

**UNIVERSIDADE FEDERAL DE SANTA MARIA  
CENTRO DE TECNOLOGIA  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA**

**BARCODE COLORIDO COMPATÍVEL COM  
O DECODIFICADOR PADRÃO DO QR CODE**

**DISSERTAÇÃO DE MESTRADO**

**Matias Américo Bortoluzzi**

**Santa Maria, RS, Brasil**

**2015**

# **BARCODE COLORIDO COMPATÍVEL COM O DECODIFICADOR PADRÃO DO QR CODE**

**Matias Américo Bortoluzzi**

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Informática (PPGI), Área de Concentração em Computação, da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de  
**Mestre em Ciência da Computação**

**Orientador: Prof. Dr. Renato Machado**

**Co-orientador: Prof. Dr. Andrei Piccinini Legg**

**Santa Maria, RS, Brasil**

**2015**

Bortoluzzi, Matias Américo

Barcode Colorido Compatível com o Decodificador Padrão do QR Code / por Matias Américo Bortoluzzi. – 2015.

124 f.: il.; 30 cm.

Orientador: Renato Machado

Co-orientador: Andrei Piccinini Legg

Dissertação (Mestrado) - Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Informática, RS, 2015.

1. QR Code. 2. Barcode Colorido Bidimensional. 3. Cores de Referência. I. Machado, Renato. II. Legg, Andrei Piccinini. III. Título.

---

© 2015

Todos os direitos autorais reservados a Matias Américo Bortoluzzi. A reprodução de partes ou do todo deste trabalho só poderá ser feita mediante a citação da fonte.

E-mail: [matiasbortoluzzi@gmail.com](mailto:matiasbortoluzzi@gmail.com)

**Universidade Federal de Santa Maria  
Centro de Tecnologia  
Programa de Pós-Graduação em Informática**

A Comissão Examinadora, abaixo assinada,  
aprova a Dissertação de Mestrado

**BARCODE COLORIDO COMPATÍVEL COM O DECODIFICADOR  
PADRÃO DO QR CODE**

elaborada por  
**Matias Américo Bortoluzzi**

como requisito parcial para obtenção do grau de  
**Mestre em Ciência da Computação**

**COMISSÃO EXAMINADORA:**

**Renato Machado, Dr.**  
(Presidente/Orientador)

**Osmar Marchi dos Santos, Dr. (UFSM)**

**Bartolomeu Ferreira Uchôa Filho, Dr. (UFSC)**

Santa Maria, 18 de Março de 2015.

*Aos meus queridos pais Zenor e Beatriz e minha irmã Franciele, pelo exemplo de vida, suporte e confiança conferidos. Sem vocês nada disso seria possível.*

## AGRADECIMENTOS

Ao amor e ao poder infinito, ao qual ousamos chamar de Deus.

Ao meu pai Zenor, minha mãe Beatriz e minha irmã Franciele, pelo força, segurança e compreensão que garantiu a quem vos escreve persistir. A vocês devo tudo o que sou e o que ainda almejo ser...

Ao meu avô [*in memorian*] pelas orações... eu sei que elas me fizeram bem.

Aos demais familiares pelo incentivo e pela confiança que em mim depositaram.

A minha namorada Franciéli Kaspary Schons pelo companheirismo e compreensão conferidos.

Aos professores Renato Machado e Andrei Piccinini Legg pela condução deste trabalho e pela compreensão da atordoada vida de um aspirante a empresário.

Ao professor Osmar Marchi dos Santos e ao colega Lucas da Rocha Souza pela contribuição na confecção deste trabalho.

Ao CNPq pelo suporte financeiro.

À empresa SRA Engenharia pela compreensão das minhas limitações. À família SRA Engenharia, em especial, nas pessoas de Márcio Depexe, Thiago Cattani, Ytalo Marin e Arthur Werlang pelos momentos vividos aquém das fronteiras da empresa.

Aos demais amigos, colegas e professores que, mesmo em participação indireta, merecem minha gratidão.

Muito obrigado!

*“A persistência é o menor caminho do êxito.”*  
— CHARLES CHAPLIN

## RESUMO

Dissertação de Mestrado  
Programa de Pós-Graduação em Informática  
Universidade Federal de Santa Maria

### **BARCODE COLORIDO COMPATÍVEL COM O DECODIFICADOR PADRÃO DO QR CODE**

AUTOR: MATIAS AMÉRICO BORTOLUZZI

ORIENTADOR: RENATO MACHADO

CO-ORIENTADOR: ANDREI PICCININI LEGG

Local da Defesa e Data: Santa Maria, 18 de Março de 2015.

Motivados pelo elevado potencial de aplicações dos *barcodes* de alta densidade de dados que podem ser facilmente fotografados e decodificados pelos *smartphones*, e analisando as distorções impostas ao *barcode* imageado, este trabalho apresenta uma nova proposta de *barcode* colorido bidimensional compatível com o decodificador padrão do QR Code. Os dados são independentemente codificados em três *barcodes* monocromáticos que, em seguida, são combinados de modo que o novo *barcode* promova um aumento de três vezes na taxa de dados, quando comparado com o correspondente código de barras monocromático. Esta metodologia não apenas facilita a decodificação, mas também permite a reutilização de melhorias e otimizações desenvolvidas para o QR Code monocromático. Para minimizar o inevitável efeito danoso do canal, utilizam-se cores de referência distribuídas espacialmente ao redor do *barcode*, promovendo uma significativa melhora da robustez com baixo custo computacional. Além disso, analisou-se os limites aceitáveis de interferência no canal de imagem que são dependentes da impressora, das condições de captura e da câmera.

**Palavras-chave:** QR Code. Barcode Colorido Bidimensional. Cores de Referência.

# **ABSTRACT**

Master Thesis  
Post-Graduate Program in Informatics  
Federal University of Santa Maria

## **COLOR BARCODE COMPLIANT WITH THE STANDARD QR CODE DECODER**

**AUTHOR: MATIAS AMÉRICO BORTOLUZZI**

**ADVISOR: RENATO MACHADO**

**COADVISOR: ANDREI PICCININI LEGG**

**Place and Date: Santa Maria, March 18<sup>st</sup>, 2015.**

Motivated by a number of potential applications for high data-density barcodes that can be easily photographed and decoded by smartphones, and analyzing the distortion imposed on a barcode by phone photography, this work presents the design of a new colored two-dimensional barcode compatible with the QR Code decoder. The data is independently encoded into three monochrome barcodes which are then combined so that the new barcode lead to a three-fold increase in data rate compared to the corresponding monochrome barcode. This methodology not only facilitates the decoding but also allows the re-use of optimizations and improvements developed for monochrome QR Codes. In order to minimize the channel impairments, we use reference colors spatially distributed around the barcode, promoting a meaningful robustness improvement with low computational cost. Furthermore, we analysed the acceptable ranges of image channel distortion parameters that are printer, capture condition and camera dependent.

**Keywords:** QR Code. Colour Two-dimensional Barcode. Reference Colors.

## LISTA DE FIGURAS

Figura 2.1 – Histórico dos símbolos barcodes (adaptado de (SOON, 2008)). . . . .	22
Figura 2.2 – Capacidade de compactação dos códigos 2D em relação aos 1D (adaptado de (CHANG, 2014)). . . . .	23
Figura 2.3 – Estrutura da versão 7 do QR Code (adaptado de (ISO, 2006a)). . . . .	25
Figura 2.4 – Símbolos das versões 1 e 2 do QR Code (adaptado de (ISO, 2006a)). . . . .	25
Figura 2.5 – QR Code simples dividido em outros quatro menores (adaptado de (ISO, 2006a)). . . . .	26
Figura 2.6 – Estrutura da <i>codeword</i> na codificação do <i>RS</i> (adaptado de (KATO; TAN; CHAI, 2010)). . . . .	28
Figura 2.7 – Alguns detalhes sobre o padrão QR Code: (a) informação de formato, (b) informação de versão (adaptado de (ISO, 2006a)). . . . .	29
Figura 2.8 – Módulo de localização do QR Code. . . . .	30
Figura 2.9 – Codificador padrão do QR Code. . . . .	31
Figura 2.10 – Arranjo e sequência de preenchimento dos bytes de redundância e informação em um símbolo 2-M (adaptado de (ISO, 2006a)). . . . .	33
Figura 2.11 – Sequência de codificação dos bytes de redundância e de dados (adaptado de (ISO, 2006a)). . . . .	34
Figura 2.12 – Resultado da aplicação dos oito padrões de máscara do QR Code, de acordo com a Tabela 2.4. . . . .	35
Figura 2.13 – Decodificador padrão do QR Code. . . . .	36
Figura 2.14 – Processo de impressão e escaneamento como parte de um sistema de comunicação (adaptado de (Vizcarra Melgar et al., 2012)). . . . .	38
Figura 2.15 – Da esquerda para a direita: imagem original, distorção <i>barrel</i> e <i>pincushion</i> (extraída de (VASS; PERLAKI, 2003)). . . . .	41
Figura 2.16 – Ilustração da distorção <i>barrel</i> (extraída de (K. T. Gribbon, C. T. Johnston, 2003)). . . . .	41
Figura 2.17 – Câmera em posição na qual o formato quadrado do barcode é preservado (adaptado de (ALFTHAN, 2008)). . . . .	42
Figura 2.18 – Imagem capturada: (a) cenário da Figura 2.17, sem distorção perspectiva; (b) imagem com distorção perspectiva (adaptado de (OHBUCHI; HANAI-ZUMI, 2004)). . . . .	43
Figura 3.1 – Espectro de luz visível com as cores continuamente interligadas. . . . .	44
Figura 3.2 – Diagrama de cromaticidade CIE xy (valores em nanômetros). . . . .	46
Figura 3.3 – Processo de formação da imagem no olho humano: (a) Estruturas do olho sensíveis à luz refletida e exemplificada em (b) (adaptado de (SCRATCHA-PIXEL, 2012)). . . . .	47
Figura 3.4 – Comparação entre os picos de sensibilidade e absorção de energia pelas células óticas no espectro de luz visível (adaptado de (OpenStax College, 2013)). . . . .	48
Figura 3.5 – Comparação entre os diferentes valores de sensibilidade e absorção de energia pelas células óticas no espectro de luz visível. . . . .	49
Figura 3.6 – Cores primárias e secundárias de luz (a) e de pigmentos (b). . . . .	50
Figura 3.7 – Espaço de cores: (a) RGB e (b) CMY. . . . .	50
Figura 3.8 – Espaço de cores RGB (à esquerda) e HSI (à direita). . . . .	52

Figura 3.9 – Interferência <i>cross-channel</i> do espaço de cores CMY nos canais de escaneamento RGB: (a) CMY impresso; (b) canal R escaneado; (c) canal G escaneado; (d) canal B escaneado; (extraída de (BULAN; SHARMA, 2011)).	54
Figura 3.10 – Capacidade do canal preto e branco (adaptado de (LIU; DOERMANN; LI, 2008)).	56
Figura 3.11 – Capacidade do canal com $k$ cores selecionadas (adaptado de (LIU; DOERMANN; LI, 2008)).	57
Figura 3.12 – Diagramas RGB para 8 (a) e 6 (b) cores em testes realizados por (MEMETI, 2012) e (LIU; DOERMANN; LI, 2008), respectivamente.	57
Figura 3.13 – Exemplo de mapeamento dos valores RGB preto e branco relativos para os valores absolutos (extraído de (MEMETI, 2012)).	59
Figura 3.14 – Efeito da iluminação na alteração das cores (extraído de (BRIGGS, 2012)).	60
Figura 3.15 – Localização dos símbolos codificados, decodificados e da cor de referência (adaptado de (ONG; CHAI; RASSAU, 2011)).	62
Figura 3.16 – Localização das 20 amostras do símbolo original RGB = (0,255,255) (adaptado de (ONG; CHAI; TAN, 2009)).	63
Figura 3.17 – Agrupamento dos símbolos RGB = (0,255,255) e RGB = (255, 255, 0) de um barcode (adaptado de (ONG; CHAI; TAN, 2009)).	63
Figura 4.1 – Técnicas para incremento da capacidade de dados dos códigos de barras 2D.	66
Figura 4.2 – (a) Método de Codificação por modulação de pontos elípticos e o barcode resultante. (b) Imagem escaneada com as distorções inerentes ao canal <i>print-scan</i> e a decomposição nas componentes R, G e B (adaptado de (BULAN; SHARMA, 2011)).	69
Figura 4.3 – Codificação e decodificação de imagens animadas (adaptado de (MEMETI, 2012)).	70
Figura 4.4 – <i>Designs</i> do ColorCode (extraído de (KATO; TAN; CHAI, 2010)).	72
Figura 4.5 – Estrutura do símbolo do ColorCode (adaptado de (TACK-DON et al., 2006)).	72
Figura 4.6 – Incremento da capacidade de dados do HCCB (adaptado de (MICROSOFT, 2006)).	73
Figura 4.7 – Estrutura do símbolo do HCCB (adaptado de (KATO; TAN; CHAI, 2010)).	74
Figura 4.8 – Símbolo do PM Code e sua estrutura em camadas (adaptado de (KATO; TAN; CHAI, 2010)).	75
Figura 4.9 – Algoritmo PM Code de conversão de cor (adaptado de (KATO; TAN; CHAI, 2010)).	76
Figura 4.10 – Estrutura do barcode MMCC (adaptado de (KATO; TAN; CHAI, 2010)).	77
Figura 4.11 – HCC2D versão 5, nível de correção de erro H: (a) 4 cores; (b) 16 cores (extraído de (GRILLO et al., 2010)).	79
Figura 4.12 – Distribuição espacial das quatro paletas com oito cores (adaptado de (QUERINI et al., 2011)).	80
Figura 4.13 – Exemplo de aplicação de máscara no QR Code padrão e no HCC2D (adaptado de (GRILLO et al., 2010)).	82
Figura 4.14 – Estrutura do CQR Code (adaptado de (Vizcarra Melgar et al., 2012)).	83
Figura 4.15 – Histograma típico usado para calcular $th$ (adaptado de (Vizcarra Melgar et al., 2012)).	84
Figura 4.16 – Exemplo de QR Code colorido multiplexado a partir dos canais YCbCr independentes (adaptado de (KIKUCHI; FUJIYOSHI; KIYA, 2013)).	85
Figura 5.1 – Esquema de codificação do CQC.	88
Figura 5.2 – Esquema de codificação do CQC.	89

