

# 動画像 QR コードとその応用

## Moving image QR Code and its application

古本 啓祐 †      渡辺 優平 ‡      森井 昌克 ‡

Keisuke Furumoto    Yuhei Watanabe    Masakatu Morii

### 1 はじめに

近年、2次元コードである QR コード [1] はデコーダを標準搭載した携帯電話の普及に伴い、Web ページへの誘導や連絡先の伝達といった様々な用途に利用されている。QR コードは縦横 2 方向に情報を保持するためバーコードのような 1 次元コードに比べて格納できるデータ量が飛躍的に増加している。このため、QR コードは静的な媒体である印刷物などにおいて、高密度な情報の共有、伝送が可能である。カラー多重化 QR コード [2] は同じ印刷物において、QR コードをカラー化することで高容量化を進めた方法である。しかし現在、紙媒体に QR コードを記載するだけでなく、デジタルサイネージなどの動的なディスプレイ上で QR コードを表示することも一般的となっている。動的なディスプレイ上では QR コードを動画像化、すなわち単位時間内に複数表示することが可能であり、高密度化を図ることができる。

本稿では動画像撮影能力、すなわち解像度を含めた画面キャプチャ能力に制限のあるスマートフォンなどのカメラを用いることを前提に、QR コードの動画像化方法の提案とその効率的な復号法を提案する。具体的には、格納したいデータを分割し、それぞれのデータから複数枚の QR コードを作成する。それらの作成した QR コードを連続して表示させることにより動画像 QR コードを形成する。動画像 QR コードを読み取る際には、スマートフォンなどに搭載されているカメラを用いて動画像として撮影する。そして、その動画像から各 QR コードのデコードを行う。それぞれのデコード結果を統合して、元のデータを復元する。本稿ではさらに、動画像 QR コードの読み取り精度を向上させるために、動画像を形成している各 QR コードに多重化を導入する方式を提案する。具体的には、動画像 QR コードを構成している複数枚の QR コードを多重化し、格納データを関連付ける。そして、読み取りの際に誤り訂正を行うことで、読み取りの精度の向上を図る。本稿で提案する動画像 QR コードにより、従来の QR コードでは格納できなかった容量の大きいデータも扱うことが可能になる。現在、最も多くの容量を埋め込めるバージョン 40 の QR コードにおけるデータ容量が 3KB 程度であることから、高々数秒内の認識時間において、その 10 倍以上の容量である数十 KB 以上を目標とする。

## 2 QR コード

### 2.1 QR コードの構成

QR コードには以下のような様々な規則がある。

- QR コードを構成する最小単位であるモジュールが

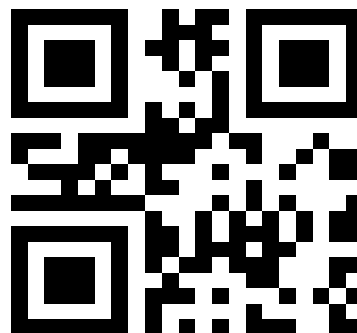


図 1 1 型 QR コード

明暗の 2 値を持ち、1 モジュールが 1bit の情報に相当する。

- QR コードは、機能パターン（位置検出パターン、タイミングパターンなど）と符号化領域（データコード語、誤り訂正コード語、形式情報など）から構成されている。
- データ容量に応じて、1 型から 40 型までの型番が存在する。1 型は  $21 \times 21$  モジュール、40 型は  $177 \times 177$  モジュールで構成される。図 1 に 1 型の QR コードを示す。
- BCH 符号 [3] や RS 符号 [4] といった誤り訂正符号が使われており、汚れや損傷にも強いという特徴がある。例えば、RS 符号の符号語は格納するデータから生成される。BCH 符号は後述する誤り訂正レベル指示子 (2 ビット) とマスクパターン参照子 (3 ビット) を含む形式情報の符号化の際に使用されている。
- 誤り訂正レベルは 4 段階 (L:7%, M:15%, Q:25%, H:30%) 規定されており、それぞれに 2 ビットの誤り訂正レベル指示子が割り当てられている。
- 全てのコードの配置が終了した後に、QR コードの読み取り精度を向上させるためにマスク処理と呼ばれる操作を行う。マスク処理は明暗が集中的に現れることを防ぐために行う操作である。8 種類のマスクパターンが用意されており、それぞれに 3 ビットのマスクパターン参照子が割り当てられている。

