Vol.2020-DPS-182 No.5
Vol.2020-CSEC-88 No.5
2020/3/12

誤り訂正符号に基づく偽装QRコードの検出手法の提案

大東　俊博[1]　川口　宗也[1]　小林　海[1]　木村　隼人[1]　鈴木　達也[1]　岡部　大地[1]　石橋　拓哉[1]
山本　宙[1]　乾　真季[2]　宮本　亮[2]　古川　和快[2]　伊豆　哲也[2]

概要：2018 年 6 月の ICSS 研究会にて大熊らは確率的に悪性サイトに誘導可能な QR コードの構成法を示した．この方法は QR コードに使われている誤り訂正符号の性質に注目しており，復号誤りが生じやすいように改変した QR コードに対してカメラによる白黒検出の誤りを誘発する汚れを付加することで確率的な挙動の変化を実現している．SCIS2019 で瀧田らは現実的な脅威とするためには悪性サイトへの誘導が低確率にする必要がある点に注目し，複数回の読み込みを試行することで悪性サイトへの誘導が生じたことを検出する対策方法を示している．この方法はユーザが自衛をする点では有効であるが，掲示された QR コードが偽装 QR コードであるか否かの判定をすることができず，脅威が継続してしまう課題がある．そこで本稿では，偽装 QR コードの攻撃手法の性質に注目した検出手法を提案し，その方法を利用することで掲示された QR コードが偽装 QR コードであるか否かの判定を可能とする．

Detection Methods for Fake QR Codes Based on Error-Correcting Codes

Toshihiro Ohigashi[1]　Shuya Kawaguchi[1]　Kai Kobayashi[1]　Hayato Kimura[1]　Tatsuya Suzuki[1]
Daichi Okabe[1]　Takuya Ishibashi[1]　Hiroshi Yamamoto[1]　Maki Inui[2]　Ryo Miyamoto[2]
Kazuyoshi Furukawa[2]　Tetsuya Izu[2]

1. はじめに

QR コード[*1]は 1994 年に株式会社デンソーによって開発された二次元バーコードであり，一次元バーコードと比べて比較的大容量のデータを扱えることやカメラを搭載している携帯端末等で容易に利用できることから普及が進んでいる．QR コード用途として LINE Pay，PayPay など決済サービスや ANA の SKIP サービスのような航空券の代替，URL を埋め込むことによる Web ページへのアクセスの簡単化などが挙げられる．

QR コードはカメラ付き携帯端末等の機械から読み取りやすいフォーマットではあるが，人間の目でどのようなデータが含まれているかを即座に確認することは一般的に難しく，不正なデータを埋め込まれて読み込んでしまうリスクが潜在的に生じている．例えば，URL を読み込んだ際に自動的にそのサイトに移動するような実装であれば，詐欺サイトやマルウェアへ感染するような悪性サイトへアクセスさせてしまえる．しかしながら，常に悪性サイトへアクセスさせる QR コードであれば，被害が生じた際にそれを発見することは比較的容易であることから，その QR コードが印刷された掲示を取り除くなどの対策は容易である．

2018 年 6 月の ICSS 研究会，FIT2018 および CANDAR 2018 にて大熊らは確率的に悪性サイトに誘導可能な QR コード（偽装 QR コード）の存在と具体的な構成法を示した [1], [2], [3]．この方法は QR コードに使われている誤り訂正符号の性質に注目しており，復号誤りが生じやすいように改変した QR コードに対してカメラによる白黒検出の誤りを誘発する汚れ（QR コードの対象のモジュールの中心のピクセルの色を変更する）を付加することで確率的に異なる URL のサイトへ誘導することが可能となる．さらに，確率的な誤りを誘発する汚れの輝度を変更することで

[1] 東海大学，Tokai University
[2] 富士通研究所，Fujitsu Laboratories

[*1] http://www.qrcode.com/index.html

復号誤りが生じる確率を変化させる実験をしており，悪性サイトへ誘導できる確率が 0.6% のように極端に小さくできたことを報告している．この方法によって作成された偽装 QR コードは，通常時には正規のサイトにアクセスさせ，低い確率で悪性サイトへアクセスさせることが可能になるため，そのような QR コードが印刷された掲示に気がつかずに放置されたままになると予想される．このとき，被害者が悪性サイトに誘導されて被害を受けたとしても，再現性の低さからどの掲示物で被害にあったかの判別も難しくなるなど，不正行為が高度化する可能性がある．

文献 [1], [2], [3] の方法は QR コードのモジュールの中心に目立つ改変が生じることから，比較的目立たない改変によって偽装 QR コードが構成可能かの検討もされている．CSS2018 で大熊らはカラフルな画像や文字が含まれている QR コードであるデザイン QR コード[*2]をベースにして偽装 QR コードを作ることにより，確率的な誤りを誘発させるための汚れに気づきにくくする方法を示している [4]．さらに，瀧田らは SCIS2019 にて QR コードのモジュールに汚れを付加するのではなく，モジュールの位置をずらすことで QR コードのデコーダプログラムで白黒判定を行う際の走査線の位置のずれによって確率的な誤りを生じさせる方法を示している [5].