Figura 5.3 – Canal verde carrega a informação principal do barcode, enquanto que os canais vermelho e azul trazem as informações complementares. ....	90
Figura 5.4 – Ilustração da paleta de cores.....	92
Figura 5.5 – Esquema de detecção e decodificação do CQC. ....	94
Figura 5.6 – Limites paramétricos de distorções no QR Code para uma imagem de 120 × 120 pixels. ....	95
Figura 5.7 – Gráfico de dispersão das oito cores utilizadas na síntese do código CQC.....	98
Figura 5.8 – Dispersão das cores sob a vista de cada uma das faces do cubo RGB. ....	99
Figura 5.9 – QR Code submetido aos efeitos destrutivos que agem sobre o canal, determinando variações tonais. ....	100
Figura 5.10 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando a mínima distância Euclidiana. ....	101
Figura 5.11 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando a mínima distância Mahalanobis.....	102
Figura 5.12 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando o algoritmo <i>k-means</i> . ....	103
Figura 5.13 – Comparativo entre as técnicas de calibração de cor: sob distorção Gaussiana (gráfico à esquerda) e com covariância entre os canais R, G e B (gráfico à direita). ....	104
Figura 5.14 – BER para as resoluções de 43 × 43 e 86 × 86 pixels, utilizando códigos de barra sem correção tonal e os algoritmos de classificação <i>k-means</i> , mínima distância Euclidiana e Mahalanobis.....	105
Figura 5.15 – FRR para as resoluções de 43 × 43 e 86 × 86 pixels, utilizando as imagens sem correção tonal e corrigida, empregando a mínima distância Euclidiana. .	106
Figura 5.16 – FRR para as resoluções de 43 × 43 e 86 × 86 pixels, utilizando as imagens corrigidas, empregando o algoritmo <i>k-means</i> e a mínima distância Mahalanobis. ....	107
Figura A.1 – Da esquerda para a direita: bloco piloto, CQC com blocos pilotos nos cantos e CQC em escala de cinza com blocos piloto. ....	122
Figura B.1 – Gráfico de dispersão das oito cores utilizadas na síntese do código CQC.....	123
Figura B.2 – Dispersão das cores sob a vista de cada uma das faces do cubo RGB. ....	124
Figura B.3 – QR Code submetido aos efeitos do canal, determinando variações tonais. ...	124

## LISTA DE TABELAS

Tabela 2.1 – Níveis de correção de erro do QR Code.....	27
Tabela 2.2 – Indicadores de modo.....	32
Tabela 2.3 – Indicador de contagem de caractere.....	32
Tabela 2.4 – Regras para criação do padrão de máscara. ....	34
Tabela 2.5 – Quatro penalidades de ranqueamento das oito máscaras.....	35
Tabela 4.1 – Símbolos do MMCC.....	78
Tabela 4.2 – Capacidade de dados do QR Code padrão (ISO, 2006a).....	81
Tabela 4.3 – Capacidade de dados do HCC2D (GRILLO et al., 2010).....	81

## LISTA DE APÊNDICES

<b>APÊNDICE A – Matriz de Variação Tonal</b> .....	121
<b>APÊNDICE B – Exemplo de Gráfico de Dispersão de Cores</b> .....	123

# SUMÁRIO

<b>1 INTRODUÇÃO</b>	17
<b>1.1 Objetivos</b>	19
<b>1.2 Contribuições do Trabalho</b>	19
<b>1.3 Organização da Dissertação</b>	20
<b>2 BARCODES 2D MONOCROMÁTICOS</b>	21
<b>2.1 Evolução dos Barcodes Monocromáticos</b>	22
<b>2.2 QR Code</b>	23
2.2.1 Região de Codificação	26
2.2.2 Funções do Padrão	30
2.2.3 Codificador do QR Code	31
2.2.4 Decodificador do QR Code	36
<b>2.3 Distorções no Canal da Imagem</b>	37
2.3.1 Modelo de Degradação da Imagem	37
2.3.1.1 Borramento por Imagem Fora de Foco	39
2.3.1.2 Borramento por Movimento	40
2.3.1.3 Distorções Radiais	40
2.3.2 Distorção Perspectiva	42
<b>3 CANAL COLORIDO: CONCEITOS, PADRÕES E DEFINIÇÕES</b>	44
<b>3.1 Sensação e Representação das Cores</b>	44
3.1.1 Formação de Imagem no Olho Humano	46
3.1.2 Espaço de Cores	49
3.1.2.1 Espaço de Cores RGB	49
3.1.2.2 Espaço de Cores CMY e CMYK	50
3.1.2.3 Espaço de Cores HSI	51
<b>3.2 Separabilidade de Cores</b>	52
3.2.1 Interferência entre Canais de Cores	53
3.2.2 Distanciamento entre Cores	54
3.2.2.1 Capacidade do Canal	54
3.2.3 Determinação da Cor em Canais Multi-Cores	56
3.2.4 Calibração das Cores	60
3.2.4.1 Métricas para Análise das Variações Cromáticas	62
<b>4 TÉCNICAS PARA EXPANSÃO DO ARMAZENAMENTO DE DADOS</b>	65
<b>4.1 Codificação em Cores - Barcode Colorido</b>	66
4.1.1 Codificação de Dados por Matrizes de Pontos Modulados	69
4.1.2 Barcodes 4D	70
<b>4.2 Barcodes 2D Coloridos Existentes na Literatura</b>	71
4.2.1 ColorCode	71
4.2.1.1 Estrutura do Símbolo	72
4.2.2 HCCB	73
4.2.3 PM Code	75
4.2.4 MMCC	77
4.2.5 HCC2D	78
4.2.6 CQR Codes	81
4.2.6.1 Estrutura do Símbolo	82
4.2.6.2 Codificação e Decodificação do CQR Code	83

4.2.7 QR Code Colorido Compatível com o Decodificador QR Padrão .....	84
4.2.7.1 Codificação .....	84
4.2.7.2 Decodificação .....	86
<b>5 CÓDIGO DE BARRAS COLORIDO PROPOSTO .....</b>	<b>87</b>
<b>5.1 QR Code Colorido Proposto .....</b>	<b>87</b>
5.1.1 Codificação .....	88
5.1.2 Detecção .....	90
5.1.2.1 Cores de Referência .....	91
5.1.3 Decodificação .....	93
<b>5.2 Análise das Distorções Geométricas .....</b>	<b>95</b>
<b>5.3 Análise das Distorções Cromáticas .....</b>	<b>97</b>
5.3.1 Modelagem do Canal de Cor .....	97
5.3.2 Análise da BER .....	99
5.3.2.1 Desempenho das Técnicas de Classificação .....	100
5.3.2.2 Desempenho do Aumento da Resolução .....	103
5.3.3 Análise da FRR .....	105
<b>6 CONCLUSÃO .....</b>	<b>108</b>
<b>6.1 Possibilidades de Investigações Futuras .....</b>	<b>109</b>
<b>REFERÊNCIAS .....</b>	<b>111</b>
<b>APÊNDICES .....</b>	<b>120</b>

# 1 INTRODUÇÃO

Atualmente, o sistema mais utilizado para a decodificação de *barcodes* unidimensionais requer um leitor a *laser*<sup>1</sup> para escanear o código. Sua implementação em telefones móveis implica em um *hardware* extra, tornando-a muito cara e, portanto, impraticável. Além disso, não há garantias que, mesmo com *hardware* extra, um *barcode* bidimensional possa ser escaneado da mesma maneira do que um código unidimensional (ALFTHAN, 2008). Logo, em vez de se utilizar sistemas baseados em *laser* (*laser-based*) utilizaremos, neste trabalho, a câmera integrada ao celular (*camera-based system*).

A integração dos celulares com câmeras CCD transformou-os em dispositivos capazes de capturar imagens pessoais. Além disso, telefones celulares equipados com câmeras que capturam *barcodes* podem realizar tarefas de processamento de imagens e utilizar o resultado como uma fonte de “dados de contexto” (*context data*<sup>2</sup>) (ROHS, 2005). Neste sentido, as câmeras podem ser consideradas como uma forma de comunicação visual, em que a câmera atua como receptor (ASHOK et al., 2014). A expressão em inglês, *mobile 2D barcode* transcreve a tecnologia utilizada para condensar as vantagens dos *barcodes* bidimensionais nos telefones celulares, como forma de facilitar o acesso a qualquer informação, a qualquer tempo e lugar e através de qualquer mídia (MENG; YANG, 2008). Isto definitivamente sobrepuja a restrição da entrada de dados em telefones móveis que sempre foi limitada ao uso do teclado numérico (FALAS; KASHANI, 2007).

O cenário para a integração entre dispositivos celulares e câmeras é alusivo à evolução tecnológica deflagrada nas últimas décadas, naquilo que chamamos de “convergência tecnológica”. Neste conceito contemporâneo, a ideia central consiste na possibilidade de embarcar diferentes tecnologias em um mesmo dispositivo. É notório, portanto, que o celular incorporou inúmeras funcionalidades que antes compunham sistemas autônomos, tais como as funções de telefone, computador, rádio, MP3, câmera digital, etc. em um único *smartphone*<sup>3</sup> (telefone inteligente). Esses dispositivos inteligentes possibilitam interagir com o ambiente, o que con-

<sup>1</sup> Já existem aplicativos para celular que leem códigos unidimensionais, utilizando a câmera integrada. No entanto, são menos confiáveis.

<sup>2</sup> *Context data* é qualquer informação que pode ser usada para caracterizar a situação de uma entidade. Uma entidade pode ser uma pessoa, um lugar, ou objeto que é considerado relevante na interação entre o usuário e a aplicação (ABOWD et al., 1999).

<sup>3</sup> O primeiro *smartphone*, tal qual conhecemos hoje, com tela *touchscreen* foi o *iPhone* introduzido pela *Apple* em 2007, muito embora, já na década de 90 chegaram ao mercado os primeiros *smartphones*, oferecendo funcionalidades como: comunicação por e-mail, teclado *Qwerty*, navegador *web* baseado em texto e câmeras integradas.

ceitualmente caracteriza a computação “pervasiva” (*pervasive<sup>4</sup> computing*) diferenciando-a da computação tradicional, na qual o computador é visto como um mero “executor de tarefas” (SAHA; MUKHERJEE, 2003).

Um elemento chave para a computação ubíqua é a identificação (BRYANT et al., 2007), na qual cada “coisa”<sup>5</sup> precisa de uma identidade única. Uma maneira simples de fornecer uma identidade a uma “coisa” de modo fácil e confiável, e permitir que o usuário interaja com ela, é através de um *barcode* bidimensional. Portanto, cada vez mais as conexões não são mais apenas entre pessoas, ou pessoas e computadores, mas entre pessoas e “coisas” e, ainda, simplesmente entre “coisas”. Essas conexões criam novas possibilidades de iterações e acesso a uma grande quantidade de informação. Neste contexto, em 1991, Mark Weiser, diretor de tecnologia da *Xerox’s Palo Alto Research Center*, descreveu a visão de computação para o século XXI:

As tecnologias mais profundas são aquelas que desaparecem. Elas tecem-se no tecido da vida cotidiana, até que sejam indistinguíveis a partir dele (WEISER, 1999, tradução nossa).

Uma vez que a câmera tornou-se onipresente, as aplicações com *barcode* têm um potencial para alcançar uma audiência global ainda maior, bem como penetrar em novos mercados. Uma importante métrica para a relevância dos *barcodes* bidimensionais está no crescimento vertiginoso do número de usuários dos celulares, abordado em (EMARKETER, 2014). Em 2013 os usuários de telefones móveis somavam 4,33 bilhões de pessoas ao redor do mundo (61,1% da população). Para 2017, estima-se que esses valores chegarão a 5,13 bilhões e 69,4%, respectivamente. Na mesma época, em 2013, os *smartphones* contabilizaram 1,43 bilhões de usuários, ou seja, 33% do total de usuários de telefones móveis. Já em 2017, projeta-se que esses números atinjam 2,50 bilhões e 48,8%, nesta ordem.