現在の QR コードに格納可能なデータ量は、最大でも約 3KB ほどであり、例えば画像や小説の一節などの大きなデータを格納するには容量が不足している。また、型番が大きくなるにしたがって、格納できるデータ量は増加するものの、読み取りの精度は低下してしまう。

### 2.2 RS 符号

RS 符号は局所的な誤りであるバースト誤りに強い誤り訂正符号であり、QR コードや CD、地上デジタル放送など多くの分野で利用されている。RS 符号は有限体 [5]GF(2)

† 神戸大学工学部, Faculty of Engineering, Kobe University

‡ 神戸大学大学院工学研究科, Graduate School of Engineering, Kobe University

の拡大体  $GF(2^m)$  上の元を処理単位 (シンボル) として構成されており, QR コードや CD には  $GF(2^8)$  上の RS 符号が利用されている. 多くの言語は 8 ビット 1 バイトが基準となっており,  $GF(2^8)$  上の RS 符号に変換しやすいという利点があるためである.

### 3 符号化変調

誤り訂正符号化とデジタル変調技術を統合的に設計し, 通信の効率性と信頼性を両立する符号化変調方式を採用することによって, 伝送速度を保ったまま通信の信頼性を向上させることができる.

#### 3.1 畳み込み符号

通信を行う際に送信データに対して符号化を行うことで通信の信頼性を向上させる技術が誤り訂正符号である. 誤り訂正符号は送信側がデジタルデータに対して冗長性を与え情報系列の系列長を長くし, それを受信側が復号処理をし, データの誤りの検出・訂正を行う.

畳み込み符号は誤り訂正符号の方式であり, RS 符号や BCH 符号などのブロック符号とは違い, 情報系列の各 bit を符号器に入力するごとに順次符号化が行われ, 現在の入力 bit と過去  $(K-1)$  bit によって出力 bit が決定する. この出力 bit の決定に影響を与える 2 元系列の長さ  $K$  を拘束長と呼ぶ. 畳み込み符号は一般に  $k$  bit の情報系列を平行に入力し,  $n$  bit の出力記号を得るため, 符号化率  $k/n$  の符号として生成される. 符号化率  $1/2$ , 拘束長 3 の畳み込み符号器を図 2 に示す. 畳み込み符号器は過去の入力を記憶するシフトレジスタと 2 を法とする加算器で構成される. 畳み込み符号におけるシフトレジスタの内部状態の遷移関係を時間軸で表したものにトレリス線図がある. トレリス線図において, 状態から状態へ遷移している様子を表す線分はブランチと呼ばれ, ブランチの連なりはパスと呼ばれる. 図 3 に符号化率  $1/2$ , 拘束長 3 の畳み込み符号におけるトレリス線図を示す. 図 3 において, 入力 bit が 0 のときを実線の矢印, 入力 bit が 1 のときを破線の矢印で表す. また, 拘束長 3 の場合, 過去の入力が 2bit 記憶されるため, 内部状態は 4 値存在し, それぞれ  $p_1p_2$  と示している. ここでは, 1bit 過去の入力を  $p_1 = \{0, 1\}$ , 2bit 過去の入力を  $p_2 = \{0, 1\}$  としている. 例えば, 時刻  $t=0$  における内部状態 00 のときの入力値が 1 であった場合, 内部状態は 10 に移行する. このときの出力値は, 図 2 の畳み込み符号器より 11 である. 符号化率の低い畳み込み符号から出力 bit を規則的に消去することによって, 間接的に構成される符号化率の高い符号をパングチャド畳み込み符号という. 畳み込み符号における代表的な復号法としてビタビ復号法が挙げられる. ビタビ復号法はトレリス線図上の前方から各パスの尤度を順次求め, 尤度の高いパスのみ継続して復号を行っていくことで, 受信系列に対して最も送信された可能性の高い符号系列を推定することができる.