SCIS2019 で瀧田らは現実的な脅威とするためには悪性サイトへの誘導を低確率にする必要がある点に注目して，カメラにより QR コードの読み取りを複数回試行し，得られる情報が変化しない場合に正規のサイトと判定する対策を示している [5]．この方法は数回程度の読み取りでは処理時間への影響はほとんどないことから，各ユーザが悪性サイトへ誘導されないように自衛するという点で有望な対策方法と言える．しかしながら，この方法では正規サイトにアクセスした場合には掲示された QR コードが偽装 QR コードか否かを判定することはできないため，偽装 QR コードが掲示されたままになり，未対策の携帯端末に対する脅威が継続してしまう課題がある．

そこで本稿では，偽装 QR コードの構成法の原理に注目した方法により，掲示されている QR コードが偽装 QR コードか否かを判定する検出手法を提案する．具体的には，(1) 偽装 QR コードで効率的に悪性サイトに誘導するためには正規サイトにアクセスする場合でも誤り訂正符号によって訂正される誤りの個数が大きくなることに注目した方法（誤り数に注目した検出手法），(2) 偽装 QR コードの攻撃手法では誤りが生じる位置が誤り訂正ブロックに集中することに注目した方法（誤り位置に注目した検出手法），(3) 印刷用紙の破れ・マジックなどによる汚れや影などで自然に誤りが生じる場合と偽装 QR コードで人為的に誤らせる場合の誤り方の違いに注目した方法（誤り方に注目した検出手法）の 3 つの観点から検出手法を示す．

[*2] https://www.unitag.io/qrcode

2. QR コード

QR コードは 1 ビットの情報を示すモジュールと呼ばれる最小単位のセルを 2 次元に配置し，情報の埋め込み・読み込みを行う．配置できるモジュールの数は型番（version）によって指定でき，例えば version 2 では 25x25 モジュールを配置できる．また，QR コードは汚れ等があったときにも正確に読み取れるようにするために誤り訂正符号を使っており，その誤り訂正レベルを L，M，Q などで指定する．QR コードは数字データ，英数字データ，8 ビットバイトデータ，漢字データなどが扱えるが，どの種類のデータであるかはモード指示子と呼ばれる 4 ビットの識別子で指定する．例として，8 ビットバイトデータ（モード識別子は 2 進数で 0100）で 2-M 型（version 2，誤り訂正レベル M）の QR コードのデータフォーマットを以下に示す．

- モード指示子（4 ビット）: $(0100)_2$
- 文字数指示子（8 ビット）: データ文字列の長さを 2 進数で指定
- データ文字列（最大 26 バイト）: 文字列指示子で指定したサイズのデータ本体
- 終端パターン（4 ビット）: 0 でパティング
- 埋め草コード: データ文字列のバイト数が最大値である 26 バイトに満たない場合に特定のパターン（236, 17 の繰り返し）でパディング
- 誤り訂正ブロック（16 バイト）: 上記の 5 つのデータを用いて RS 符号で計算した冗長ビット

最初の 5 つのデータ（28 バイト分）を D1〜D28，誤り訂正ブロックを E1〜E16 としたときの 2-M 型の QR コードのコード配置を図 1 に示す．各バイトに 8 つのモジュール（ビット）が確保されていることがわかる．また，QR コードを撮影時に向きを合わせたり，モジュールの走査線を得るために，QR コードでは上記のデータ以外に「位置検出パターン」や「位置合わせパターン」という特定のパターンを配置している．

QR コードでは Reed-Solomon(RS) 符号 [6] を用いて誤りが生じた際に正しいデータを復元する．RS 符号は符号語（冗長ビットを付加された系列）をビットの集まり（シンボル）単位で表わし，シンボル単位で誤り訂正を行うため連続して起こるビット単位の誤り（バースト誤り）に強いという特徴を持つ．QR コードでは 1 バイトを 1 シンボルとして符号化を行っており，図 1 のように連続したビットを隣接したモジュールに配置することでマジック等で連続した複数のモジュールが汚れた場合でも少数のシンボルの誤り訂正で対応できる．符号長 n と情報点数 k とした (n, k) RS 符号において，ユークリッド復号法 [6] を用いて訂正可能な誤りの数は設計距離 $d = n - k + 1$ により保証されており，訂正可能な誤りシンボルの数は $t = \lfloor \frac{d-1}{2} \rfloor = \lfloor \frac{n-k}{2} \rfloor$ で