Desde a inserção dos primeiros *barcodes*, inúmeros outros foram propostos com o propósito de, intercambiavelmente, aumentar a capacidade de armazenamento dados, ou de reduzir área ocupada. Diversas são as aplicações dos *barcodes*, compreendendo a identificação automatizada e captura de dados (*Automatic Identification and Data Capture - AIDC<sup>6</sup>*) (LIN; FUH, 2013) e a capacidade de rastreamento (BULAN; MONGA; SHARMA, 2009). Nessas aplicações, é desejável maximizar o número de bits armazenados por unidade de área, com o intuito de reduzir a área de impressão ou aumentar a carga de dados. O QR Code, por exemplo, é o *barcode* 2D de fácil legibilidade mais predominante na maioria dos *smartphones*. Embora

<sup>4</sup> Também chamada de computação ubíqua, ou *ubiquitous computing*.

<sup>5</sup> Aqui “coisa” significa objetos e dispositivos que não são computadores.

<sup>6</sup> Trata-se entrada de dados “sem tecla”, cujo objetivo é minimizar falhas humanas durante a digitação.

inicialmente designado para a indústria automotiva, a aplicação mais difundida do QR Code é o estabelecimento da conexão do telefone com a Internet, utilizando uma *uniform resource locator* (URL) (KATO; TAN; CHAI, 2010). Em uma ampla faixa de aplicações, o QR Code também se destina ao rastreamento comercial, passagens, autenticação, aplicações médicas (YEH; YOU; JONG, 2008) e *M-Commerce* (GAO; PRAKASH; JAGATESAN, 2007).

Apesar de suas inúmeras vantagens e aplicações, os *barcodes* capturados pelas câmeras estão sujeitos ao ruído, borramento, rotação, distorção perspectiva, iluminação desigual, ou ocultação de parte do código de barra (BELUSSI; HIRATA, 2011), dentre outros problemas comuns em *barcodes* monocromáticos. Devido aos danos resultantes do canal de imagem muitos decodificadores podem falhar na detecção do *barcode*. Logo, muitos trabalhos na literatura têm se preocupado com o processamento desse tipo de imagens para facilitar a decodificação. Além disso, com o passar dos anos as câmeras dos *smartphones* tornaram-se sensíveis às cores. Deste modo, uma simples maneira de aumentar a capacidade de dados, ou equivalentemente, reduzir tamanho de impressão dos *barcodes* é através do uso de módulos<sup>7</sup> coloridos. Contudo, além dos efeitos danosos que agem sobre o canal de imagem, comuns nos *barcodes* monocromáticos, os módulos coloridos tornam-se sujeitos aos efeitos das variações tonais dependentes dos dispositivos de reprodução e captura dos *barcodes* em cores.

## 1.1 Objetivos

Este trabalho tem como objetivo principal propor um novo *barcode* colorido baseado no decodificador padrão do QR Code, capaz de aumentar a densidade de dados por unidade de área. Paralelamente, desenvolver um *framework* para emulação do canal de imagem, contemplando diferentes distorções que atuam sobre o código de barras capturado.

## 1.2 Contribuições do Trabalho

Neste trabalho, podem-se destacar três importantes contribuições. A primeira delas refere-se ao codificador do padrão QR Code, baseado na ISO 18004:2006, totalmente desenvolvido em *software* Matlab. A segunda contribuição concentra-se na proposta de um novo *barcode* colorido, baseado no padrão QR Code e, portanto, apropriando-se de suas principais características. A terceira foca-se no estudo e implementação de modelos que simulam as distorções

---

<sup>7</sup> Módulo é a menor porção representativa que armazena dados em um *barcode*. Um módulo é composto por no mínimo um pixel.

mais presentes sobre o canal da imagem, a fim de se avaliar a robustez do código proposto.

### **1.3 Organização da Dissertação**

O restante da dissertação encontra-se organizado da seguinte maneira. Os Capítulos 2, 3 e 4 destinam-se a revisão da literatura técnica. O Capítulo 2 reserva-se à apresentação dos *barcodes* monocromáticos, com especial destaque ao QR Code. Além disso, reúne as principais distorções que afetam os códigos de barra. O Capítulo 3 analisa o espectro de luz visível e sua representação digital em espaço de cores. Ainda, conduz um importante estudo acerca das distorções cromáticas em *barcodes* coloridos.

No Capítulo 4, os principais códigos de barras coloridos são apresentados. Já no Capítulo 5, conduz-se a proposta principal deste trabalho, culminando com o *barcode* colorido compatível com o decodificador padrão do QR Code. Algumas análises são realizadas para validar o código proposto. Por fim, no Capítulo 6, tecem-se algumas conclusões sobre este trabalho, bem como possíveis futuras investigações da proposta.

## 2 BARCODES 2D MONOCROMÁTICOS

A palavra “*barcode*”, ou código de barras, originalmente utilizada para designar os tradicionais códigos unidimensionais (ISO, 2013), é geralmente preservada nas outras formas de codificação de dados em formatos de imagem que usam representação bidimensional (GRILLO et al., 2010), (BLASINSKI; BULAN; SHARMA, 2013). Neste contexto, quadrados, hexágonos, pontos, triângulos e demais padrões geométricos são referidos, sem prejuízo, como *barcodes*. Ademais, a palavra *barcode* de ascendência do inglês será adotada neste trabalho, sem o grifo itálico.

O barcode, também chamado de mídia de leitura óptica (*optically readable media - ORM*) pela ISO/IEC 19762-2, é uma representação de informação legível por máquina que é formada por combinações de altas e baixas regiões de refletância da superfície do objeto (KATO; TAN; CHAI, 2010) e que são convertidas para os níveis lógicos “0” ou “1” de acordo com alguma regra de mapeamento. São, portanto, capazes de armazenar informação digital (QUERINI; ITALIANO, 2013). A representação dos barcodes como uma sequência de bytes é mapeada para sinais analógicos (símbolos) que são transmitidos sobre um canal de impressão e recebidos através de um sensor fotosensível (câmera ou scanner). Este canal introduz ruído, distorção e interferência, corrompendo o sinal recebido, ou seja, o barcode digitalizado.

Os barcodes 2D utilizados para aplicações celulares podem ser funcionalmente classificados em duas categorias (KATO; TAN; CHAI, 2010), (LIU; DOERMANN; LI, 2008): banco de dados (*database barcode*), também chamado de orientado ao conteúdo (*content oriented*); e índice (*index barcode*), também referido como orientado a interação (*interaction orient*). Os primeiros se referem a qualquer barcode 2D que é escalonável em termos de capacidade de dados e que entregam diretamente metadados aos *smartphones*, ou qualquer outro dispositivo que possa ser utilizado para leitura e decodificação dos barcodes. Já os segundos codificam poucos bytes de informação que pode ser utilizada como um índice em direção a um conteúdo online.

Os *database barcodes* incluem o QR Code, Data Matrix, VeriCode, mCode, Trillcode e o HCCB, enquanto que o ShotCode, Visual Code, BeeTagg e ColorCode são categorizados como *index barcodes*. Mesmo os barcodes orientados ao conteúdo possuem restrições na sua capacidade de dados. Este cenário restrito traz enormes desafios aos pesquisadores que têm se debruçado em propostas de barcodes coloridos e animados, conforme ambicionamos apresentar no Capítulo 4.

Este capítulo, dividido em três seções, tem como interesse principal descrever o padrão QR Code, baseado na ISO/IEC 18004:2006 (ISO, 2006a), e caracterizar o canal de imagem. Para isso, na Seção 2.1, realiza-se uma revisão histórica do desenvolvimento dos barcodes monocromáticos. Em seguida, na Seção 2.2, uma apresentação detalhada do padrão QR Code é concebida. Finalmente, a Seção 2.3 descreve o canal de imagem, apresentando detalhes sobre os efeitos de ruído e distorção.

## 2.1 Evolução dos Barcodes Monocromáticos

Em 1970, a IBM desenvolveu os símbolos *Universal Product Code* (UPC), consistindo em 13 dígitos numéricos para habilitar a entrada automatizada em computadores. Em 1974, o Code 39 com capacidade de codificação de até 30 dígitos numéricos ou alfanuméricos foi desenvolvido. Por volta dos anos 80, símbolos multiestágios (empilhados) com aproximadamente 100 dígitos de capacidade foram desenvolvidos, a saber o Code 16K e o Code 49. Com a necessidade do aumento da capacidade de informação e a popularização dos computadores, em 1994 o QR Code foi desenvolvido, substituindo os códigos multiestágios por códigos matriciais. Desde então, os códigos matriciais são predominantemente utilizados. A Figura 2.1 ilustra a evolução temporal das tecnologias supracitadas.

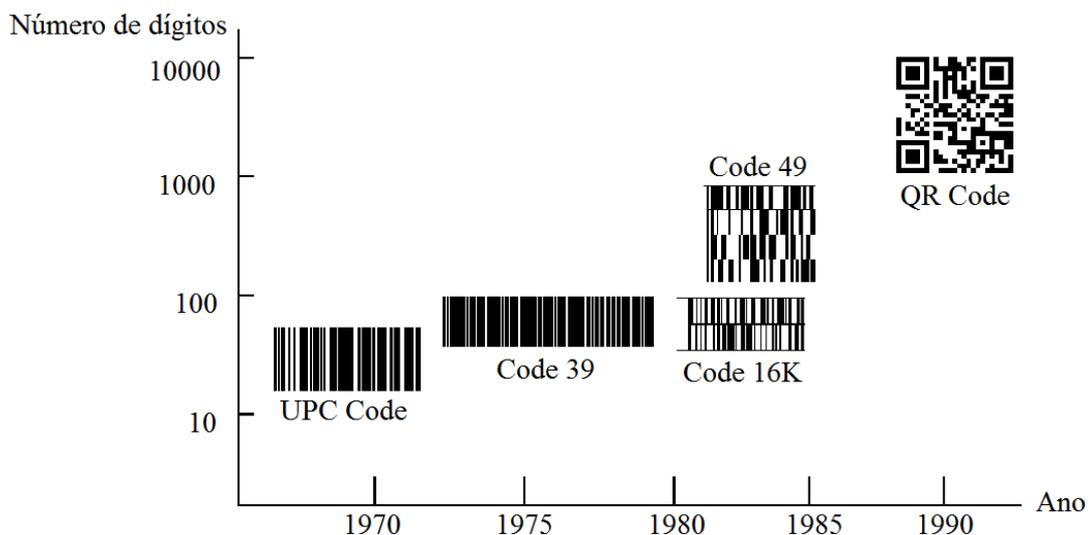


Figura 2.1 – Histórico dos símbolos barcodes (adaptado de (SOON, 2008)).

De acordo com a estrutura e forma (vide Figura 2.1) os barcodes podem ser divididos em Linear (unidimensional - 1D) e Matricial (bidimensional - 2D) (Yunhua Gu; Weixiang Zhang,

2011). Sendo assim, quando comparados aos barcodes 1D, os barcodes 2D apresentam as seguintes vantagens (MENG; YANG, 2008):

- Alta densidade de informação: resume-se a capacidade de armazenar mais dados em um mesmo espaço, ou de reduzir o espaço ocupado para uma mesma quantidade de informação (ilustrado na Figura 2.2). Assim, milhares de caracteres alfanuméricos podem ser alocados em um único símbolo;



Figura 2.2 – Capacidade de compactação dos códigos 2D em relação aos 1D (adaptado de (CHANG, 2014)).

- Ampla faixa de codificação: os barcodes 2D permitem codificar não apenas números, mas textos, figuras, arquivos de voz, e dados digitais em diferentes linguagens;
- Alta confiança e durabilidade: os símbolos 2D podem ser recuperados mesmo com acentuada degradação da imagem;
- Forte segurança: os barcodes 2D consideram o uso de técnicas de compressão e encriptação de dados, tendo como possível aplicação a inclusão de dados em imagens. Por exemplo, em (GARATEGUY et al., 2014), QR Codes são embarcados em imagens coloridas;
- Alto custo-benefício: os barcodes 2D podem ser facilmente impressos, utilizando as mídias de impressão disponíveis, na maioria dos substratos de impressão, como por exemplo, papel, PVC, metal, etc.

## 2.2 QR Code

O potencial dos *mobile 2D barcodes* tem atraído a atenção de pesquisadores e engenheiros para o desenvolvimento de barcodes com características cada vez mais aprimoradas. Dentre os barcodes 2D existentes o QR Code e o DataMatrix foram pioneiros na implementação de

aplicações *mobile*, demonstrando suas incríveis capacidades. Apesar da grande variedade de barcodes existentes, as próximas seções deste capítulo focam no padrão QR code com o devido rigor, sapientes que suas principais estruturas também integram outros códigos 2D.

O *Quick Response Code* (QR Code) é um barcode bidimensional na forma de código matricial desenvolvido pela *Japanese Denso Corporation* em setembro de 1994 (Yunhua Gu; Weixiang Zhang, 2011) e que, depois, foi reconhecido como um padrão pela *AIM Standard*, *JIS Standard* e *ISO Standard* (ISO, 2006a). Existem 40 versões do QR Code, quatro níveis de correção de erro e, em seu tamanho máximo de símbolos (versão 40), podem ser codificados 7089 dados numéricos, ou 4296 dados alfanuméricos. O algoritmo de correção de erro *Reed-Solomon* é utilizado no QR Code, sendo que o nível de correção de erro mais elevado permite corrigir até 30% das *codewords*<sup>8</sup> presentes no símbolo. Adicionalmente, o QR Code é composto por três módulos de localização com quadrados concêntricos.

