗長な情報を付与することで符号語を生成する方法である. これにより符号語の一部が誤った場合でも, その誤りを検出し訂正することが可能となる. ここで, 符号語の中の本来の情報を情報点, 追加した冗長な情報を検査点と呼び, 情報点はデータコード語と埋め草コード語, 検査点は誤り訂正コード語に相当する. 符号語の長さが n, 情報点が k 個, 最小距離が d の符号を符号 (n, k, d) と呼ぶ. (n, k, d) のぞれぞれの値は QR コードの型番と誤り訂正コード語によって決められている. RS 符号の誤り訂正能力 t は

$$t = \left\lfloor \frac{n-k}{2} \right\rfloor$$

で与えられ, RS 符号は最大距離分離符号であるので最小距離は,

$$d = n - k + 1$$

で与えられる.

3. デジタル署名

デジタル署名 [3] は公開鍵暗号方式の一つである. デジタル署名を用いることで, メッセージを作成したのが確かにその送信者であることを証明でき, かつ送信者がそのメッセージを送信したことを第三者に対して否認できなくなる. また, メッセージが改竄されていないことの証明にも利用可能である.

デジタル署名は署名鍵が秘密鍵, 検証鍵が公開鍵であり, 送信者は署名鍵を用いて送信メッセージから署名を作成する. 受信者は検証鍵を用いて, 受け取ったメッセージと 署名の正当性を検証する. 署名鍵を持っているのは送信者だけであるので確かに送信者により作られたメッセージであることを証明でき, また, 検証鍵によって正当な署名であると証明された場合メッセージが改変されていないことも同様に証明できる.

3.1 デジタル署名の検証手順

以下に, 受け取るメッセージが間違いなく本人のものであるのかを証明する方法を示す.

step1 送信者は公開鍵と秘密鍵を生成する.
step2 送信者は, 受信者に公開鍵を公開する.
step3 受信者は, 送信者が生成した公開鍵を入手する.
step4 送信者が受信者に送付するデータを作成する.
step5 作成したデータとにハッシュ関数を使用してハッシュ値を算出する.
step6 そのハッシュ値を秘密鍵を使用して暗号化する.
step7 step4 で作成したデータと step6 で暗号化されたハッシュ値を, デジタル署名として送信する.
step8 受信者は, 暗号化されたハッシュ値を, 送信者から入手した公開鍵を使用して復号する.
step9 受信者は, 受信データをもとに, 送信者側と同じハッシュ関数を使用して, ハッシュ値を算出する.
step10 step8 で復号されたハッシュ値と, step9 で算出されたハッシュ値を比較して一致すれば, 正しいデータと判断する.

4. QR コードと画像の一体化

顔画像とそのデジタル署名の一体化, すなわち顔画像データと, そのデジタル署名を紐付けした QR コードを作成する. 具体的には個々の顔画像での特徴点から抽出される, 理想的には終生不変, 万人不同の PIN を用い, その PIN にデジタル署名を施し, QR コードのデータとして格納する. 本提案の特徴の第一は, 身分証明書用写真に QR コードを重畳させることにある. したがって, 最大でも 40?四方に印刷された QR コードに, 十分視認可能な顔写真あるいは身分証明書用の上半身写真を重畳させる. この分野はエステティック（Aesthetic）QR コードと呼ばれ従来から研究がなされている [4]-[7]. 本章では文献 [4][5] の手法の応用として, 顔写真自体を QR コードのモジュールとして認識させ, QR コードの誤り訂正能力と非組織符号化を利用して顔画像を QR コード全面に組み入れ, RS 符号化, すなわち QR コードとして認識させるとともに, デジタル署名に必要な 64 ビット以上のデータを ver.17 の QR コードとして格納する方法を与える.