Vol.2020-DPS-182 No.5
Vol.2020-CSEC-88 No.5
2020/3/12

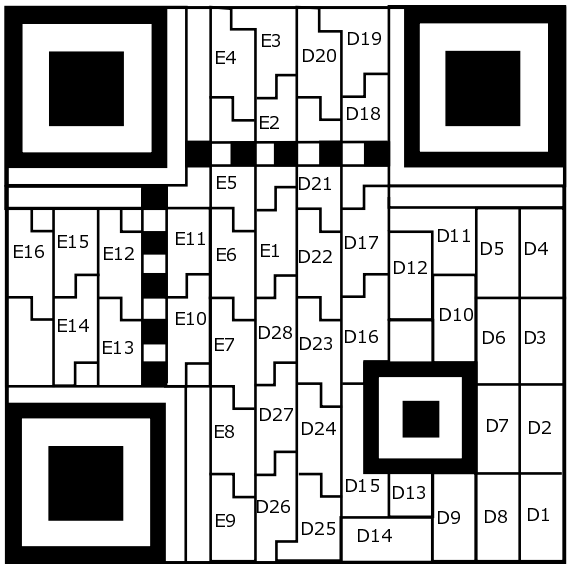

図 1　QR コード（2-M 型）のコード配置

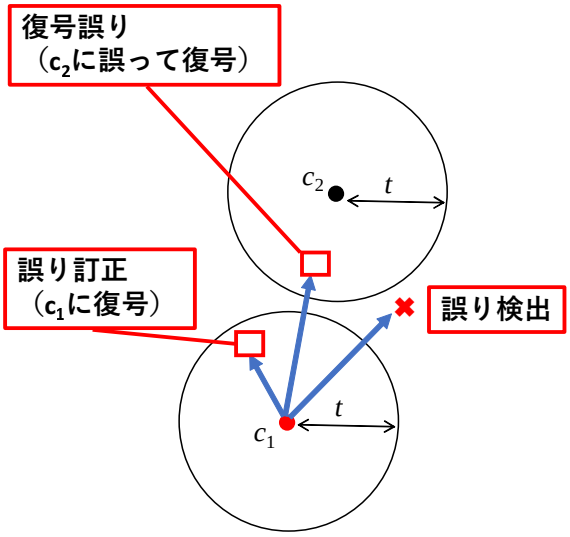

図 2　誤り訂正，誤り検出，復号誤り

与えられる．QR コードでは誤り訂正ブロック以外のデータのサイズが k バイト，誤り訂正ブロックを含んだ全体のサイズが n バイトという対応になり，2-M 型では $n = 44$，$k = 28$，$t = 8$ となる．誤り訂正能力 t の符号を用いた場合，図 2 のように t シンボルまでの誤りまで訂正可能であり，t シンボルを超えた誤りが生じたときには誤り検出または復号誤りが生じる．t シンボルを超えて誤ったときに別の符号語の t シンボル以下の誤った受信後になっていた場合にはその受信語に誤って復号されてしまい（復号誤り），どの符号語にも復号できない受信語になっている場合には誤り検出は行えるが復号はできない．

3. 確率的に悪性サイトへ誘導可能な偽装 QR コードとその対策

本章では確率的に悪性サイトへ誘導可能な偽装 QR コードの原理を示し，さらに瀧田らが示した悪性 URL へ誘導されないための対策方法について説明する [5].

3.1 確率的に悪性サイトへ誘導可能な偽装 QR コード

3.1.1 原理

大熊らは復号誤りを誘発する QR コード（偽装 QR コード）によって確率的に悪性サイトへアクセスさせる方法を示している [1], [2], [3].

正規のサイトの URL を含んだ QR コード用の符号語を c_1，ある悪性サイトの URL を含んだ QR コード用の符号語を c_2，二つの符号語の距離（異なるシンボルの個数）を $d(c_1, c_2)$ とする．このとき，まず初めに c_1 よりも c_2 との距離が $a(\leq t)$ 小さくなるように a 個の誤りを付加した系列 c_1' を作成する．系列 c_1' から作成された QR コードは誤りに対する耐性が低くなるが，t 個以下の誤りしか含んでいないため撮影環境が良好であれば正規のサイトの URL が復号される．さらに系列 c_1' よりも c_2 との距離が b 小さくなるように b 個の誤りを更に付加して系列 c_1'' を作成する．ここで，b は $d(c_1, c_2) \geq a + b \geq d(c_1, c_2) - t$ を満たすとする．このとき，系列 c_1'' は誤り訂正によって c_2 に復号される（復号誤り）ことから，系列 c_1'' から作成された QR コードでは撮影環境が良好であれば悪性のサイトの URL が復号される．偽装 QR コードでは a 個の誤りは対応するモジュールの白黒を反転させることで「確定的な誤り」を付加し，残りの b 個の誤りは「確率的な誤り」を付加するというテクニックで作成される．b 個の誤りは確率的に生じることから，誤らなかった場合には正規サイトにアクセスし，誤りが生じた場合には悪性サイトにアクセスさせるといった確率的な悪性サイトへの誘導が実現される．図 3 に偽装 QR コードと誤りの与え方について示す．