#### 3.2 デジタル変調

実際に, 通信を行う際にはデジタルデータを電波や電気信号などのアナログ信号に変調して伝送する. その際にアナログ信号には通信路上の雑音などの物理的な問題が生じてしまう. その物理的な問題を考慮し, 信頼性を高めた上で高速通信を行うことを目的とした様々な変調方式が提案されている. 例えば, 送信するデジタル

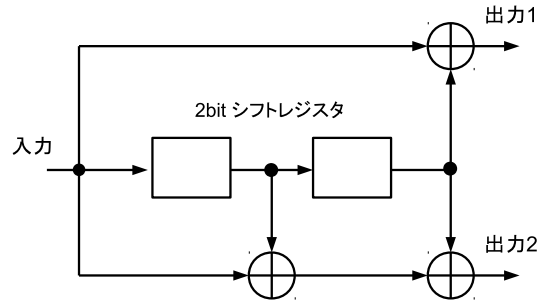


図 2 符号化率  $1/2$ , 拘束長 3 の畳み込み符号器

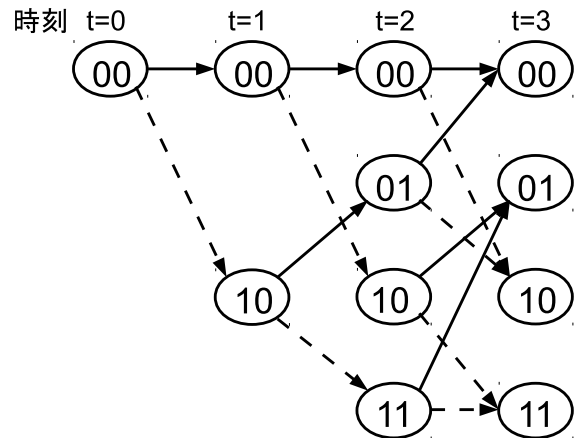


図 3 トレリス線図

データに従って信号の位相を変調する PSK や周波数を変調する FSK などがある.

### 4 動画 QR コードの生成とその読み取り

動画 QR コードの概要は以下ようになる. 容量の大きなデータを分割し, それぞれのデータから複数枚の QR コードを作成する. それらの QR コードを連続して表示する動画を作成する. その動画をスマートフォンなどに搭載されているカメラで動画撮影し, その動画データから各 QR コードを抜き出しデコードする. それらの各 QR コードから読み取ったデータを統合して元のデータを復元する. このような動画 QR コードを導入することで, 従来の QR コードと比較して扱えるデータ量の向上を図る.

#### 4.1 動画 QR コードの生成について

以下の手順で動画 QR コードを生成する. また, 図 4 に動画 QR コードの生成方式の概念図を示す.

1. 格納したい容量の大きなデータを分割し, それぞれのデータから QR コードを作成する.
2. 静止画の形式で作成された各 QR コードを, 1 秒間に表示する枚数を指定して連続的に表示させる.