4.1 誤り訂正能力を利用した手法

QR コードに格納されている情報は RS 符号であるので, 誤り訂正能力以内の情報の誤りであれば, 読み取り時に誤りが発生した場合, 正しい情報に復号することができる. これを利用して, 誤り訂正能力の範囲内であれば, QR コード上に画像を直接貼り付けた場合, QR コードの読み

取りが可能となる．この手法では, 機能パターンを除く任意の場所に画像を配置可能であり, 原理が単純であるため実装が容易である．一方, 誤り訂正能力を利用するため QR コード本来の誤り訂正能力が減少する．埋め込むことのできる画像の最大の大きさは誤り訂正能力に依存し, 誤り訂正レベル H の QR コードを用いることで最大で QR コード全体の約 30 %となる．

4.2 従来の QR コードの埋め草コード語を利用した手法

QR コードは型番と誤り訂正レベルによって格納可能な情報の大きさが決まる．格納する情報が最大格納容量より小さい場合, その差分領域は埋め草コード語と呼ばれ, 有意義な情報を持たず, 格納情報に影響を与えない．よって, この埋め草コード語に画像を格納することで, 理論的には誤りを発生させること無く QR コード上に画像を付加することが可能である．

しかし, QR コードの符号化領域は, 符号語は右下から順番に右詰めの配置となるため, 通常, QR コードの右側から順に, データコード語, 埋め草コード語, 誤り訂正コード語という配置になる．

これは, QR コードを構成する RS 符号は前半に情報点, 後半に検査点となる配置である組織符号であるためである．したがって, 組織符号である RS 符号で構成する QR コードの埋め草コード語に画像を付加する場合, 画像の位置の自由度は小さくなる．

4.3 非組織符号化の利用

既存の研究で提案された手法**???**で, RS 符号について非組織符号化を用いることで, QR コードへ画像を付加した場合のデザインの位置の自由度を高めることが可能である．非組織符号化とは符号化方法の一つである．組織符号化では, 符号長 n の符号語の前半に情報点部が k シンボル配置され, 後半に検査点部が $n-k$ シンボル配置される．一方で, 非組織符号化では符号長 n ビットの符号語において任意の k 箇所に情報点を配置し, 残った ${}_nC_k$ 個の箇所に検査点を配置する．非組織符号化を用いることで符号生成に ${}_nC_k$ の自由度を与えることが出来る．

4.3.1 非組織符号化された RS 符号の作成方法

非組織符号化の方法の一つとして, 誤り消失訂正 [2] を用いる方法がある．誤り消失訂正では, 事前に訂正するシンボル（消失点）を決めておき, そのシンボルが誤っていた場合, 誤り訂正を行う．これを利用することで, RS 符号でないシンボル列に対し, 消失点と指定したシンボルを変更することで, そのシンボル列を RS 符号化することが可能である．ここでは, n シンボルの任意のシンボル列から検査点に該当する $n-k$ シンボルを指定して誤り消失訂正を行うことで非組織符号化を実現する．以下に誤り消失訂正を用いた非組織符号化された RS 符号の作成方法を示す．

step1　作成したい RS 符号の符号長 n, 訂正可能シンボル数 t を決定する．検査シンボル数 p は $p=2t$ となる．

step2　検査点とするシンボルの位置を p 個決める, また, 符号長 n の任意のシンボル列を決定する．

step3　p 個の検査点の位置, t を入力として, step2 で得たシンボル列を誤り消失訂正を行う. 復号後, 得られたものが非組織符号化された RS 符号である．

4.3.2 非組織符号化を用いた QR コードと画像の一体化

非組織符号化を用いた QR コードへの画像の付加する方法**???**を示す．

step1　埋め込む画像データからピクセル毎の RGB 値を取得する．

step2　ピクセル毎に RGB 成分 (r,g,b) を取得し, 輝度値 y を求める．y は以下の式で求める．

$y = 0.299 \times r + 0.587 \times g + 0.114 \times b$

そして, $y \leq 110$ なら 0, $y > 110$ なら 1 として 2 値化を行う．

step3　画像データのピクセルを QR コードのモジュールの大きさに合わせる．

step4　モジュールの明暗状態からビット列 b_1 を取得する．

step5　b_1 と予め決定しておいたマスクパターンとの XOR 演算を行い, ビット列 b_1' を作成する．

step6　b_1' の先頭から順に格納データと置換し, 得られたビット列を c_1 とする．

step7　c_1 からコード語を作成し, 各 RS 符号に対応するコード語をそれぞれ取得する．

step8　c_2 の n 個のコード語の内, 検査点部・情報点部の両方を含む特定の p 個の検査点の位置を選択し, 非組織符号化により RS 符号の符号語を導出する．