O QR Code possui características mais avançadas (LIU; LIU, 2006) quando comparadas ao código PDF417 (ISO, 2006b):

- Alta capacidade para codificar dados: o QR Code pode embarcar até 7089 caracteres, enquanto que o PDF417 limita-se a apenas 2710 caracteres;
- Rápida leitura do barcode: mais símbolos por segundo podem ser reconhecidos com o QR Code quando comparado ao PDF417, considerando-se a mesma quantidade de dados armazenados;
- Legível a partir de qualquer direção: o QR Code pode ser lido em qualquer ângulo rotacional, enquanto que o PDF417 apresenta dificuldades para reconhecimento do barcode em ângulos diversos.

O símbolo do QR Code é constituído por uma área de codificação (*encoding region*) e pelas funções do padrão (*function patterns*) (Yue Liu; Ju Yang; Mingjun Liu, 2008). Cada uma dessas regiões é composta por outras estruturas, conforme pode ser visto na Figura 2.3. De acordo com a refletância de luz dos módulos do QR Code, essas estruturas codificam o bit “0” em módulos brancos e o bit “1” nos módulos pretos.

O padrão QR Code é estruturado no formato “QR Code”, com todas as características e máxima capacidade, e no formato Micro QR Code que possui capacidade reduzida. Neste trabalho, nosso foco está no QR Code, uma vez que estamos interessados em explorar altas

<sup>8</sup> Uma *codeword* também é chamada de palavra-código.

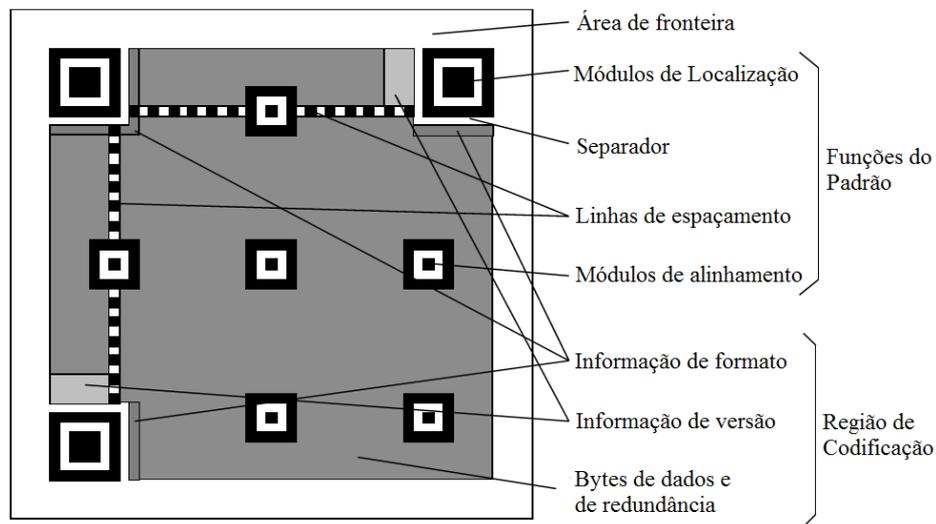


Figura 2.3 – Estrutura da versão 7 do QR Code (adaptado de (ISO, 2006a)).

densidades de dados. O tamanho do QR Code pode variar de  $21 \times 21$  (versão 1) a  $177 \times 177$  (versão 40) módulos, com incrementos de quatro módulos por versão. A relação entre versão ( $V$ ) e número de módulos ( $n_{mod}$ ) é dada por:

$$n_{mod} = 21 + 4 \times (V - 1). \quad (2.1)$$

A título ilustrativo, a Figura 2.4 traz as versões 1 e 2, com seus respectivos tamanhos em módulos, apresentando-as em consonância com a Equação 2.1.

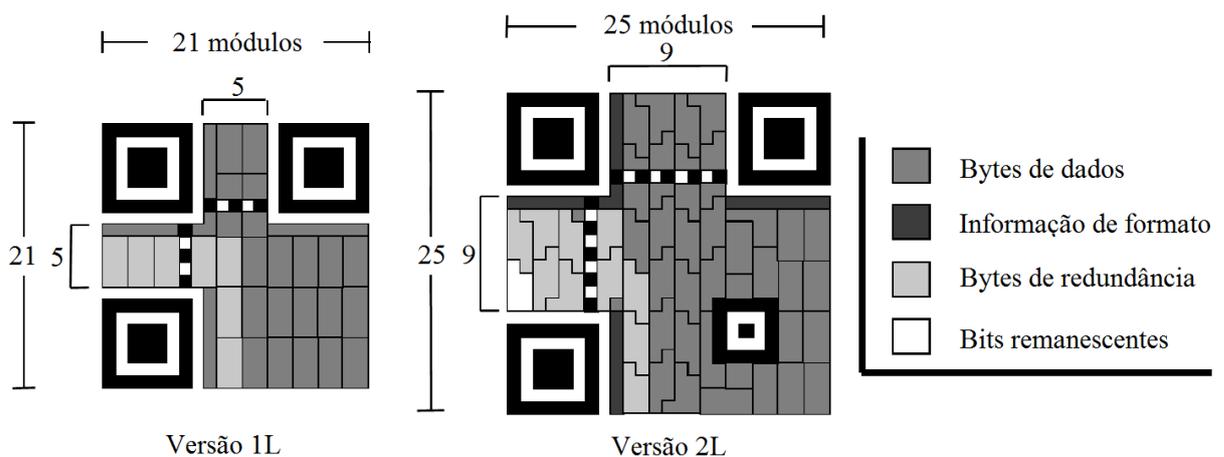


Figura 2.4 – Símbolos das versões 1 e 2 do QR Code (adaptado de (ISO, 2006a)).

O padrão QR Code possui o recurso de “acréscimo estruturado” (*structured append feature*) que permite dividir um QR Code em até 16 outros, quando o espaço de impressão

disponível não for suficientemente grande. Este recurso é exemplificado na Figura 2.5, na qual a informação de um QR Code é dividida em quatro outros barcodes.

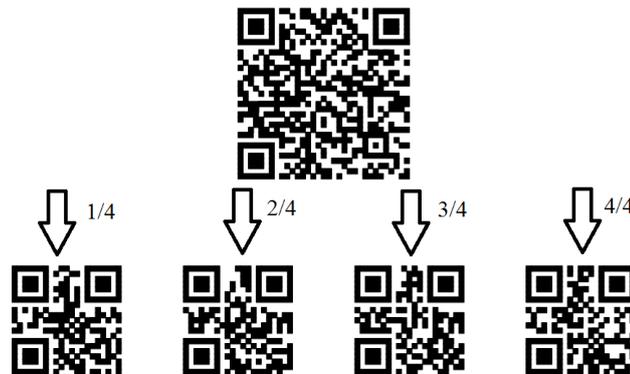


Figura 2.5 – QR Code simples dividido em outros quatro menores (adaptado de (ISO, 2006a)).

A seguir as estruturas constituintes do QR Code serão discutidas, baseadas em ampla documentação presente na literatura. As próximas seções consideram a versão 7 do QR Code (Figura 2.3), sem realizar uma referência formal a ela.

### 2.2.1 Região de Codificação

A região de codificação é composta por *codewords*<sup>9</sup> responsáveis pelo armazenamento dos dados e pelos bits de redundância do *Reed-Solomon*. Além disso, duas regiões menores da região de codificação embarcam as informações de versão (*version information*) e formato (*format information*) do QR Code.

- **Dados**

O padrão QR Code permite a codificação dos seguintes tipos de dados:

1. Números (dígitos de 0 ~ 9). Capacidade máxima: 7089 caracteres;
2. Alfanuméricos (dígitos 0 ~ 9, letras maiúsculas A ~ Z e outros nove caracteres especiais: espaço, \$ % \* + - . / :). Capacidade máxima: 4296 caracteres;
3. Byte (padrão ISO/IEC 8859-1). Capacidade máxima: 2953 caracteres;
4. Caracteres Kanji que podem ser compactados em 13 bits. Capacidade máxima: 1817 caracteres.

<sup>9</sup> Conjunto de 8 bits alocados na região de codificação.

Para maiores detalhes, a Tabela 7 da ISO/IEC 18004:2006 apresenta a capacidade de codificação para cada um dos tipos de dados, variável com a versão e com a capacidade de correção de erro.

- **Correção de Erro**

Nos códigos 1D a redundância vertical nas barras, associada com a impressão numérica (legível) em cima ou abaixo do barcode, permite que a detecção do erro por meio da implementação do *checksum* seja suficiente, pois o usuário pode conferir a numeração do código de barras que é geralmente impressa. No entanto, para os barcodes 2D essa solução não se aplica, uma vez que, além de detectar, necessita-se corrigir os erros. Neste sentido, dentre várias técnicas de detecção e/ou correção de erros a mais comumente utilizada em barcodes 2D é o algoritmo *Reed-Solomon (RS)*.

Os códigos corretores de erro são projetados para corrigir erros de transmissão, adicionando informação extra (redundância), que faz uma checagem cruzada com os dados originais (BLAHUT, 2003). Desta forma, a correção de erro do QR Code, baseada no algoritmo *Reed-Solomon*, permite quatro níveis de reparação de erros, todos eles apresentados na Tabela 2.1:

Tabela 2.1 – Níveis de correção de erro do QR Code.

Nível	Correção Aproximada
L	7%
H	15%
M	25%
H	30%

Os códigos *Reed-Solomon* são especificados na literatura como  $RS(n, k)$  com  $r$  bits por símbolo. Isto significa que  $k$  símbolos de dados, cada qual com  $r$  bits, são conjuntamente codificados com  $n - k$  símbolos de paridade. O código *Reed-Solomon* é capaz de corrigir até  $t$  símbolos de erro em uma *codeword*, ou seja, metade dos símbolos de paridade. Matematicamente:

$$t = \frac{n - k}{2}. \quad (2.2)$$

A estrutura de codificação do *RS* concatenada é ilustrada na Figura 2.6. Observe que o conjunto de  $n$  símbolos representa uma *codeword*.

Para exemplificar, considere o  $RS(255,233)$ , com  $r = 8$  (um byte por símbolo, total de 255 bytes), no qual a *codeword* possui 255 símbolos, dos quais 233 são informação. Resulta,

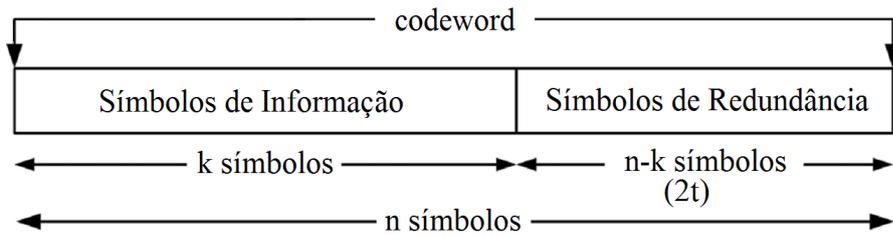


Figura 2.6 – Estrutura da *codeword* na codificação do RS (adaptado de (KATO; TAN; CHAI, 2010)).

portanto, 32 símbolos de redundância. Em consonância à Equação 2.2, conclui-se que  $t = 16$ , ou seja, 16 símbolos dos 255 podem ser recuperados, conduzindo ao percentual de correção de 6,27%. Na decodificação o algoritmo RS irá decodificar os  $n$  símbolos e apontará quais deles estão errôneos. Em um cenário no qual mais do que  $t$  símbolos estão incorretos não será possível recuperar os símbolos de informação  $k$ .

Na geração dos  $n - k$  símbolos de paridade, o algoritmo RS utiliza operações aritméticas em um Corpo de *Galois* GF finito, baseado em um polinômio gerador<sup>10</sup> ( $PG$ ), o qual pode ser expressa por:

$$PG = (x - \alpha^0)(x - \alpha^1) \dots (x - \alpha^{n-1}), \quad (2.3)$$

em que  $n$  é o número de *codewords* de correção de erro (Tabela 9 da ISO/IEC 18004:2006) e  $\alpha$  é um elemento primitivo do Corpo de *Galois* ( $\alpha = 2$ ).

Por outro lado, as *codewords* da mensagem<sup>11</sup> convertidas para decimal constituem os coeficientes do polinômio mensagem ( $PM$ ), que pode ser sumarizada por

$$PM = CW_1x^{m-1} + CW_2x^{m-2} + \dots + CW_mx^0, \quad (2.4)$$

em que  $CW$  corresponde a cada uma das *codewords* da *bitstream* codificada.

As *codewords* do RS serão geradas pela divisão polinomial entre os polinômios mensagem e gerador.

### • Informação de Formato

A informação de formato contém uma sequência de 15 bits, dos quais 5 são bits de dados, e outros 10 são bits de correção de erro calculados usando o código BCH<sup>12</sup> (15,5). Os

<sup>10</sup> Todos os polinômios geradores do QR Code estão apresentados na Tabela A.1 da ISO/IEC 18004:2006.

<sup>11</sup> *Bitstream* resultante da conversão dos caracteres de dados em bits.

<sup>12</sup> *Bose-Chaudhuri-Hocquenghem code*.

primeiros dois bits correspondem a sequência L (“01”), M (“00”), Q (“11”) e H (“10”). Do terceiro ao quinto bits está alocada a informação de máscara. São ao todo oito máscaras<sup>13</sup>, cujo valor em bits de cada máscara corresponde ao valor decimal (0 ~ 7) convertido para binário.