偽装 QR コードを実現するために用いる確率的な誤りを発生させる方法は 2 種類提案されている．文献 [1], [2], [3] ではモジュールの中心部の輝度値を変更すること，文献 [5] ではモジュールの位置をずらして QR コードのデコーダプログラムで白黒判定を行う際の走査線の位置をずらすことによってカメラでの QR コードの読み取り後の白黒判定処理が不安定になって誤りが確率的に生じるようになる．特に文献 [1] の実験では，確率的な誤りを生じさせるモジュール（白色）の中心に打つ黒色の点の輝度値を 0 から 255 まで段階的に変更した結果，輝度値 0 では誤る確率が 65.8% であったのが輝度値 160 では 0.6% という低確率での誤りが生じたことが報告された．著者らも予備実験によって確認しているが輝度値と誤り率の関係は QR コードを表示する媒体，周囲の明るさや使用しているカメラの性能などに影響されるため，この結果が直ちに低確率で悪性サイトへ誘導を可能な QR コードの脅威を示すわけではない．しかしながら，誤り率をコントロールする方法が発展する可能性があることから注視が必要な技術と言える．

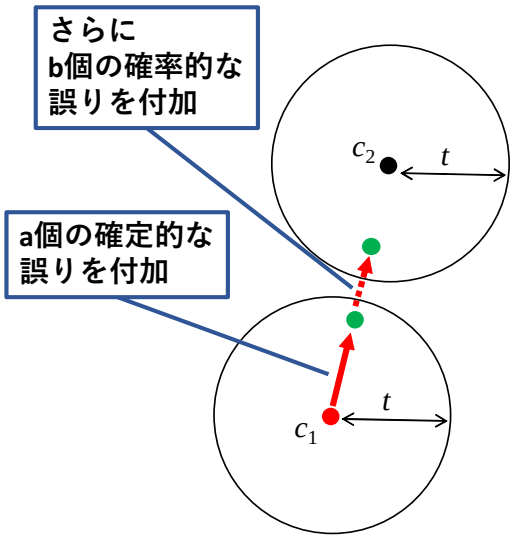

図 3　偽装 QR コードの原理

3.1.2　偽装 QR コードの作成手順

ここで正規サイトとして http://www.u-tokai.ac.jp/，悪性サイトとして http://www.u-yokai.ac.jp/ を例にして偽装 QR コードの作成手順を説明する．このとき，正規サイトの符号語は

$$\begin{aligned} c_1 = (&65, 150, 135, 71, 71, 3, 162, 242, 247, 119, 119, 114,\\ &231, 82, 215, 70, 246, 182, 22, 146, 230, 22, 50, 230,\\ &167, 2, 240, 236, 36, 111, 113, 168, 84, 82, 21, 223,\\ &143, 148, 4, 29, 86, 247, 145, 151) \end{aligned}$$

悪性サイトの符号語は

$$\begin{aligned} c_2 = (&65, 150, 135, 71, 71, 3, 162, 242, 247, 119, 119, 114,\\ &231, 82, 215, \mathbf{150}, 246, 182, 22, 146, 230, 22, 50, 230,\\ &167, 2, 240, 236, \mathbf{207}, \mathbf{252}, \mathbf{132}, \mathbf{239}, \mathbf{96}, \mathbf{205}, \mathbf{117}, \mathbf{61},\\ &\mathbf{38}, \mathbf{171}, \mathbf{42}, \mathbf{232}, \mathbf{175}, \mathbf{206}, \mathbf{111}, \mathbf{215}) \end{aligned}$$

となる．上記の c_2 の符号語の表記で太字にしているのは，c_1 と c_2 でシンボルが異なっているバイトである．URL で 1 文字変更したことで生じた 1 バイト分の差分とそれによって誤り訂正ブロック 16 バイト全てに差が生じたことから，c_1 と c_2 の距離は $d(c_1, c_2) = 17$ となる．なお，これは設計距離 $d = n - k + 1$ と一致しており，符号語間の距離は最小になり，復号誤りを生じさせやすい符号語のペアと言える．

偽装 QR コードを作成するために a 個の確定的な誤りと b 個の確率的な誤りを与える位置を決定する．文献 [1] に示された例に合わせて $a = 8$，$b = 1$ とする．RS 符号で復号可能な限界である 8 シンボルの誤りを与えて，追加で 1 シンボル誤る場合に符号語 c_2 への復号誤りを生じさせる．確率的な誤りは変化させるビット数が多いほどモジュールに変化を加える数が増えることから，c_1 と c_2 の異なる箇所の中でもハミング距離が小さいものを選択したい．その結果，確定的な誤りは E1〜E8，確率的な誤りを生じさせる箇所はハミング距離が 1 であった E16 にした．c_1 の E16 は $151 = (10010111)_2$，c_2 の E16 は $215 = (11010111)_2$ であり 2 ビット目だけ異なっており，そこに対応するモジュールを確率的に変化させる．確率的に変化するビットを ? の表記として，E16 にセットする値を $X = (1?010111)_2$ とした場合に，偽装 QR コードのための系列 c_1'' は以下のように与えられる．