step9　導出した符号語に対して, step5 と同種のマスクパターンを用いてマスク処理を行い生成されたビット列を QR コード上に配置する．

5. 顔写真と一体化したデジタル署名付き QR コード

本章では, QR コードを用いて従来の身分証明書として併用できる認証装置を提案する．この実現のために 4.3.2 を用いて QR コードと顔写真を一体化させることに加え, 格納情報と顔写真に対してデジタル署名を施した QR コードである idQR コードを作成する．またスマートフォンのアプリで, 格納情報からデジタル署名の検証まで行うアプリを作成する．

以下に, idQR コードが満たすべき要件を以下に示す．

- 64byte 以下のテキストデータである個人情報を読み取りが可能（従来の QR コードリーダーで読み取り可能）
- 持ち主の顔写真が付加されており, 視覚的に持ち主かどうか判断可能

- 専用のデコーダで読み取ることで, 正式な個人情報と顔写真である証明が可能
- 社員証等の身分証明書に付加することを想定するため, 4cm × 4cm 以下の大きさで紙に印刷して読み取り可能

5.1 idQR コードの作成手順

型番 17, 誤り訂正レベル M の時の idQR コードを作成する手順を以下に示す.

step1 本人の個人情報と顔写真を入力とする. また, QR コード上の顔写真上の顔の位置と埋め草コード語に格納するデジタル署名の位置を任意に決定する.

step2 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの白色の正方形を作る.

step3 この正方形上に顔が step1 で決定した顔の位置に配置されるように顔写真を付加し, 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの顔写真を作成する.

step4 顔検出ツールを用いて, 顔領域を正方形領域で検出する.

また, f を顔領域にある 1 つ目の RS 符号のシンボル数とし, t を誤り訂正可能シンボル数とする. step3 で得られた正方形上の顔写真が以下の式を満たさない時, 顔写真の大きさを小さくして, step3 に戻る. 以下の式を満たした場合の 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの顔写真を I_1 とする.

$$f \leq t - 1 \tag{1}$$

step5 （QR コードのデータコード語）と（顔写真による顔認証方式から生成される PIN）を連結させたデータからデジタル署名を作成する.

step6 顔写真 I_1 からピクセル毎の RGB 成分 (r, g, b) を取得する.

step7 顔写真 I_1 の 1 モジュールに対応する 9 × 9 ピクセルを, その中央の 1 ピクセルに置換し, 1 モジュール= 1 ピクセルの大きさの正方形上に付加された顔写真 I_2 を取得する.

step8 顔写真 I_2 の全てのピクセルについて, RGB 成分 (r, g, b) から輝度値 y を求める.

y は以下の式で求める.

$y = 0.299 \times r + 0.587 \times g + 0.114 \times b$

そして, $y \leq 110$ なら 0, $y > 110$ なら 1 として 2 値化を行う.

step9 モジュールの明暗状態からビット列 b_1 を取得する.

step10 b_1 と予め決定しておいたマスクパターンとの XOR 演算を行い, ビット列 b_1' を作成する.

step11 b_1' の先頭から順に格納データと置換し, あらかじめ決めた座標に該当するシンボルをデジタル署名と置換する. 得られたビット列を c_1 とする.

step12 c_1 からコード語を作成し, 各 RS 符号に対応するコード語をそれぞれ取得する.

step13 step14 で得たそれぞれのコード語を 4.3.1 の方法で RS 符号化する.

step14 導出した符号語に対して, step10 と同種のマスクパターンを用いてマスク処理を行い生成されたビット列を QR コード上に配置する. 得られた QR コードを Q とする.

step15 顔写真 I_1 の 9 × 9 ピクセルで構成されるモジュールに対応する QR コード Q のモジュールの黒 or 白を取得する.