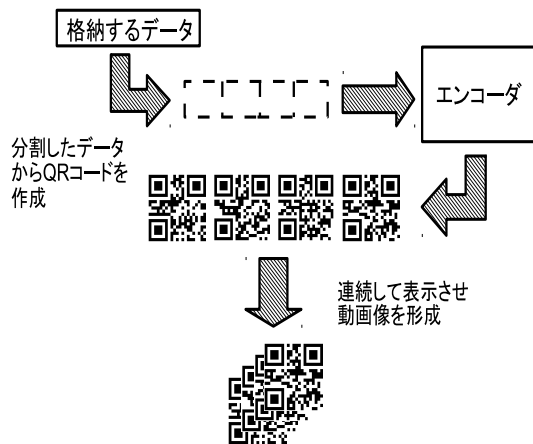


図4 動画 QR コードの生成

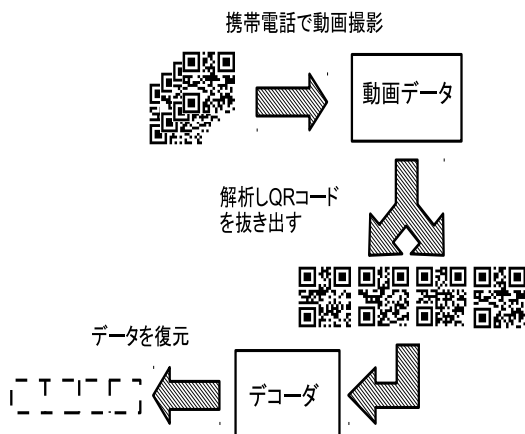


図5 動画 QR コードの読み取り

#### 4.2 動画 QR コードの読み取りについて

以下に動画 QR コードの読み取り手順を示す。また、図5に読み取り方式の概念図を示す。

1. 動画 QR コードをスマートフォンなどに搭載されているカメラで動画撮影し、撮影した動画データから各 QR コードを静止画として切り出す。
2. 静止画として切り出した QR コードをデコーダで読み取る。各 QR コードから読み取ったデータを統合し、元のデータを復元する。

スマートフォンなどに搭載されているカメラにより撮影された動画の仕様（圧縮形式等）も踏まえて、動画 QR コードの読み取りに関するデータを集める。

表1 実験条件

型番	誤り訂正レベル	1秒間に表示する枚数
1	L	10~30
7	L	10~30
9	L	10~30
10	L	10~30
13	L	10~30
13	M	10~30
13	Q	10~30
13	H	10~30

### 5 動画 QR コードの復元精度の評価

あるデータを分割し、型番と誤り訂正レベルを指定してそれぞれのデータから QR コードを作成する。作成する QR コードの型番と誤り訂正レベル、また動画 QR コードにおいて1秒間に表示する QR コードの枚数を表1に示す。これらの条件で動画 QR コードを作成する。それを PC の画面上で表示させ、スマートフォンなどに搭載されているカメラで動画撮影する。その動画データを PC 上で再生させ、再生画面に対して連続してキャプチャを実行する。1秒間にキャプチャする回数は1秒間に表示する QR コードの枚数に応じて指定する。例えば、1秒間に10枚の QR コードを連続して表示させている動画 QR コードに対しては、1秒間に10回のキャプチャを実行するように指定する。キャプチャして切り出した QR コードをデコードし、デコードの成功率を算出する。さらに動画 QR コードの構成に利用した全種類の QR コードについて読み取りの成否を確認する。動画 QR コードの構成に利用した全種類の QR コードにおいて読み取りに成功した場合に、各 QR コードの容量の合計をその動画 QR コードのデータ容量として記録する。なお、今回の評価において動画 QR コードを撮影した動画データを再生させ、キャプチャを実行する時間は、全体の処理時間が実用的な範囲に収まるよう3秒間と設定する。例えば、1秒間に10枚の QR コードを連続して表示させている動画 QR コードに対しては、3秒間では合計して30回のキャプチャが実行される。1秒間に表示する枚数が多いほど、キャプチャした際にノイズが発生する確率は高くなるものの、キャプチャの実行回数は多くなる。1秒間に表示する枚数とデコード成功率の関係などから、実用的な処理精度について検証を行う。この1秒間に表示する枚数を以後、動画において単位時間当たりに処理されるフレームの枚数であるフレームレート (fps) を用いて、再生速度 (fps) として扱う。

今回の実験では iPhone4S に搭載されているカメラを使用した。iPhone4S の動画撮影に関する規格は以下のようになっている。

- ・ファイル形式：mov
- ・圧縮規格：MPEG-4 AVC
- ・動画撮影時のフレームレート：30fps

本実験において、例えば 10fps の動画 QR コードを撮影した動画データに対して、キャプチャを1秒間に10回のペースで行った場合、動画 QR コードが表示され

表 2 型番を変化させた場合の復元データ量

型番	誤り訂正レベル	再生速度 (fps)	デコード成功率 (%)	復元データ量 (KB)
1	L	30	70	0.57
7	L	26	40	4.06
9	L	13	60	2.03
10	L	12	53	3.28
13	L	10	40	(3.42)