Os 15 bits da informação de formato aparecem duas vezes no QR Code, como pode ser visto na Figura 2.7 (a). Na mesma figura é observável o módulo preto (*dark module*) posicionado ao lado do módulo de localização inferior esquerdo. Mais especificamente na posição  $([4V + 9], 8)$ , em que  $V$  é a versão do QR Code.

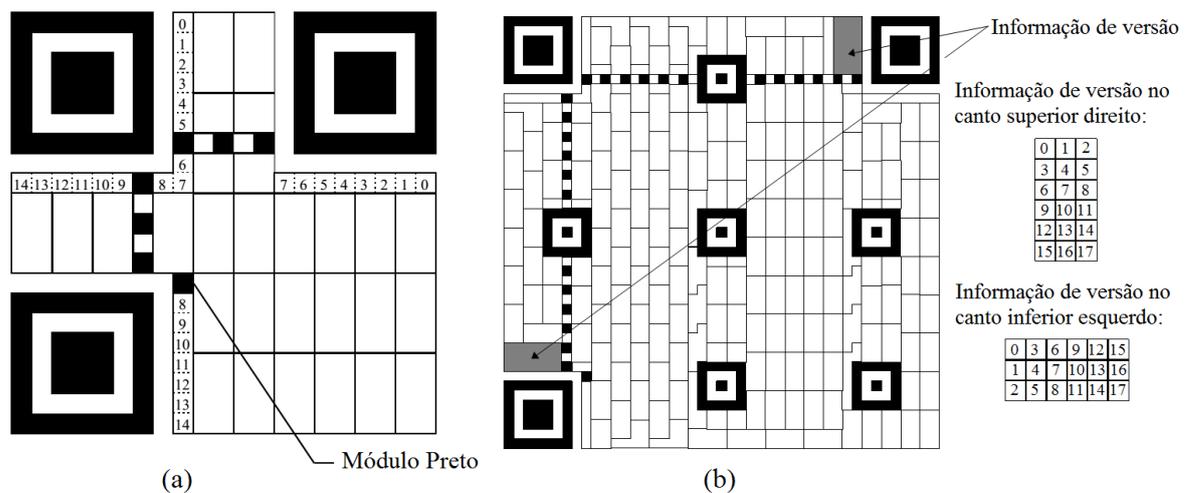


Figura 2.7 – Alguns detalhes sobre o padrão QR Code: (a) informação de formato, (b) informação de versão (adaptado de (ISO, 2006a)).

### • Informação de Versão

É uma sequência de seis<sup>14</sup> bits de informação e doze de redundância calculados pelo código BCH (18,6). Prevê redundância espacial, alocando os 16 bits em duas posições distintas do QR Code e na sequência apresentada na Figura 2.7 (b). A informação de versão está presente apenas para as versões 7 ou superiores e cuja *bitstream* pode ser encontrada na Tabela D.1 da ISO/IEC 18004:2006.

É importante salientar que a informação de versão sozinha não determina o tamanho de impressão final (expresso em polegada ou centímetro quadrado). O tamanho impresso também depende dos parâmetros de *hardware*, ou seja, da resolução de impressão e da quantidade de pontos de impressão que compõem cada célula de cor (QUERINI; ITALIANO, 2013).

<sup>13</sup> Portanto, três bits são necessários.

<sup>14</sup> 40 versões requerem seis bits para sua representação.

### 2.2.2 Funções do Padrão

As funções do padrão determinam estruturas invariantes nas diferentes versões e níveis de correção de erro do QR Code. São elas: módulos de localização (*finder patterns*), separador (*separator*), área de fronteira (*quiet zone*), linhas de espaçamento (*timing patterns*) e módulo de alinhamento (*alignment patterns*).

- **Módulos de Localização**

No QR Code existem três módulos de localização nos cantos direito superior, esquerdo inferior e superior. Cada módulo de localização pode ser visto como três quadrados concêntricos com dimensões  $7 \times 7$ ,  $5 \times 5$  e  $3 \times 3$ . A largura dos módulos obedece à razão 1:1:3:1:1, conforme ilustrado na Figura 2.8. Os quadrados concêntricos são dispostos de tal maneira que permitem a localização do *finder pattern* ao longo de  $360^\circ$ , como por exemplo nas direções (a), (b) e (c) da Figura 2.8.

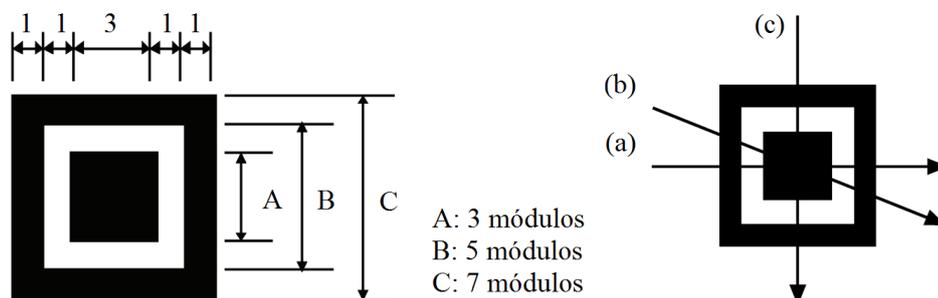


Figura 2.8 – Módulo de localização do QR Code.

- **Separador**

É uma linha ou coluna de módulos claros entorno do padrão de localização. Serve para facilitar a identificação dos quadrados concêntricos dos módulos de localização.

- **Área de Fronteira**

Trata-se de uma margem em volta dos quatro lados do símbolo para leitura do QR Code. Através da *quiet zone* torna-se mais fácil extrair o símbolo a partir da imagem adquirida pela câmera.

- **Linhas de Espaçamento**

Os espaçamentos horizontal e vertical consistem em linhas ou colunas que alternam módulos claros e escuros, começando e terminando com o módulo escuro. Esta estrutura permite que a densidade e a versão do símbolo sejam determinadas, bem como auxilia a correção da coordenada central dos módulos quando o símbolo está distorcido. Além disso, as linhas de espaçamento fornecem uma informação secundária na identificação do símbolo, quando um dos módulos de localização está parcialmente sujo ou danificado (Yue Liu; Ju Yang; Mingjun Liu, 2008).

- **Módulos de Alinhamento**

Os padrões de alinhamento estão presentes na versão 2 ou superiores (ISO, 2006a). A exemplo dos módulos de localização, os módulos de alinhamento são constituídos de três quadrados concêntricos distribuídos em conjuntos de  $5 \times 5$  módulos escuros,  $3 \times 3$  claros e um único módulo escuro central. O número e a posição dos módulos de alinhamento são apresentados no Anexo E da ISO/IEC 18004:2006. A correção de distorções, especialmente as não lineares, é função dos módulos de alinhamento. A correção emerge a partir da localização do módulo central do alinhador.

### 2.2.3 Codificador do QR Code

Simplificadamente, o esquema de codificação do QR Code é apresentado na Figura 2.9. Na etapa de análise de dados estes são examinados a fim de identificar qual o modo mais eficiente para a codificação, qual seja: numérico, alfanumérico, byte, ou Kanji (em ordem decrescente de eficiência). A seleção é baseada na análise caractere por caractere da *string* de entrada.

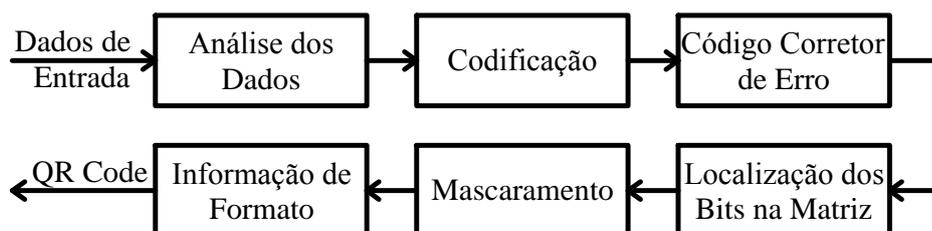


Figura 2.9 – Codificador padrão do QR Code.

Na codificação dos dados a *string* de entrada, em determinado modo, é convertida para a menor *string* de bits. Através do conhecimento prévio da capacidade máxima<sup>15</sup> de caracteres de cada versão (Tabela 7 da ISO/IEC 18004:2006) é possível determinar qual a versão mais otimizada para codificar os dados de entrada.

A codificação propriamente dita se inicia com a especificação dos quatro bits do indicador de modo, já conhecido da etapa inicial e em concordância com a Tabela 2.2.

Tabela 2.2 – Indicadores de modo.

<b>Modo</b>	<b>Indicador de Modo</b>
Numérico	0001
Alfanumérico	0010
Byte	0100
Kanji	1000

Em seguida, a *string* deve informar o número de caracteres dos dados de entrada (*character count indicator* - CCI). Para isso, o valor do número de caracteres é convertido para binário e preenchido com zeros até completar o número de bits apresentado na Tabela 2.3. Esta *string* é inserida à direita do indicador de modo.

Tabela 2.3 – Indicador de contagem de caractere.

<b>Versão</b>	<b>Numérico</b>	<b>Alfanumérico</b>	<b>Byte</b>	<b>Kanji</b>
1 ~ 9	10	9	8	8
10 ~ 26	12	11	16	10
27 ~ 40	14	13	16	12

Após as duas etapas anteriores, os caracteres de entrada são convertidos para binário (cada modo possui um método particular de conversão). A *string* de dados é adicionada à direita do número de caracteres. Por fim, separa-se a *string* de bits em *codewords*. Caso seja necessário completa-se com zeros até que a sequência de bits torne-se múltipla de oito. Mesmo assim, se o número de *codewords* não atingir o valor especificado na Tabela 7 da ISO/IEC 18004:2006 procede-se o preenchimento da *string* com os bytes “11101100” e “00010001” alternadamente, iniciando pelo primeiro.

Com a *string* de bits construída, os dados são submetidos ao algoritmo do *Reed-Solomon*. Depois da inserção das *codewords* do *RS*, algumas versões podem requerer bits “restantes”<sup>16</sup> (*remainder bits*) para completar o barcode.

<sup>15</sup> Em função do nível de correção de erro e do tipo de dado definido na etapa de análise dos dados.

<sup>16</sup> Cada versão tem especificado o número de bits restantes na Tabela 1 da ISO/IEC 18004:2006. Podem assumir o valor 3 (versões 14 ~ 20 e 28 ~ 34), 4 (versões 21 ~ 27) e 7 (versões 2 ~ 7).

### • Posicionamento na Matriz QR Code

Depois de já posicionadas as funções do padrão e as informações de versão e de formato (módulos preto e branco da Figura 2.10), ocorre a inserção da *string* binária composta pelas *codewords* de dados e do *RS*. O preenchimento de cada byte é realizado em duas colunas na sequência ordenada na Figura 2.10, correndo alternadamente em sentido ascendente para descendente e da direita para a esquerda.

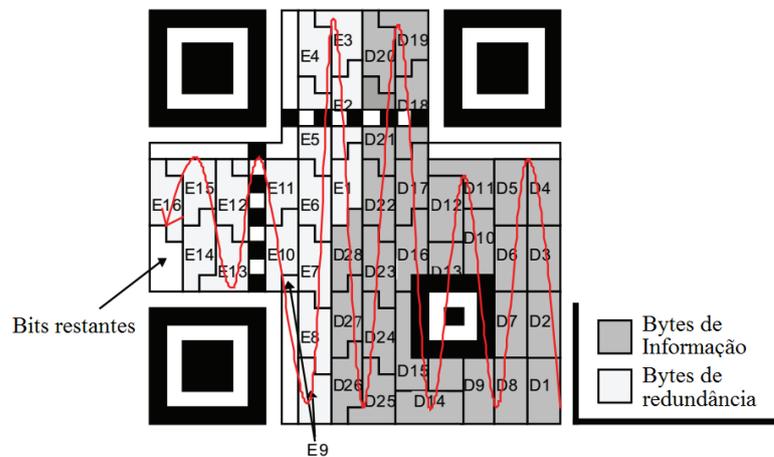


Figura 2.10 – Arranjo e sequência de preenchimento dos bytes de redundância e informação em um símbolo 2-M (adaptado de (ISO, 2006a)).

O bit mais significativo (bit “7”) de cada *codeword* deve ser posicionado no primeiro módulo disponível. O bit 7 ocupa o módulo direito inferior no sentido ascendente, e o módulo direito superior no sentido descendente, conforme ilustrado na Figura 2.11 (a). Os demais bits são alocados na sequência.

Quando a *codeword* encontra a borda superior ou inferior, por exemplo, dados D11 e D14 presentes na Figura 2.10, quaisquer bits remanescentes devem ser posicionados na próxima coluna à esquerda. Estes casos são ilustrados na Figura 2.11 (b).

Quando os bits da *codeword* atingem os módulos de alinhamento, ou as linhas de espaçamento, eles devem continuar acima, ou abaixo do padrão. Da mesma forma, quando os bits da *codeword* encontram lateralmente uma coluna do padrão de alinhamento, ou uma região ocupada pela informação de versão, os bits são posicionados em uma coluna simples adjacente ao obstáculo até que a segunda coluna esteja disponível. Essas assimetrias podem ser vistas na Figura 2.11 (c).