そして, I_1 のモジュールをカメラで読み取った時に, 高確率で上で得た黒 or 白と認識するように, 以下の条件 1〜4 に従って, I_1 上にモジュールまたはドットを付加する.

そして得られた QR コード Q と認識する顔写真 I_1 が idQR コードである.

条件 1 顔領域はモジュールやドットを付加しない

条件 2 QR コードの 4 辺から 9 モジュール幅の領域にモジュールを付加する

条件 3 情報点全てがデータコード語である RS 符号のモジュールに対して, モジュールの中央 7 × 7 ピクセルにドットを付加する

条件 4 その他のモジュールに対して, モジュールの中 5 × 5 ピクセルにドットを付加する

5.2 idQR コードの読み取り, 認証手順

idQR コードをカメラで読み取り, 認証を行う手順を示す.

step1 カメラから QR コードを撮影する.

step2 従来の QR コードと同様に, QR コードを取得し, 個人情報のテキストデータを取得する. また, その過程で得た誤り訂正後のコード語から, デジタル署名を得る.

step3 idQR コード上の顔写真を用いて, 顔認証方式から生成される PIN を取得する.

step4 step1 で得た個人情報, デジタル署名と step2 で得た顔データから, デジタル署名の検証を行う. reject された場合は, step1 に戻る.

5.3 idQR コードのまとめ

5.1 では, 従来の QR コードとして読み取りが可能とするために step1 4 を行う. step5 以降では, RS 符号において非組織符号化し, 埋め草コード部の任意のシンボルに情報シンボルと検査シンボルを配置できることを利用し, QR コードと顔写真の一体化とデジタル署名の格納を行う. このようにして作成した idQR コードを図 1 に示す.

図 1 idQR コード

また, 5.2 では, 従来の QR コードリーダーの機能に加えて, デジタル署名の検証を行う QR コードリーダの作成方法について述べた.

6. 顔認証方式を利用した具体的な idQR コードの構成

本来は身分証明書用写真から個人の特徴量である識別可能な 256 ビットの PIN を抽出し, その PIN に対してデジタル署名を施す. ここでは顔写真にデジタル署名を施した簡便な方式により作成した idQR コードについて述べる. この方式では, 写真データの読みこみ精度と誤り訂正能力の関係から他人の顔写真を見逃すためには大幅に顔写真を修整する必要がある. この方式によって作成した idQR コードを図 2 に示す.

6.1 作成手順

簡便な方式における idQR コードを作成する手順を以下に示す.

step1 本人の個人情報と顔写真を入力とする. また, QR コード上の顔写真上の顔の位置と埋め草コード語に格納するデジタル署名と署名検証用の検査点の位置を任意に決定する.

step2 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの白色の正方形を作る.

step3 この正方形上に顔が step1 で決定した顔の位置に配置されるように顔写真を付加し, 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの顔写真を作成する.

step4 顔検出ツールを用いて, 顔領域を正方形領域で検出する.
また, f を顔領域にある 1 つ目の RS 符号のシンボル数とし, t を誤り訂正可能シンボル数とする. step3 で得られた正方形上の顔写真が以下の式を満たさない時, 顔写真の大きさを小さくして, step3 に戻る. 以下の式を満たした場合の 1 モジュール= 9 × 9 ピクセルの QR コードと同じサイズの顔写真を I_1 とする.

$$f \leq t - 1 \tag{2}$$

step5 顔写真 I_1 の顔の肌の輝度値の平均が 200 となるように, I_1 の各ピクセルに対してガンマ補正を行う. ただし, ガンマ補正は以下の式に従う. 顔写真 I_1 の肌の輝度値の平均が 200 となる γ を求め, その γ を用いて, 全てのピクセルに対してガンマ補正を行う.