図 4　偽装 QR コードの例

$$\begin{aligned} c_1'' = (&65, 150, 135, 71, 71, 3, 162, 242, 247, 119, 119, 114,\\ &231, 82, 215, 70, 246, 182, 22, 146, 230, 22, 50, 230,\\ &167, 2, 240, 236, \underline{207, 252, 132, 239, 96, 205, 117, 61},\\ &143, 148, 4, 29, 86, 247, 145, \underline{X}) \end{aligned}$$

なお，下線を引いたところが c_1 からの変更した箇所である．図 4 に c_1'' を用いて作成した QR コードを例として示す．確率的な誤りはモジュールの中心部の輝度値を変更する方法を採用し，白色 (0) を黒色 (1) に誤認識させるようにモジュールの中心の 1 ピクセルを輝度 0 に変更している．

上記の例では誤り訂正の範囲外に出るときに「誤り検出」でエラーが生じることが無いように必ず「復号誤り」が生じるように確率的な誤りを 1 ビットだけに適用していた．しかしながら，QR コードリーダでは誤り検出を起こした際にエラー出力をするのではなくリトライをする実装になっていることから，確率的な誤りは複数のシンボルやビットに付加しても通常の QR コードと同じように偽装 QR コードを作成することが可能となる．ただし，確率的な誤りを与えるビット数（改変するモジュール数）を増やしたときには誤り検出時のリトライが増えることで読み込みが成功するまでの時間が増加するというデメリットもある．

3.2　瀧田らの対策方法

瀧田らは SCIS2019 にて偽装 QR コードを用いた不正行為への対策について考察している [5]．

専用アプリケーションを利用せずに実施できる対策とし

て，読み取り後の動作を自動で行わないように設定を変更し，印刷物などに併記された情報と注意深く比較することが挙げられている．これは読みだした情報をむやみに信用しないように利用者に周知することで実現されるが，啓蒙活動には限界もあり，利用者に積極的に気付きを与える方法が必要であると思われる．

QR コードの読み込みアプリケーションを変更することで実施できる対策として，デコーダを用いて複数回の読み取りを試行し，出力方法を比較する方法が示されている．偽装 QR コードにより効果的に不正行為を行うためには，悪性サイトに誘導する確率を低確率にする必要がある．したがって，複数回の読み取りを試行して得られた情報が全て一致したときのみ出力することによって，利用者の利便性を損ねることなく安全性を確認できる．アプリケーション内部で 10 回程度の読み込みを繰り返したとしても処理時間に大きな影響は無いことから，この対策方法は現実的であると考えられる．

4. 偽装 QR コードの構成法の原理に基づく偽装 QR コードの検出手法

3.2 節で紹介した複数回の QR コードの読み込みにより悪性サイトに誘導されていないことを確認する対策方法はユーザが自衛する意味では十分に意味のある対策と思われる．しかしながら，この対策では各ユーザが意識せずに偽装 QR コードによる誘導を回避することとなり，偽装 QR コード自体は発見されずに掲示されたままとなる．この場合，対策法を導入していないユーザについては偽装 QR コードの脅威が継続することになるため，掲示板の管理者などが偽装 QR コードが掲示されていないかどうかを確認するツールの開発が必要となると思われる．そこで本節では掲示された偽装 QR コードを発見できるアプリケーションの開発を目指し，偽装 QR コードの構成法の原理に注目した検出手法を提案する．

4.1 誤り数に注目した手法

本節では QR コードを撮影後に RS 符号の復号する際に訂正された誤りの個数に注目した検出手法を提案する．QR コードで用いられている RS 符号の復号は

Step 1 受信系列からシンドロームを求める．
Step 2 シンドロームを用いて，誤り位置多項式を導出．
Step 3 誤り位置多項式を用いて，誤り位置を計算する．
Step 4 誤り評価多項式を用いて，誤っている数値の正しい数値を計算する．
Step 5 求められた誤り位置と正しい数値を用いて訂正を行い，復号結果を得る．

の順に実行され，Step 3 で得られる誤り位置から誤っている個数を得ることができる．したがって，QR コードのデコード時に誤りの個数を用いた偽装 QR コードの検出手法を実行することが可能となる．