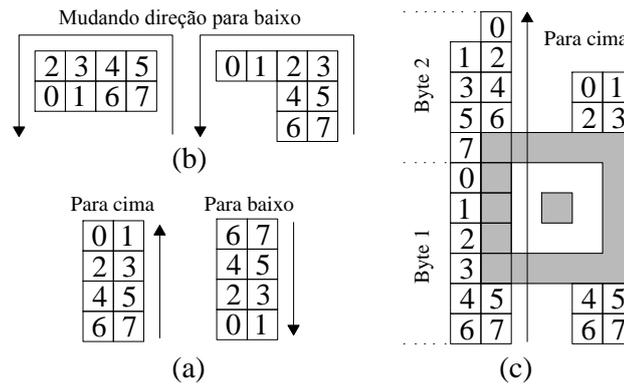


Figura 2.11 – Sequência de codificação dos bytes de redundância e de dados (adaptado de (ISO, 2006a)).

### • Padrão de Máscara

Finalmente, a região de codificação, excetuando das regiões da informação, de versão e de formato, é submetida à operação de mascaramento. O padrão de máscara implementado tem o objetivo de arranjar os módulos pretos e brancos de uma maneira balanceada e, particularmente, evitar regiões com sequência binária “1011101” correspondentes aos módulos de localização.

Os oito padrões de máscara modificam a matriz de saída baseados em sua coordenada<sup>17</sup>  $(i, j)$ . As fórmulas definidas na Tabela 2.4 determinam se a cor do módulo deve mudar ou não por meio de uma operação XOR entre o bit resultante da fórmula e o bit  $(i, j)$  da matriz. Dito de outra forma, se para determinada coordenada o bit for “1” e o resultado da fórmula for igual a 0, então, é necessário modificar o valor do bit de “1” para “0”.

Tabela 2.4 – Regras para criação do padrão de máscara.

Máscara	Fórmula
000	$(i + j) \bmod 2 = 0$
001	$i \bmod 2 = 0$
010	$j \bmod 3 = 0$
011	$(i + j) \bmod 3 = 0$
100	$[(i \text{ div } 2) + (j \text{ div } 3)] \bmod 2 = 0$
101	$(ij) \bmod 2 + (ij) \bmod 3 = 0$
110	$[(ij) \bmod 2 + (ij) \bmod 3] \bmod 2 = 0$
111	$[(i + j) \bmod 2 + (ij) \bmod 3] \bmod 2 = 0$

Na Figura 2.12 é apresentado um exemplo de mascaramento dos dados originais (à esquerda), resultante da aplicação das oito regras vistas da Tabela 2.4. Observe que todas as

<sup>17</sup> A coordenada  $(i, j) = (0, 0)$  está localizada no canto superior esquerdo do QR Code.

máscaras podem ser decodificadas, mesmo que apenas uma delas seja efetivamente escolhida pelo padrão.



Figura 2.12 – Resultado da aplicação dos oito padrões de máscara do QR Code, de acordo com a Tabela 2.4.

O QR Code deve selecionar, dentre as oito máscaras, aquela que apresenta a menor penalidade. Para isso, são definidas quatro penalidades que buscam satisfazer a melhor distribuição dos módulos preto e branco no QR Code, com o intuito de facilitar a decodificação. As penalidades são apresentadas na Tabela 2.5, na qual os valores de  $N$  são  $N_1 = 3$ ,  $N_2 = 3$ ,  $N_3 = 40$  e  $N_4 = 40$ ;  $i$  é a quantidade em que o número de módulos adjacentes de mesma cor excede 5; e  $k$  corresponde ao desvio da proporção de módulos pretos entorno dos 50% esperados em passos de 5%. É importante salientar que, embora a operação de mascaramento ocorra somente na região de codificação, todo o barcode é avaliado para fins de ranqueamento nas quatro penalidades.

Tabela 2.5 – Quatro penalidades de ranqueamento das oito máscaras.

Penalidade	Característica	Condição	Pontos
1	Módulos adjacentes de mesma cor em linha ou coluna	Número de módulos = $(5 + i)$	$N_1 + i$
2	Bloco de módulos da mesma cor	Tamanho do bloco = $m \times n$	$N_2 \times (m - 1) \times (n - 1)$
3	Razão 1:1:3:1:1 em linha e coluna (padrão de localização), precedido ou seguido de 4 módulos brancos	Existência do padrão	$N_3$
4	Proporção entre módulos brancos e pretos	$50 \pm (5 \times k)\%$ a $50 \pm (5 \times (k + 1))\%$	$N_4 \times k$

O padrão QR Code irá selecionar a máscara que apresentar o menor valor do somatório

de pontos das quatro penalidades listadas na Tabela 2.5. Na Figura 2.12 a máscara que obtém o menor valor de penalidade é a máscara 7. Conhecendo-se a informação da máscara que será utilizada, codifica-se a informação de formato, como já foi visto.

E, como etapa final de codificação, as informações de versão e formato são codificadas nas regiões padronizadas já contempladas na Figura 2.7.

#### 2.2.4 Decodificador do QR Code

O processo de decodificação do QR Code se inicia a partir de sua captura por uma câmera digital (OHBUCHI; HANAIZUMI, 2004), seguindo a partir daí de maneira inversa à codificação. Tal processamento é ilustrado na Figura 2.13. A etapa inicial busca os módulos de localização do QR Code a fim de identificar seu posicionamento. Depois, uma matriz de zeros e uns é extraída a partir do reconhecimento de módulos pretos e brancos. Nesta etapa, a diferenciação dos módulos claros e escuros ocorre pela definição de um limiar global ( $L$ ), expresso por (KIKUCHI; FUJIYOSHI; KIYA, 2013):

$$L = \frac{L_{max} + L_{min}}{2}, \quad (2.5)$$

em que  $L_{max}$  e  $L_{min}$  são as luminâncias máxima e mínima, respectivamente, da imagem capturada.

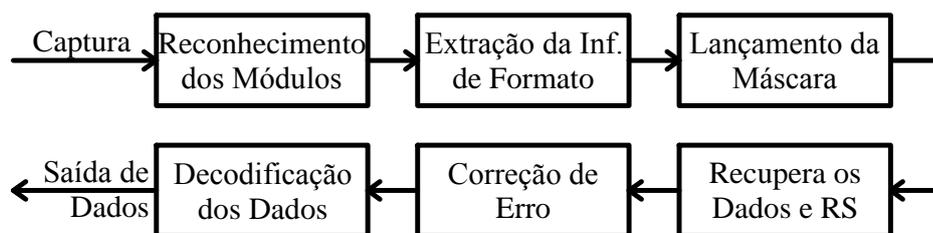


Figura 2.13 – Decodificador padrão do QR Code.

A identificação dos módulos de localização e alinhamento permite a construção de uma grade, constituída pela intersecção das retas que cruzam os pontos centrais dos módulos de alinhamento. Esta grade atua de forma a minimizar os danos causados pela inclinação entre o barcode imageado e o sensor da câmera. Desta maneira, a amostragem dos módulos se torna mais exata.

Em segundo lugar a informação de formato e de versão (quando aplicável) devem ser lidas. Nas duas situações, caso seja necessário, deve-se utilizar os bits de redundância alocados

em cada *string*. Extraindo-se a máscara codificada na informação de formato, operações de XOR são aplicadas entre os bits da região de codificação e os dados da máscara de referência, uma vez conhecidas as regras da Tabela 2.4.

Em seguida, os módulos do QR Code são lidos, de acordo com as regras de alocação da região de codificação. Em caso de erros na transmissão, estes são detectados e corrigidos de acordo com o nível de correção de erro utilizado. Por fim, os dados são divididos em função do indicador de modo e do CCI para que, finalmente, os caracteres sejam decodificados e os dados originais recompostos.

### 2.3 Distorções no Canal da Imagem

Durante a captura do barcode a imagem poderá sofrer a influência de vários fatores que podem reduzir a qualidade da imagem (XU et al., 2012). Nesse sentido, alguns desafios necessitam ser superados para que a imagem seja corretamente decodificada (LIN; FUH, 2013):

1. O barcode está sob um fundo não uniforme e submetido a uma iluminação desigual;
2. O ângulo de captura da imagem pode causar distorções perspectivadas.

Além desses dois problemas apresentados, (TIWARI et al., 2014a) acrescentam a degradação devido ao borramento (*blurring*) da imagem e ao ruído (*noise*). Os dois tipos principais de borramento são devido a imagem fora de foco (*out-of-focus blur*) e ao movimento entre câmera e o objeto (*motion blur*). Devido a estes problemas, a Figura 2.14 considera que o processo de impressão-digitalização pode ser analisado como um sistema de comunicação, na qual a informação é transmitida através do barcode, estando este sujeito aos erros causados pelo ruído no canal de imagem.

#### 2.3.1 Modelo de Degradação da Imagem

O modelo de degradação da imagem formulado neste trabalho pode ser descrito como uma função de degradação  $h(x, y)$  que, conjuntamente ao ruído aditivo  $\eta(x, y)$ , atua sobre uma imagem de entrada  $f(x, y)$  produzindo uma imagem degradada  $g(x, y)$ . Esta imagem degradada pode ser representada no domínio espacial por (MOGHADDAM, 2007), (GONZALEZ; WOODS, 2010):

$$g(x, y) = h(x, y) \star f(x, y) + \eta(x, y), \quad (2.6)$$

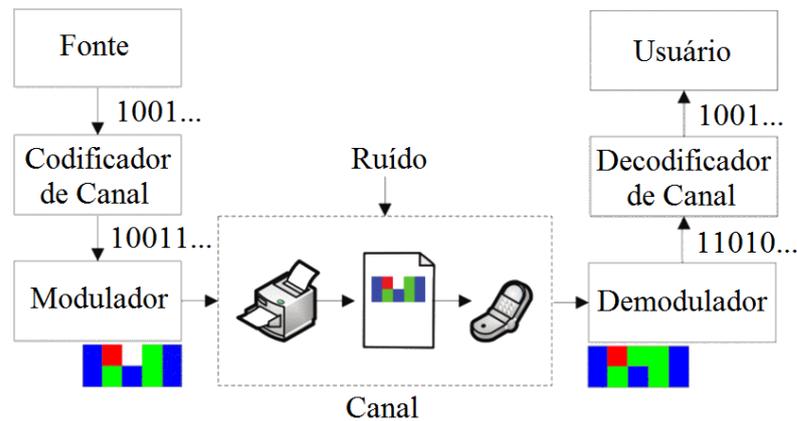


Figura 2.14 – Processo de impressão e escaneamento como parte de um sistema de comunicação (adaptado de (Vizcarra Melgar et al., 2012)).

na qual o símbolo  $\star$  indica convolução.

A função de degradação  $h(x, y)$  é determinada pelo *hardware* que constitui o sistema geral de imageamento (microscópio, telescópio, lente de câmera, etc.). Já as principais fontes de ruído  $\eta(x, y)$  em imagens digitais surgem durante a digitalização e/ou transmissão das imagens (GONZALEZ; WOODS, 2010). Desta forma, o ruído está associado a captura eletrônica da imagem por um sensor, sendo comumente atrelado aos efeitos da temperatura, quantização, detecção de fótons e perda de bits na transmissão (ruído do tipo sal e pimenta).

Apesar de certas limitações físicas, a função de degradação permite algum controle sobre si, mas que se torna fixo uma vez que o sistema tenha sido projetado. Em contraste, o ruído é de natureza tipicamente estocástica, produzindo flutuações randômicas indesejadas no sinal detectado. Assim, não há como controlá-lo nem prevê-lo (SOLOMON; BRECKON, 2011).

A modelagem do ruído é baseada no comportamento estatístico dos valores de intensidade, caracterizados por funções densidade de probabilidade (*probability density function - pdf*). As *pdfs* mais comuns são a *Gaussiana*, *Rayleigh*, *Erlang (gama)*, exponencial, uniforme e impulsiva (sal e pimenta). Oportunamente, ressalta-se que a corrida por mais pixels em um sensor, torna-os menor e, portanto, muito mais susceptíveis ao ruído, afinal, capturam menos fótons em determinado tempo de exposição.

Por outro lado, a função  $h(x, y)$  representa a resposta ao impulso do sistema de imageamento, no caso de imagens, é a resposta a um ponto. Por esse motivo,  $h(x, y)$  é chamada de função de espalhamento de ponto (*point spread function - psf*), pois um ponto no plano do objeto se espalha em muitos pontos no plano da imagem. As *psfs* descrevem funções de borra-

mento (*motion blur* e *out-of-focus blur*) e as distorções radiais na lente (*barrel* e *pincushion*). As funções de degradação podem ser matematicamente modeladas ou, então, obtidas por observação e experimentação.

### 2.3.1.1 Borrramento por Imagem Fora de Foco

Imagens borradas por estarem fora de foco (*out-of-focus blur*) ocorrem quando o foco da câmera é fixo, quando o barcode está muito próximo ou, ainda, quando a câmera está focando um fundo distante, no qual o barcode é pequeno (SÖRÖS; FLÖRKEMEIER, 2013). Essas imagens são borradas em função do algoritmo de focagem automática dos celulares que focará o plano do maior objeto (KAO; CHIU; YANG, 2009). Nestas situações em que o barcode está fora de foco do sistema óptico, a imagem é borrada através da convolução pela *psf*. Uma vez que a imagem esteja borrada, torna-se bastante difícil extrair o barcode, já que as cores preta e branca dos padrões de localização do símbolo são afetadas (LIU et al., 2013).