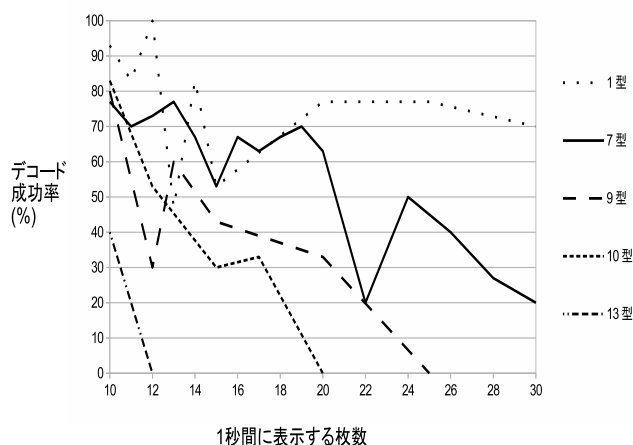


図 6 型番を変化させた場合のデコード成功率の比較

ている順序通りにキャプチャされていない実験結果も存在した。これは、コンピュータのキャプチャの実行に関する処理能力に原因があると考えられる。このため、QRコードが表示されている順番とは関係なく、ランダムに近い順番で QR コードがキャプチャされてしまう。しかし、本実験では 3 秒間撮影した動画画像から全種類の QR コードを復元することを目的としている。このため、1 秒間にキャプチャされた QR コードの順番を考慮する必要はないと考えられる。例えば QR コードの再生枚数 10 枚に対してキャプチャを 30 回実行し、その中から全ての QR コードの情報を復元できるのであれば、問題は生じないと考えている。

表 1 の条件に従い、型番を変化させた場合の実験結果は表 2、図 6 のような結果となった。復元データ量が () 付きで書かれている場合、動画画像 QR コードを構成している全ての QR コードを復元できなかった場合を表している。() 内の値は動画画像 QR コードを構成している QR コードの枚数に対して、復元できた QR コードの枚数の比が最も大きかったときの復元データ量である。

表 2、図 6 の実験結果により、以下のような事が確認できた。

- 1 型の QR コードからなる動画画像 QR コードの場合、再生速度を 30fps まで上げた場合でも、キャプチャした画像を 70% の確率でデコードすることに成功した。し

かし、1 枚あたりの格納データ量が少ないので、復元できたデータ量は 0.57KB であった。

- 7 型の QR コードからなる動画画像 QR コードの場合、再生速度を 26fps まで上げた場合でも、動画画像 QR コードを構成している全ての QR コードを復元できた。そのため、今回実験を行った中では 4.06KB と最も大きな値となった。
- QR コードの型番を 9 型、10 型と上げていった場合、QR コード 1 枚あたりに格納できるデータ量は増加するものの、動画画像 QR コードを構成している全ての QR コードを復元するためには再生速度の増加には限度があり、復元データ量は 7 型の QR コードからなる動画画像 QR コードのときよりも小さな値となった。
- QR コードの型番を 13 とした実験においては、再生速度が 10fps のときの動画画像 QR コードを構成している QR コード 10 枚の内の 2 枚を復元することができなかった。よって、復元データ量は高い値となっているが、もとの分割したデータを全て復元できていないので、ここでは他の条件での値との比較は行わない。
- 再生速度を上げるほど 3 秒間あたりのキャプチャの回数は増加するが、どの型番においてもデコードの成功率は低下する傾向にある。その中で 1 型や 7 型の QR コードからなる動画画像 QR コードであれば、再生速度を 30fps まで上げていった場合でもある程度はデコードに成功しデータを復元することができる。

表 1 の条件に従い、誤り訂正レベルを変化させた場合の実験結果は表 3、図 7 のような結果となった。

表 3、図 7 の実験結果により、以下のような事が確認できた。

- それぞれの誤り訂正レベルにおける、動画画像 QR コードを構成している QR コードの枚数に対して復元できた QR コードの枚数の比が最も大きかったのは、誤り訂正レベル L : 10 枚中 8 枚、誤り訂正レベル M : 10 枚中 7 枚、誤り訂正レベル Q : 12 枚中 7 枚、誤り訂正レベル H : 12 枚中 7 枚が復元できたときであり、どの誤り訂正レベルにおいても動画画像 QR コードを構成している全ての QR コードを復元することはできなかった。
- 誤り訂正レベルを上げた場合、デコードの成功率の改善は見られず、QR コード 1 枚あたりの格納データ量の減少しているため、全体の復元データ量も減少している。
- 再生速度を 12fps 以上に上げた場合、どの誤り訂正レベルにおいてもデコードの成功率は非常に低い結果となった。