偽装 QR コードでは確率的に誤るビット数が多くなると「誤り検出」の判定によるリトライが繰り返されて読み込みの時間が増加することから，確率的な誤りの個数 a を大きくし，確率的な誤りの個数 b を小さくすることになる．したがって，偽装 QR コードを読み込んだ場合，悪性サイトへ誘導されない場合でも誤り数は常に a 以上の大きなものとなる．これは通常の QR コードでは起こりにくい現象であり，通常の QR コードなのか偽装 QR コードなのかを判別する際に用いることが可能と考える．少なくとも QR コードが破損や汚れが無く，明るい場所などで安定して誤り数が多いようなことがあれば，不正 QR コードである可能性が高いと判定できる．具体的には，誤り数が事前に設定しておいた閾値を超える場合にユーザの画面に警告を出すなどして偽装 QR コードの可能性が高いことを示す方法が考えられる．

上記の検出手法は確率的な誤りの個数 a が大きくなることが前提であったが，誤り検出の判定によるリトライが増加して読み込み時間が増加することを許容する場合，確率的な誤りの個数 b を増やすことで平均的な誤り数を減らすことも可能である．しかしながら，そのような偽装 QR コードでは撮影するたびに誤り数が比較的高頻度で増減するといった特徴を持つことが予想される．悪性サイトへ誘導するために確率的な誤りが ℓ 個分必要であるとき，それぞれの誤る確率を $P_1, P_2, \ldots, P_\ell$ とする．悪性サイトへ誘導されるのは，全てが同時に誤る場合であるが，その確率は $\prod_{i=1}^{\ell} P_i$ となる．偽装 QR コードを効果的に使う場合には $\prod_{i=1}^{\ell} P_i$ を現実的に生じる程度に小さくするが，それぞれの確率的な誤りの生起確率 P_i は比較的大きくなる．例えば，4 つの確率的な誤りを付加するとき，それぞれの誤りは 20% で生じるように設定をして，悪性サイトへの誘導確率を 0.16% にするような使い方になると思われる．このような場合，悪性サイトに誘導されるまでには数百回から数千回の撮影が必要になるが，復号時の誤り数に注目すれば数回の撮影で誤り数が変化するような挙動を検出することが可能となる．以上のように複数回の撮影を実行した際の誤り数の変化の頻度は偽装 QR コードか否かの判定の補助的な情報となる可能性がある．

4.2 誤り位置に注目した手法

本節では QR コードを撮影後に RS 符号の復号する際に訂正された誤りの位置が誤り訂正ブロックに集中することに注目した検出手法を提案する．4.1 節で示した RS 符号の復号処理の Step 3 によると，誤り位置多項式を用いて誤り位置を取得することができる．したがって，QR コードのデコード時に誤り位置を用いた偽装 QR コードの検出手法を実行することが可能となる．

偽装 QR コードでは復号誤りを起こさせるための確率的な誤りの個数を少なくするために，正規サイトの符号語 c_1 と誘導する悪性サイトの符号語 c_2 の距離 $d(c_1, c_2)$ を小さくする必要がある．文献 [1] では正規サイトの URL を 1 バイト変更したものを悪性サイトの URL として偽装 QR コードを作成する方法が示されている．この場合，URL 部分で 1〜2 シンボルの差分が生じ，それ以外の差分は誤り訂正ブロックに生じる．これらの差分の個数である $d(c_1, c_2)$ は RS 符号の設計距離 $d = n - k + 1$ に近いものとなる．このように c_1 と c_2 の差分は誤り訂正ブロックに集中している．

c_1 から c_1'' を作成するために付加する確定的な誤りや確率的な誤りは，c_2 に近づくように誤らせるために符号語同士に差分がある箇所を移植することになる．そのため，c_1'' と c_1 の差分は誤り訂正ブロックに集中することになる．このとき，確率的な誤りが生じなかった場合には c_1 へ訂正されるため移植した c_2 のシンボル（3.1.2 節で言えば c_1'' の下線部分）が誤りとして訂正され，確率的な誤りが生じた場合には c_1 へ訂正されるため移植した c_2 のシンボル以外の c_1 と c_2 の差分（3.1.2 節で言えば c_1'' の下線部分以外の c_2 の太字部分）が誤りとして訂正される．これらは双方ともに大部分が誤り訂正ブロックである．したがって，誤り位置が誤り訂正ブロックである割合を利用することにより，偽装 QR コードを検出できる可能性がある．誤り訂正ブロックは図 1 のように近い位置にまとまって設置されているため，マジック等でバースト的に汚した場合に偽装 QR コードの検出に誤りが生じる可能性もある．そこで，誤り位置が誤り訂正ブロックである割合が事前に設定した閾値を超えた場合に利用者に対して汚している場所を確認するための表示をし，心当たりがない場合に偽装 QR コードの可能性が高いと判断する方法が考えられる．