$$Y = 255 \times (Y \div 255)^{1/\gamma} \tag{3}$$

step6 顔写真 I_1 の各モジュールの中央 5 × 5 ピクセルに対して, 以下のように疑似的な 4 値化を行う. ただし, ピクセルの RGB 成分 (r, g, b), 輝度値 y とすると,
y=0.299 × r+0.587 × g+0.114 × b

- y が 0〜60 のピクセルは, 輝度 0 とする
- y が 61〜150 のピクセルは, 輝度 150 とする
- y が 150〜180 のピクセルは, 輝度 180 とする
- y が 181〜210 のピクセルは, 輝度 210 とする
- y が 211 以上のピクセルについては, 変更しない

step7 6.3 に従って, デジタル署名と署名検証用の検査点を取得する.

step6 顔写真 I_1 からピクセル毎の RGB 成分 (r, g, b) を取得する.

step8 顔写真 I_1 の 1 モジュールに対応する 9 × 9 ピクセルを, その中央の 1 ピクセルに置換し, 1 モジュール= 1 ピクセルの大きさの正方形上に付加された顔写真 I_2 を取得する.

step9 顔写真 I_2 の全てのピクセルについて, RGB 成分 (r, g, b) から輝度値 y を求める.
y は以下の式で求める.
$y = 0.299 \times r + 0.587 \times g + 0.114 \times b$
そして, $y \leq 110$ なら 0, $y > 110$ なら 1 として 2 値化を行う.

step10 モジュールの明暗状態からビット列 b_1 を取得する.

step11 b_1 と予め決定しておいたマスクパターンとの XOR 演算を行い, ビット列 b_1' を作成する.

step12 b_1' の先頭から順に格納データと置換し, あらかじめ決め座標に該当するシンボルをデジタル署名と置換する. 得られたビット列を c_1 とする.

step13 c_1 からコード語を作成し, 各 RS 符号に対応するコード語をそれぞれ取得する.

step14 step14 で得たそれぞれのコード語を 4.3.1 の方法で RS 符号化する.

step15 導出した符号語に対して, step10 と同種のマスクパターンを用いてマスク処理を行い生成されたビット列を QR コード上に配置する. 得られた QR コードを Q とする.

step16 顔写真 I_1 の 9 × 9 ピクセルで構成されるモジュールに対応する QR コード Q のモジュールの黒 or 白を取得する.
そして, I_1 のモジュールをカメラで読み取った時に, 高確率で上で得た黒 or 白と認識するように, 以下の条件

1〜4 に従って, I_1 上にモジュールまたはドットを付加する.

そして得られた QR コード Q と認識する顔写真 I_1 が idQR コードである.

条件 1　顔領域はモジュールやドットを付加しない

条件 2　QR コードの 4 辺から 9 モジュール幅の領域にモジュールを付加する

条件 3　情報点全てがデータコード語である RS 符号のモジュールに対して, モジュールの中央 7 × 7 ピクセルにドットを付加する

条件 4　その他のモジュールに対して, モジュールの中 5 × 5 ピクセルにドットを付加する

6.2　読み取り, 認証手順

簡便な方式における idQR コードをカメラで読み取り, 認証を行う手順を示す.

step1　カメラから QR コードを撮影する.

step2　従来の QR コードと同様に, QR コードを取得し, 個人情報のテキストデータを取得する. また, その過程で得た誤り訂正後のコード語から, デジタル署名と署名検証用の検査点を得る.

step3　idQR コード上の顔領域から顔データを取得する. 顔領域上 728 モジュールの中央のピクセルの輝度値を取得していき, 長さ 728 の輝度値の列を取得する.

step4　長さ 728 の輝度値それぞれを閾値 t_0, t_1, t_2 として 4 値化し, シンボル列を取得する. ただし, 4 値化時の閾値 t_0, t_1, t_2 は, 撮影した idQR コードの平均の輝度値を用いて以下のように求める.

t_0=（平均の輝度値）−80, t_1=（平均の輝度値）−50, t_2=（平均の輝度値）−10 とする.

step5　step2 で得た検査点を用いて, 顔データの誤り訂正を行い, カメラでの読み取り誤りのない顔データを取得する.