表 3 誤り訂正レベルを変化させた場合の復元データ量

型番	誤り訂正レベル	再生速度 (fps)	デコード成功率 (%)	復元データ量 (KB)
13	L	10	40	(3.42)
13	M	10	43	(2.34)
13	Q	12	23	(1.71)
13	H	12	23	(0.26)

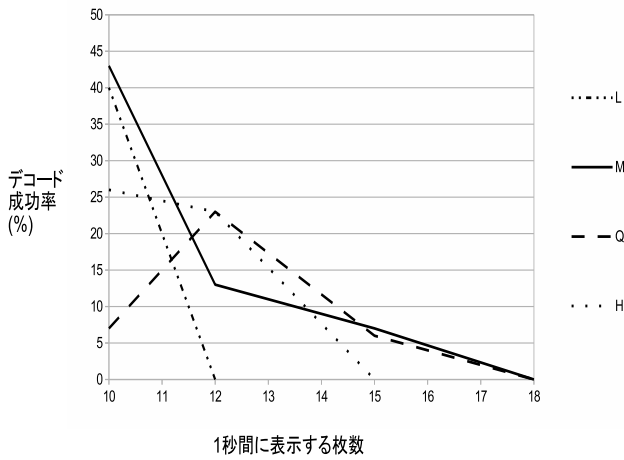


図 7 誤り訂正レベルを変化させた場合のデコード成功率の比較



図 8 デコード成功例



図 9 デコード失敗例

7 型の QR コードからなる動画 QR コードに対して実験を行った場合の、デコードの成功例と失敗例を図 8, 9 に示す。

## 6 符号化変調を利用した動画 QR コード

動画 QR コードのデコードの読み取り精度を向上させ、さらに動画 QR コードで扱える QR コードの枚数を増加させるために、符号化変調を利用して多重化を動画 QR コードに導入する。この多重化を導入した動画 QR コードの生成方法を以下に述べる。また、図 10 に符号化変調を利用した動画 QR コードの概念図を示す。

1. 格納したいデータを分割して、分割したデータから複数枚の QR コードを作成する。生成した  $i$  個目 ( $i = 1, 2, 3, \dots$ ) の QR コードのモジュールの明暗状態を 2 元系列で抜き出す。

$$a_i = (a_{1,i}, a_{2,i}, a_{3,i}, \dots) (i = 1, 2, 3, \dots) \quad (1)$$

式 (1) において、明暗情報を表す  $a_{t,i}$  ( $t = 1, 2, 3, \dots$ ) は QR コードの配置規則における  $t$  番目におけるモ

ジュールの明暗状態を表しており、ここでは明ならば  $a_{t,i} = 0$ 、暗ならば  $a_{t,i} = 1$  とする。

2.  $a_i$  における各 bit の明暗情報を抜き出し、情報系列  $b_1 b_2 b_3 \dots$  を生成する。情報ブロック  $b_t$  はそれぞれの QR コードにおける  $t$  番目の明暗情報からなる bit 列であり、式 (2) で表される。

$$b_t = (a_{t,1}, a_{t,2}, a_{t,3}, \dots) \quad (2)$$

3.  $b_1 b_2 b_3 \dots$  をパングチャド畳み込み符号化し、符号系列  $c_1 c_2 c_3 \dots$  を得る。
4. 符号系列  $c_1 c_2 c_3 \dots$  における情報ブロック  $c_t$  を複数枚の QR コードにおける  $t$  番目の明暗情報からなる bit 列であるとし、この bit 列を元に新たに複数枚の QR コードを生成する。そして、これらの QR コードからなる動画 QR コードを生成させる。

多重化を導入した動画 QR コードの復元方法は以下のようになる。

1. 動画 QR コードを構成している複数枚の QR コードを全てキャプチャを実行して抜き出す。それらの QR コードの中に、ノイズにより誤りが発生した QR コードがあったとする。
2. それらの QR コードから、生成した  $j$  個目 ( $j = 1, 2, 3, \dots$ ) の QR コードのモジュールの明暗状態を 2 元系列で抜き出す。

$$d_j = (d_{1,j}, d_{2,j}, d_{3,j}, \dots) (j = 1, 2, 3, \dots) \quad (3)$$

式 (3) において、明暗情報を表す  $d_{u,j}$  ( $u = 1, 2, 3, \dots$ ) は QR コードの配置規則における  $u$  番目におけるモジュールの明暗状態を表しており、ここでは明ならば  $d_{u,j} = 0$ 、暗ならば  $d_{u,j} = 1$  とする。