4.3 誤り方に注目した手法

用紙に印刷された正規の QR コードに誤りが生じるとき，マジックで文字を書くなどして特定の範囲に色がついてしまう，用紙の一部が破れて白色になってしまう，撮影時にカメラの前に指がかかってしまうなど連続した誤りが生じるケースが想定される．一方，偽装 QR コードの確率的な誤りではモジュールの中心だけ色が変わるような特徴的な汚れの付き方で誤りを生じさせている．このように自然な QR コードの使い方での誤りの生じ方と偽装 QR コードでの誤りの生じ方で差異があり（図 5），その差異を偽装 QR コードの検出に役立てることが可能と考える．例えば，モジュールをまたいだ誤りが生じた際に，撮影した QR コードの画像を用いた画像処理により誤りを生じさせている汚れがモジュールをまたいで連続しているかを確認をするなどの方法が考えられる．

別の視点として，破れやマジックなどでの汚れではある範囲のモジュールが全て 1（黒）になるか 0（白）になるような誤り方をすることに基づく検出手法も考えられる．偽装 QR コードでは，悪性サイトの誤り訂正ブロックを移植することによる誤りの付加の際に，モジュール単位でみると白黒が交互に誤るなどの通常の汚れ方ではあり得ない誤りが検出されて訂正される．例えば，3.1.2 節の例では，c_1 の E1 は $36 = (00100100)_2$，c_1'' の E1 は $207 = (1101111)_2$ であり黒 (1) から白 (0) への誤りと白 (1) から黒 (0) への誤りが近いモジュールで複数見られる．他の誤りでも同様となっていることから，バースト的に全体が白や黒に変化するような特徴的な誤りと区別でき，それを用いて偽装 QR コードの検出に利用できる可能性がある．

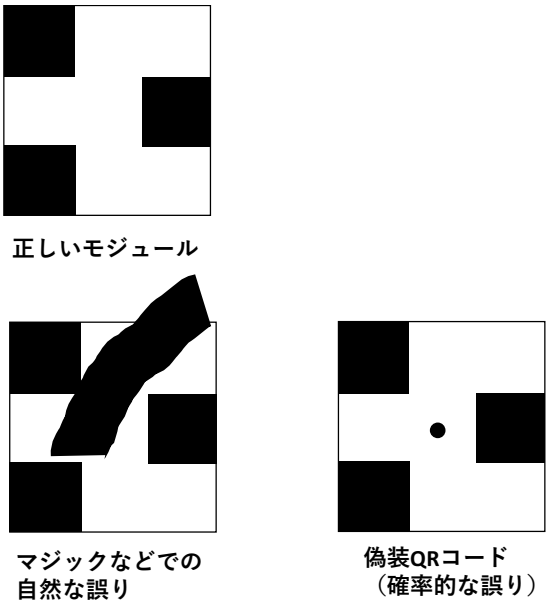

図 5　モジュールの誤り方の違い

5. 考察

本節では提案した偽装 QR コードの検出手法の制限や回避できる可能性について考察する．

5.1 デザイン QR コードをベースにした偽装 QR コード

デザイン QR コードは QR コードの誤り訂正能力を利用して復号可能な範囲でイラストなどを QR コードの埋め込んだものであり，偽装 QR コードのような改変をしない場合でも一定数の誤りが生じてしまう．したがって，大熊らが CSS2018 で議論していたデザイン QR コードをベースにした偽装 QR コードに対してはデザイン QR コードのために誤り数が多いのか偽装 QR コードであるために誤り数が多いのかの区別をすることが難しいため，誤り数に注目した偽装 QR コードの検出手法はデザイン QR コードには適さないと考えられる．なお，この場合にも誤り位置に注目した偽装 QR コードの検出手法は有効に働くと考えている．

5.2 誤り位置を変更する方法とその対策

誤り位置に注目した偽装 QR コードの検出手法を回避するために，偽装 QR コードの撮影時に誤り訂正される位置を誤り訂正ブロックから別のデータの位置に変更する方法を検討する．

まず初めに埋め草コードを利用した方法を考える．埋め草コードは URL 等のデータが短い際に特定のパターンで埋めて固定長のデータにするためのものである．フォーマットに沿ってデータを作成すれば 236 と 17 の繰り返しによって埋めることになるが，これを異なる値にしたとしても正常の QR コードと同じようにデコードすることができる．したがって，埋め草コードを変更しながら正規サイトの符号語 c_1 の誤り訂正ブロックと一致するシンボルが多い悪性サイトの符号語 c_2 を探索することによって，偽装 QR コード復号時の誤り訂正ブロックの位置の誤りの割合を減少させることができると考えられる．それを実現するために，埋め草コードを全数探索して誤り訂正ブロックの距離が短い符号語探索することもできるが，誤り訂正ブロックの一部のシンボルを c_1 のものに固定した上で，誤り訂正ブロックの残りのシンボルと埋め草コードを対象範囲にして消失訂正を実行することで符号語を生成する効率的な方法も考えられる．消失訂正を用いた符号語の生成方法は文献 [5] で議論されているものを参考にすれば良い．なお，この埋め草コードを用いた回避方法は，デコーダで復号後に埋め草コードのフォーマットチェックをした上で 236 と 17 の繰り返しになっていなければ出力しないような対策をすれば無効にできる．