O grau de desfocagem depende do diâmetro efetivo da lente (limitada pela abertura circular) e da distância entre o objeto e a câmera (KAO; CHIU; YANG, 2009). A abertura circular da imagem de qualquer ponto de um objeto é um pequeno disco, conhecido como círculo de confusão (*circle of confusion* - *COC*). Na prática um objeto é dito em foco se o diâmetro do COC no plano da imagem é menor que o tamanho de um pixel no sensor, ou da resolução do olho humano. Caso contrário, a imagem será borrada, pois os raios de luz refletidos por um objeto e que atravessam a lente, não desembocam na distância focal ( $F$ ) do sistema óptico. A lei básica que governa a formação da imagem por refração através das lentes pode ser descrita por

$$\frac{1}{O} + \frac{1}{I} = \frac{1}{F}, \quad (2.7)$$

em que  $O$  e  $I$  são as distâncias entre a lente e o objeto e entre a lente e a imagem, respectivamente; e  $F$  é o comprimento focal.

Em geral, a desfocagem causada por um sistema de captura com abertura circular é modelada com um disco uniforme, de raio  $R$  (MOGHADDAM, 2007), sendo definida por

$$h(x, y) = \begin{cases} \frac{1}{\pi R^2}, & \text{se } \sqrt{x^2 + y^2} \leq R, \\ 0, & \text{caso contrário.} \end{cases} \quad (2.8)$$

Na modelagem da função de desfocagem, há o interesse de se definir  $R$  como o raio de COC. Neste modelo, considera-se que o valor da distorção fora do disco COC é aproximadamente nulo.

### 2.3.1.2 Borramento por Movimento

O borrão por movimento (*motion blur*) caracteriza-se pela aquisição de uma imagem borrada por movimento entre o sensor e o alvo (XU et al., 2012). Pode ser modelado por uma *psf* que consiste em dois parâmetros (TIWARI; SHUKLA; SINGH, 2013): ângulo e comprimento.

Vários tipos de *motion blur* podem ser distinguidos, dentre eles a translação, a rotação, a escala, ou alguma combinação destes. Aqui, somente o caso mais importante de translação será considerado, no qual o objeto, ou câmera, translada em um velocidade relativa constante  $V$ , durante um intervalo de exposição  $[0, T]$  e em ângulo  $\alpha$  radianos em relação eixo horizontal. Expressando o movimento como  $L = V \times T$ , a função *psf* no domínio espacial pode ser modelada como:

$$h(x, y) = \begin{cases} \frac{1}{L}, & \text{se } \sqrt{x^2 + y^2} \leq \frac{L}{2} \text{ e } \frac{x}{y} = -\tan \alpha, \\ 0, & \text{caso contrário.} \end{cases} \quad (2.9)$$

Enfim, (TIWARI et al., 2014b) destacam que os borramentos por desfocagem (*psf*  $a(x, y)$ ) e por movimento (*psf*  $b(x, y)$ ) podem coexistir. Neste caso, a degradação fica definida por

$$h(x, y) = a(x, y) \star b(x, y). \quad (2.10)$$

### 2.3.1.3 Distorções Radiais

Uma importante propriedade do modelo *pinhole*<sup>18</sup> para a projeção de câmera é que linhas 3D na cena são projetadas em linhas 2D. Infelizmente, as lentes, especialmente as de baixo custo, podem introduzir deficiências na transmissão de linhas retas, tornando proibitivo a utilização do modelo *pinhole*. Essas inconsistências na transmissão de linhas retas são chamadas de distorções radiais que podem ser de dois tipos: distorções de “barril” (*barrel*) e de “almofada” (*pincushion*) (VASS; PERLAKI, 2003), as quais podem ser observadas na Figura 2.15. A primeira ocorre quando as lentes possuem grande ampliação em relação à periferia. Já a segunda, possui um efeito inverso.

Embora as distorções radiais não sejam as únicas aberrações das lentes que alteram a geometria das imagens, elas são as mais significativas e as mais frequentes (VASS; PERLAKI, 2003). A modelagem que agora será apresentada é a mais simples e popular, tendo sido inicialmente proposta por (TSAI, 1987), no entanto, não serve para lentes do tipo *fish-eye*.

<sup>18</sup> Trata-se do modelo mais simples de uma câmera, sem lente, na qual a imagem é projetada em uma caixa que inibe a entrada de luz, passando através de um “buraco de alfinete” (*pinhole*).

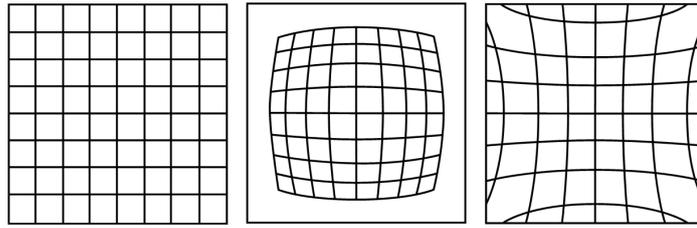


Figura 2.15 – Da esquerda para a direita: imagem original, distorção *barrel* e *pincushion* (extraída de (VASS; PERLAKI, 2003)).

Considere que o deslocamento radial das coordenadas somente modifica a distância de cada pixel ao centro da imagem. Neste sentido,  $r_u$  e  $r_d$  são as distâncias em relação ao centro das imagens não distorcida (*undistorted*) e distorcida (*distorted*), respectivamente, conforme pode ser visto na Figura 2.16. Assim,  $r_u = f(r_d)$ , em que  $f$  é função de distorção aproximada pela expansão de Taylor:

$$r_u = r_d + k_1 r_d^3 + k_2 r_d^5 + k_3 r_d^7 + \dots \quad (2.11)$$

em que  $k_i$  são os coeficientes radiais de distorção. Testes mostram que a truncagem da série de Taylor na primeira ordem é suficiente para a média das câmeras, tal que:

$$r_u = r_d + k_1 r_d^3 = r_d(1 + k_1 r_d^2). \quad (2.12)$$

A Equação 2.12 pode ser descrita em termos de  $(x,y)$ <sup>19</sup>, de acordo com a Figura 2.16, ficando em consonância com a representação matricial das imagens.

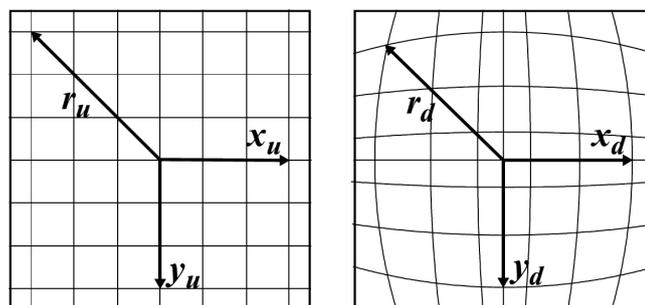


Figura 2.16 – Ilustração da distorção *barrel* (extraída de (K. T. Gribbon, C. T. Johnston, 2003)).

<sup>19</sup>  $(0,0)$  é o centro da imagem.

### 2.3.2 Distorção Perspectiva

Em uma configuração prática ideal, a forma do barcode deve ser preservada havendo, portanto, a necessidade do plano  $\pi$  da Figura 2.17 ser paralelo à câmera.

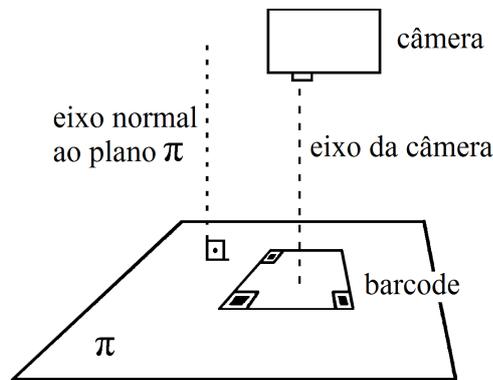


Figura 2.17 – Câmera em posição na qual o formato quadrado do barcode é preservado (adaptado de (ALFTHAN, 2008)).

A distorção perspectiva é uma distorção geométrica comum que é causada pelo fato dos sensores das câmeras digitais não estarem em paralelo com a superfície da imagem do QR Code, tal como apresentado na Figura 2.17. Desta forma, o efeito da perspectiva distorce o QR Code, como pode ser visto na Figura 2.18 (b). É interessante ressaltar que a distorção não necessariamente tem origem na câmera, mas no próprio barcode, como por exemplo, barcodes colocados em latas de refrigerante. (CHANG; CHU; CHEN, 2007).

Propositalmente os pontos A, B, C e D e seus complementares da Figura 2.18 são mantidos nos quatro cantos da imagem para ilustrar a transformação linear que mapeia um espaço de vetores  $(x,y)$  em outro  $(u,v)$ . Esse mapeamento é considerado linear, pois preserva a propriedade de multiplicação e adição entre vetores. A distorção projetiva<sup>20</sup>, caracterizada pela imagem distorcida  $P(u, v)$ , pode ser obtida pela multiplicação entre a matriz de transformação projetiva  $T$  e a imagem original  $O(x, y)$ :

$$T \times O = P \quad (2.13a)$$

$$\begin{bmatrix} A & B & C \\ D & E & F \\ G & H & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}, \quad (2.13b)$$

<sup>20</sup> A distorção perspectiva é uma aplicação da transformação projetiva.

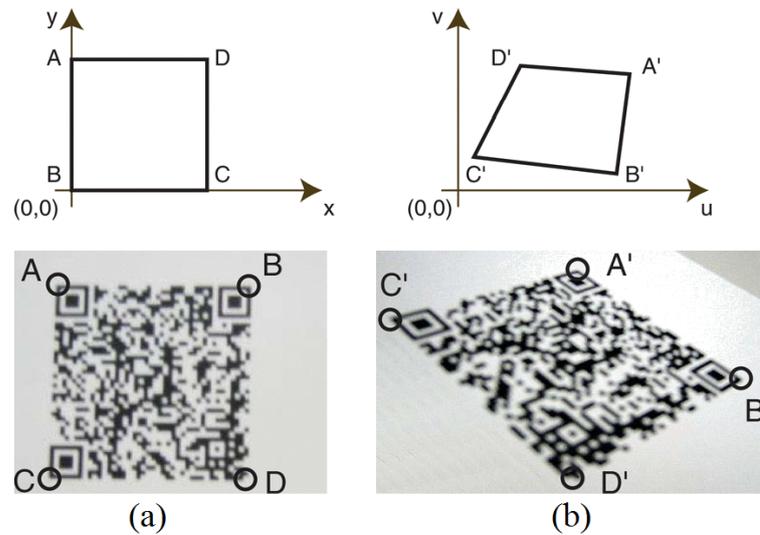


Figura 2.18 – Imagem capturada: (a) cenário da Figura 2.17, sem distorção perspectiva; (b) imagem com distorção perspectiva (adaptado de (OHBUCHI; HANAIZUMI, 2004)).

que resulta em (CHANG; CHU; CHEN, 2007):

$$u = \frac{Ax + By + C}{Gx + Hy + 1}, \quad v = \frac{Dx + Ey + F}{Gx + Hy + 1}. \quad (2.14)$$

A transformação projetiva é a mais geral das transformações lineares, da qual outras transformações que preservam paralelismo derivam. Da matriz  $T$  as seguintes transformações lineares podem ser extraídas:

$$\text{Transformação Afim: } \begin{bmatrix} A & B & C \\ D & E & F \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.15a)$$

$$\text{Rotação de } \theta \text{ graus: } \begin{bmatrix} \cos \theta & \sin \theta & 0 \\ -\sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.15b)$$

$$\text{Translação em x e y: } \begin{bmatrix} 1 & 0 & T_x \\ 0 & 1 & T_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.15c)$$

$$\text{Escala em x e y: } \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.15d)$$

### 3 CANAL COLORIDO: CONCEITOS, PADRÕES E DEFINIÇÕES

Este capítulo tem o objetivo de apresentar a (re)produção das cores, tanto na perspectiva da visão humana como em dispositivos digitais. Para isso, o comportamento espectral das cores será analisado, conjuntamente a sua inter-relação com alguns dos principais sistemas de representação de cor, os espaços de cores. Por fim, apresenta-se um estudo acerca dos problemas que acometem o canal das imagens coloridas, no âmbito das distorções cromáticas.

#### 3.1 Sensação e Representação das Cores

Em 1666, Sir Isaac Newton descobriu que quando um feixe de luz solar atravessa um prisma de vidro ele é refratado em um espectro contínuo de cores, variando do violeta ao vermelho (GONZALEZ; WOODS, 2010). As cores emergidas do prisma são apresentadas em seis grandes regiões: violeta, azul, verde, amarela, laranja e vermelha, todas elas continuamente interligadas, conforme pode ser observado no espectro da luz visível ilustrado na Figura 3.1. Na banda de energia eletromagnética o intervalo do comprimento de onda ( $\lambda$ ) da luz visível varia, aproximadamente, de 400 nm até 700 nm.