顔データの誤り訂正に失敗した場合は, step1 に戻る.

step6　step1 で得た個人情報, デジタル署名と step2 で得た顔データから, デジタル署名の検証を行う. reject された場合は, step1 に戻る.

6.3　デジタル署名の作成方法

デジタル署名は個人情報と顔写真から得られる顔データから作成する. 顔データは, 顔写真の顔領域内のモジュールの中央の輝度を取得し, その輝度を使用して求めるが, 読み取り時に雑音が発生する. そこで, 顔データは誤り訂正符号である 4 元 BCH 符号 $(n, k, d) = (1256, 728, 101)[2]$ で構成し, idQR コード作成時にその検査点（署名検証用の検査点）を埋め草コード語に格納する.

以下に, デジタル署名の作成手順を示す.

step1　従来の QR コードと同様に読み取るデータを求める.

step2　顔写真上の顔の領域に位置する 728 モジュールの中央の輝度値を取得する.

step3　step2 で得た輝度値を以下に従って 4 値化する. 得られた値を並べたシンボル列を顔データとする.

- 得た輝度値が 0 のピクセルを 0 とする
- 得た輝度値が 60 のピクセルを 1 とする
- 得た輝度値が 150 のピクセルを 2 とする
- 得た輝度値が 210 以上のピクセルを 3 とする

step4　（QR コードのデータコード語）と（顔データ）を連結させたデータからデジタル署名を作成する.

6.4　提案した idQR コードの評価実験

本章では 6 人の顔写真から作成した 6 つの idQR コードを用いてカメラでの読み取り精度と安全性の評価についての評価実験を行った. ただし, 検証を行った idQR コードは簡便な方式 6 で作成した idQR コードである.

6.4.1　実験環境

pc 上で作成した idQR コードを 4cm × 4cm の大きさで印刷し, 専用のアプリケーションをインストールした端末で読み取る. 端末とプリンターは以下の通りである.

- 端末：モトローラ・モビリティ・ジャパン moto Moto G5S
  - Android version:8
  - カメラ（外側：約 1600 万画素, 内側：約 500 万画素）
- プリンター：Canon MG3630（解像度：4800 × 1200）

6.4.2　読み取りから検証までの時間

以下の表 1 に idQR コードを 5 回読み取りを試行した場合の読み取りにかかる時間を示す. ただし, ここで測定した時間は, 以下にかかる時間の合計時間である.

1　カメラの起動までの時間

2　カメラで idQR コードをかざし, ピントがあうまでの時間

3　idQR コードを従来の QR コードとしてデコードする時間

4　デジタル署名の検証にかかる時間

6.4.3　安全性の検証

idQR コードは読み取り時の雑音が発生するため, デジタ

図 2　顔認証方式を利用した具体的な idQR コード

表 1 idQR コードの読み取りにかかる時間 [s]

撮影	1 回目	2 回目	3 回目	4 回目	5 回目	平均
No.0	9	4	10	3	4	6
No.1	4	5	4	3	4	4
No.2	7	6	5	2	3	4.6
No.3	6	4	6	5	6	5.4
No.4	10	8	7	3	4	6.4
No.5	5	12	8	5	10	5

表 2 (a)〜(e) を読み取った時の No.i との最小の距離

i	a	b	c	d	e
0	281	296	238	279	293
1	270	200	197	295	179
2	269	194	216	307	212
3	209	183	249	281	158
4	263	286	304	244	259
5	292	197	222	221	286

ル署名の検証の際に使用する顔データに対する誤り訂正を行う. この時の誤り訂正能力が大きい場合, 他人の顔に改竄された idQR コードであっても, 誤り訂正してしまい, デジタル署名が accept されてしまう恐れがある. この点について, 安全性の評価を行う. ここでは, 顔データは, 4 元 BCH 符号 $(n, k, d) = (1256, 728, 101)$[2] で構成され, 顔データのデコード時に 50 シンボルまで訂正可能である.