3.  $d_j$  における各 bit の明暗情報を抜き出し、情報系列  $e_1 e_2 e_3 \dots$  を生成する。情報ブロック  $e_u$  はそれぞれの

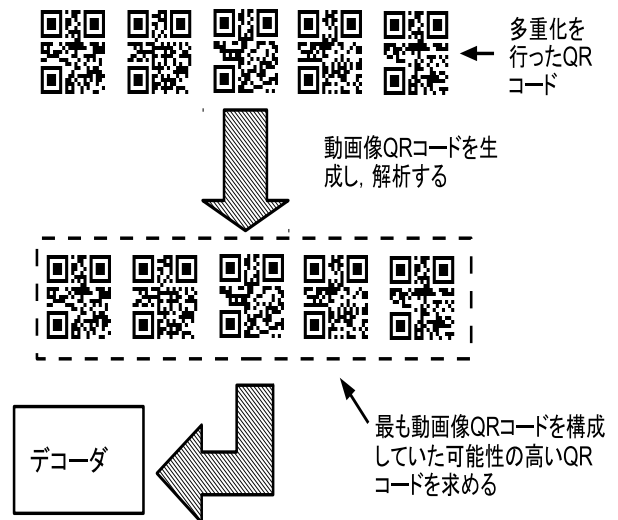


図 10 符号化変調を利用した動画 QR コード

QR コードにおける  $u$  番目の明暗情報からなる bit 列であり、式 (4) で表される。

$$e_u = (d_{u,1}, d_{u,2}, d_{u,3}, \dots) \quad (4)$$

4. 式 (4) に対しビタビ復号法を実行し、最も動画像 QR コードを構成していた可能性の高い各 QR コードの bit 列を求め、符号系列  $f_1, f_2, f_3, \dots$  を得る..
5. 符号系列  $f_1, f_2, f_3, \dots$  における情報ブロック  $f_u$  を複数枚の QR コードにおける  $u$  番目の明暗情報からなる bit 列であるとし、これらの bit 列からなる QR コードに対しデコードを実行する。

畳み込み符号化する際の拘束長を大きくするほど、復号の精度を上げることができるが、復号の際の計算量も指数関数的に増加してしまう。よって、実用的な処理時間となる範囲で拘束長を可能な限り大きく設定することが望ましい。

## 7 まとめ

本稿では QR コードのデータ容量の拡張を目的として、従来の QR コードに時間軸を加えた動画像 QR コードの構成方式、及び読み取り方式を提案した。さらに動画像 QR コードのデコード精度を向上させるために符号化変調を利用した多重化方式について考察した。動画像 QR コードにおいては、再生速度や QR コードの型番によって復元できるデータ量は変動するが、その中で復元できるデータ量が最大となる型番や再生速度の条件について評価を行った。

提案方式では、符号化変調を利用した多重化を動画像 QR コードに導入することにより、読み取りの際に誤り訂正を行い、デコードの精度を向上させることが可能となる。上記の評価では、誤り訂正レベル L の 7 型 QR コードを 26fps で表示した際に復元できるデータ量は最大となった。符号化変調を利用した動画像 QR コードを導入することで復元できるデータ量の拡張を目指す。多重化を導入した動画像 QR コードの生成方式と読み取り方式の実装と評価については今後の課題とする。

## 参考文献

- [1] 日本工業規格, “JIS,X0510, 二次元コードシンボル—QR コード—基本仕様, 2004”
- [2] 遠藤祐介, 廣友雅徳, 森井昌克, “高密度情報化を可能とする QR コード符号化方式について,” Forum on Information Technology 2010(FIT2010), 2010.
- [3] 今井秀樹, 符号理論, 電子情報通信学会, pp.151-155, 1990 年.
- [4] 今井秀樹, 符号理論, 電子情報通信学会, pp.155-161, 1990 年.
- [5] 今井秀樹, 符号理論, 電子情報通信学会, pp.42-75, 1990 年.