埋め草コード以外で上記のような回避を実現できる方法として URL の GET メソッドのパラメータを利用する方法が考えらえる．例えば，`http://www.u-tokai.ac.jp/` に GET メソッドで意味のないパラメータを付加した `http://www.u-tokai.ac.jp/?ABCDEF` で Web アクセスをしたとすると，静的なページである場合や動的なページでも対応するパラメータが存在しない場合にはパラメータを付加していない場合と同様に Web サイトにアクセス可能である．そこで，正規のサイトを `http://www.u-tokai.ac.jp/?ABCDEF`，悪性サイトを `http://www.u-yokai.ac.jp/?ABCDEF` のように URL の長さを揃えておいて悪性サイトの URL の「?」以降を自由に変更することで，埋め草コードを対象にした上記の方法と同様に，全数探索や消失訂正によって誤り位置を変更した符号語を作成することが可能になると考えられる．「?」以外にも HTML のアンカーである「#」も同様に利用可能である．これらの URL の無視される部分を利用した方法は QR コードのフォーマットチェックでは無効化することができないため，「?」や「#」を含む URL が含まれていることを利用者に伝えた上で表示されている他の情報と比べるなどして悪性サイトかどうかの判断をする必要がある．

6. まとめ

本稿では，確率的に悪性サイトに誘導可能な偽装 QR コードを検出するための手法を提案した．具体的には，偽装 QR コードの構成法の原理に注目し，偽装 QR コードでは通常時の誤りの個数が比較的多くなることに注目した方法（誤り数に注目した検出手法），偽装 QR コードで生じる誤り訂正が誤り訂正ブロックに集中することに注目した方法（誤り位置に注目した検出手法），QR コードが誤る際の汚れの付き方が人為的な場合と偽装 QR コードで異なることに注目した方法（誤り方に注目した検出手法）の 3 種類の観点の手法を提案した．さらに，これらの方法が有効ではないケースについての考察も行っている．

提案検出手法は単独では検出を回避される可能性があるものもあり，偽装 QR コードを検出アプリケーションとして利用する際には各手法を併用する必要があると考えている．なお，本稿ではモジュールの中心の輝度値が異なる点を打つタイプの確率的な誤りの付加の仕方で議論をしているが，モジュールの位置をずらすことによる確率的な誤りの付加の仕方でも誤り数に注目した検出手法および誤り位置に注目した検出手法は同様に有効であると考えている．

現在，QR コードリーダのオープンソースライブラリ Zxing をベースにして提案検出手法の一部を実装することに成功しており，複数の検出手法の結果をスコア化する方法についての検討も進めている．実装を含めた詳細は稿をあらためて報告する予定である．

参考文献

[1] 大熊浩也，瀧田　愼，森井昌克：悪性サイトに誘導する QR コードの存在とそれを利用した偽造攻撃，電子情報通信学会技術研究報告, ICSS， Vol. 118, No. 109, pp. 33–38 (2018).
[2] 瀧田　愼，大熊浩也，森井昌克：誤り訂正符号に基づく偽装 QR コードの構成法とその脅威，情報科学技術フォーラム講演論文集，17 巻，第 4 分冊，pp. 1–6 (2018).
[3] Takita, M., Okuma, H. and Morii, M.: A Construction of Fake QR Codes Based on Error-Correcting Codes, *Sixth International Symposium on Computing and Networking, CANDAR 2018, Takayama, Japan, November 23-27, 2018*, IEEE Computer Society, pp. 188–193 (online), DOI: 10.1109/CANDAR.2018.00033 (2018).
[4] 大熊浩也，瀧田　愼，森井昌克：偽装 QR コードの構成とその効果，およびその対策について，コンピュータセキュリティシンポジウム 2018 論文集，pp. 1–6 (2018).
[5] 瀧田　愼，北川理太，大熊浩也，森井昌克：消失訂正を用いた偽装 QR コードの構成について，2019 年暗号と情報セキュリティシンポジウム予稿集，pp. 1–8 (2019).
[6] 今井秀樹：符号理論，電子情報通信学会 (1990).

付　　録

A.1 研究倫理について

本研究は現在も使用されている規格である QR コード

を用いた攻撃手法に関連しており，我々は本稿の執筆にあたって幾らかの視点から研究倫理に関して検討した．

本稿で扱っている偽装 QR コードの構成法は文献 [1], [2], [3], [4], [5] で既発表の内容であり，本研究は対策法について議論したものであり新規の脅威を示していない．また，偽装 QR コードによる悪性サイトへの誘導を個人で防御する手段は文献 [5] で既に提案されている．なお，原稿内では提案した掲示された偽装 QR コードを検出する方法を回避する手段について攻撃者の視点から考察しているが，提案検出法が用いられた場合の限定的な回避策であり，本稿を公開することにより QR コードの安全性が低下するものではない．