ゆえに, 正式な idQR コード上に, 他人の顔写真を貼った場合でも, 正式な顔データとの距離が 50 シンボル以内の顔データを取得できた場合, デジタル署名は accept される.

これが可能であるか検証する手順を以下に示す.

step1 i=0 とする.

step2 正式な idQR コードである No.i の顔写真の上に, 敵 5 人の顔写真を, 左目の位置を合わせて張り替え, 改竄された idQR コード No.i である (a) (e) を作成する.

step3 (a)〜(e) の顔データの読み取り, 正式な顔データとの距離を測定する試行を 10 回行い, 10 回測定した中で最小の距離を記録する.

step4 i=0〜5 で同様の作業を行う.

以下の表 2 に, 上の結果を示す.

正式な No.i との距離から 50 シンボル以上の場合, 顔データは誤り訂正不可能であるため, 表 2 より, idQR コードは安全である.

7. まとめ

本稿ではクレジットカード大の身分証明書に印刷された QR コードによる認証を行う方法として, 顔写真と一体化したデジタル署名付き QR コードを構成した. 特に特徴量が確保できる鮮明な写真と, 氏名所属等の属性情報, およびデジタル署名を誤りなく読み取れる QR コードとなっており, 属性情報はスマートフォンに搭載されている標準の QR コードデコーダで読み取り可能であり, デジタル署名での認証は, アプリとして開発可能である. この QR コードにおいて, 写真を他人にすり替えてデジタル署名を無効にすることは困難であり, 複製しても本人の写真と利用する敵対者の顔が異なることから不正利用の困難性を確保できる. 何よりも従来からの身分証明書と仕様がほぼ同じであり, かつ顔認証に関わる生体情報を認証側で保存する必要がないため, 生体情報を扱うセキュリティ上の意識の不安を低減することが出来る.

謝辞 本研究の一部は JSPS 科研費 20K11810 の助成を受けたものである. また研究にご協力いただいたセイコーソリューションズ（株）の方々に謝意を表する.

参考文献

[1] 日本工業規格, JIS X 0510:2004 二次元コードシンボル-QR コード-基本仕様, 日本産業標準調査会, https://kikakurui.com/x0/X0510-2004-01.html, 文書年月 2004

[2] 今井秀樹, 宮川洋, 岩垂 好裕, ”符号理論”, 社団法人電子情報通信学会, 株式会社コロナ社, 東京, 1990.

[3] 黒沢馨, 尾形わかは, ”現代暗号の基礎数理”, 社団法人電子情報通信学会, 株式会社コロナ社, 東京, 2009

[4] M. Kuribayashi and M. Morii, Enrichment of visual appearance of aesthetic QR code, Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics), vol. 9569, pp. 220-231, 2016.

[5] M. Kuribayashi, and M. Morii, “Aesthetic QR code based on modified systematic encoding function, ” IEICE Trans. Info. Syst., vol. E100-D, no. 1, pp. 42?51, 2017

[6] M. Xu, Q. Li, J. Niu, S. Hao, X. Liu, W. Xu, P. Lv, and B. Zhou, “Art-up: A novel method for generating scanning-robust aesthetic QR codes, ” ACM Transactions on Multimedia Computing, Communications, and Applications, 2020.

[7] M. Xu, H. Su, Y. Li, X. Li, J. Liao, J. Niu, P. Lv, and B. Zhou, “Stylized 7aesthetic QR code, ” IEEE Transactions on Multimedia, vol. 21, no. 8, pp. 1960?1970, 2019.