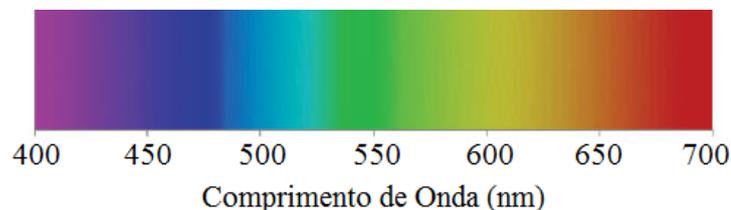


Figura 3.1 – Espectro de luz visível com as cores continuamente interligadas.

A luz é fundamental para a ciência das cores (GONZALEZ; WOODS, 2010). Se a luz for acromática seu único atributo será a *intensidade*, ou *nível de cinza*, que se refere a uma medida escalar variável desde o branco, passando pelo cinza até o preto. Já a luz cromática é descrita por três grandezas: *radiância*, *luminância* e *brilho*. A primeira é referida como a quantidade total luz emanada pela fonte, sendo, portanto, medida em *Watts* (W). Por conseguinte, a luminância, cuja medida é o *lumens* (lm), mensura a quantidade de energia que um observador percebe de uma fonte de luz. O brilho, por sua vez, é um descritor subjetivo, de difícil mensuração e que incorpora a noção acromática de intensidade.

A distinção entre cores só é possível por meio dos atributos de *brilho*, *matiz* e *satura-*

ção (GONZALEZ; WOODS, 2010). O brilho, como vimos, incorpora a noção acromática de intensidade. Já a matiz está associada ao comprimento de onda dominante em uma mistura de ondas de luz e que é percebida pelo observador. Por fim, a saturação se refere a quantidade de luz branca adicionada a uma matiz, ou à pureza da cor. Matiz e saturação juntas determinam a *cromaticidade* e, portanto, uma cor pode ser representada por seu brilho e sua cromaticidade.

Qualquer cor pode ser representada por valores *triestímulos*  $X$ ,  $Y$  e  $Z$  (também referidos como tricromaticidade) que são quantidades de vermelho, verde e azul. A tricromaticidade provê uma forte e indireta evidência para o fato do olho humano possuir três receptores coloridos (SHARMA; TRUSSELL, 1997). Este fato foi confirmado apenas muito tempo depois por estudos anatômicos e fisiológicos. Todavia, para a representação das cores por valores triestímulos em representações 2-D, tais valores necessitam ser normalizados (isto é, a soma das três coordenadas de cromaticidade deverá resultar “1”) para constituir o *diagrama de cromaticidade*. Assim, apenas duas coordenadas são utilizadas ao longo dos eixos perpendiculares. O digrama de cromaticidade mais comumente utilizado é o *digrama de cromaticidade CIE xy*, apresentado na Figura 3.2. As coordenadas do diagrama de cromaticidade CIE xyz podem ser obtidas da seguinte forma:

$$x = \frac{X}{X + Y + Z}, \quad (3.1a)$$

$$y = \frac{Y}{X + Y + Z}, \quad (3.1b)$$

$$z = \frac{Z}{X + Y + Z}. \quad (3.1c)$$

As posições das várias cores (puras) no espectro da Figura 3.1 são indicadas na borda do diagrama de cromaticidade da Figura 3.2. Qualquer ponto no interior do diagrama representa uma mistura de cores do espectro. Assim, diz-se que as cores na borda do diagrama estão completamente saturadas, e quanto mais afastado das bordas mais luz branca é adicionada à cor. No específico ponto interno ao diagrama chamado de *igual energia* será constituída a luz branca e a saturação será nula.

Traçando-se segmentos de reta que conectem três cores fixas quaisquer do diagrama da Figura 3.2 haverá a constituição de um triângulo, cujas cores interiores serão totalmente produzidas pela combinação dessas três cores. Esse conjunto de cores sintetizáveis por estas três cores é denominado *gama de cores*, ou *gamut*, em inglês. No entanto, as cores para além das fronteiras do triângulo não serão obtidas pela combinação das três cores selecionadas. Isso demonstra que não é possível reproduzir todas as cores a partir de apenas três cores primárias

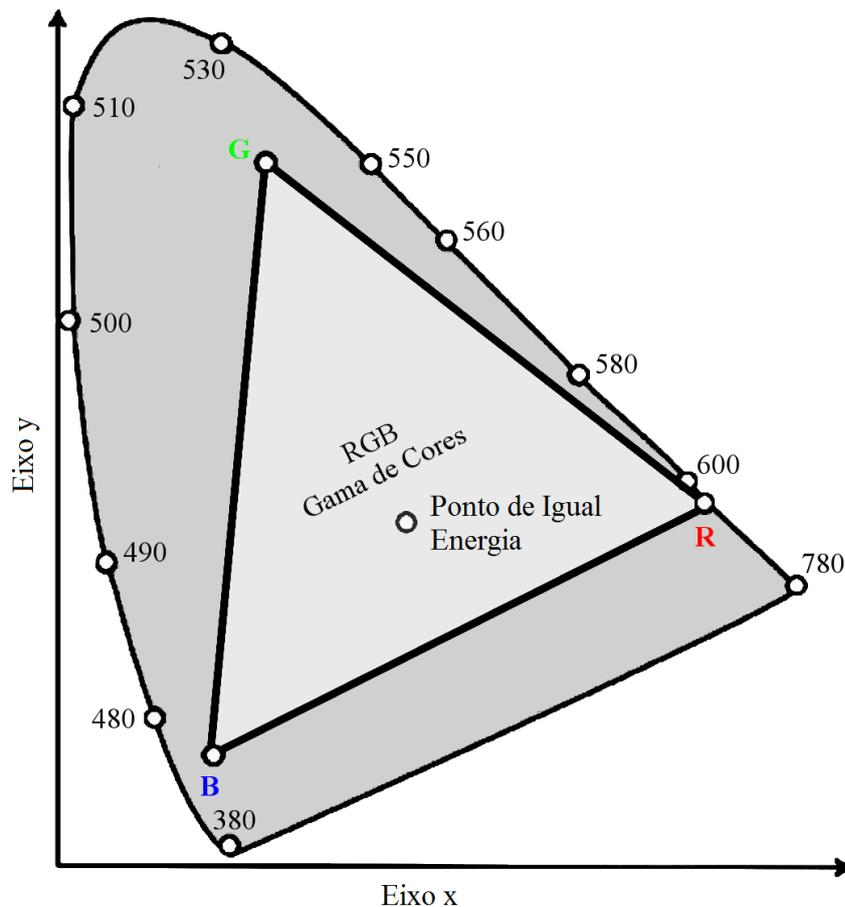


Figura 3.2 – Diagrama de cromaticidade CIE xy (valores em nanômetros).

fixas (GONZALEZ; WOODS, 2010). Essa observação é oportuna para análise do sistema de cores RGB, baseado no triângulo delimitado pelas cores vermelho, verde e azul, conforme pode ser visto na Figura 3.2.

### 3.1.1 Formação de Imagem no Olho Humano

Os fotossensores das câmeras digitais são projetados para capturar imagens pessoais, buscando promover a melhor excitação das células óticas do olho humano. No entanto, devido à onipresença das câmeras digitais em dispositivos celulares, a aquisição de barcodes deve se adequar às características de captura da câmera. Portanto, nesta seção, faz sentido promover a compreensão da fisiologia do olho humano, já que sua característica determina, por outro lado, as propriedades dos sensores digitais de aquisição de imagens. Desta maneira, será possível atingir um melhor entendimento das restrições dos barcodes imageados por estes sensores.

A luz refletida em uma superfície atinge a lente do *crystalino* (Figura 3.3 (a)) que permite

a focalização da imagem em diferentes distâncias, por meio da variação de seu formato. Os raios de luz focalizados pelo cristalino se chocam com a parte posterior do olho, onde fica a *retina* que é dotada de duas células fotossensíveis, os *cones* e os *bastonetes*. Os cones são responsáveis pela sensação das cores, enquanto que os bastonetes são células sensíveis à energia da luz, não produzindo nenhuma informação sobre a cor (SHIRLEY; MARSCHNER, 2009).

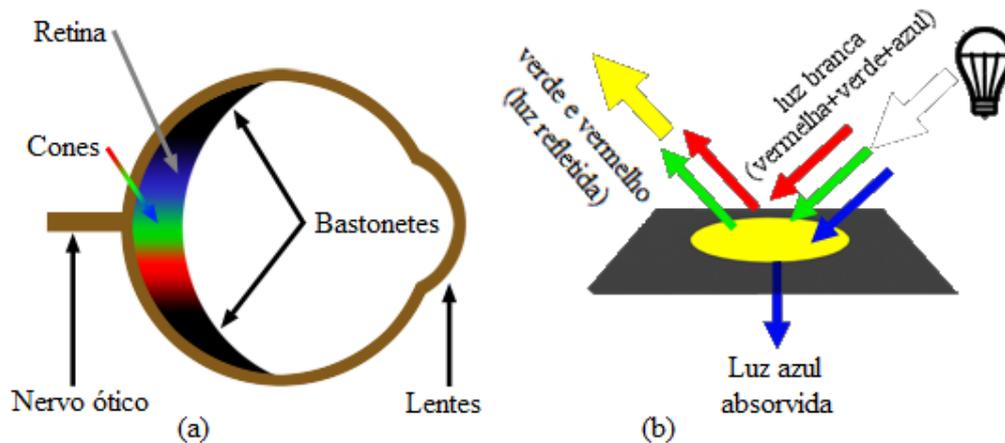


Figura 3.3 – Processo de formação da imagem no olho humano: (a) Estruturas do olho sensíveis à luz refletida e exemplificada em (b) (adaptado de (SCRATCHAPIXEL, 2012)).

Na Figura 3.4 são apresentadas as curvas experimentais médias detalhando a absorção de luz pelos cones e bastonetes. Observe que cada cone responde sobre comprimentos de onda suavemente diferentes. São chamados tipicamente de receptores com comprimento de onda longo, médio e curto, ou de maneira menos precisa, como sensores vermelho, verde e azul. As cores são vistas como combinações das chamadas *cores primárias*: vermelho (R, *red*), verde (G, *green*) e azul (B, *blue*) que possuem picos de absorção de energia, aproximadamente, em 560, 530 e 420 nm, respectivamente (RUSS, 2011). Já os bastonetes são mais sensíveis a luz em  $\lambda = 495$  nm, conforme ilustrado pela curva pontilhada na Figura 3.4.

Em virtude das diferentes densidades dos cones vermelho, verde e azul, a sensibilidade total do olho é maior para o verde<sup>21</sup> claro e menor para o azul claro (RUSS, 2011), o que pode ser graficamente entendido na Figura 3.5. No entanto, a habilidade do olho humano em distinguir uma cor da outra nessa região do verde claro é menor, se comparado às redondezas do azul claro.

<sup>21</sup> É por esse motivo que a maioria dos sensores digitais CCD e CMOS têm o dobro de detectores verde em detrimento ao vermelho e ao azul. Evidentemente, essa característica afeta o desempenho de detecção dos barcodes que, em princípio, não necessitam ser visualizados depois de digitalizados.

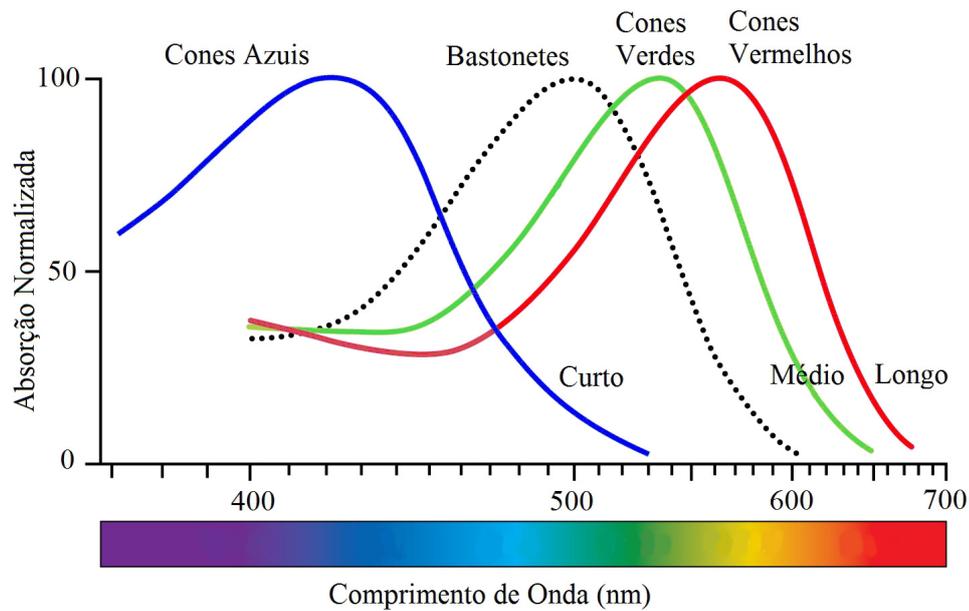


Figura 3.4 – Comparação entre os picos de sensibilidade e absorção de energia pelas células ópticas no espectro de luz visível (adaptado de (OpenStax College, 2013)).

A padronização das cores primárias foi realizada em 1931 pela CIE (*Commission Internationale de l'Éclairage* - Comissão Internacional de Iluminação) para os seguintes valores específicos: vermelho ( $\lambda = 700$  nm), verde ( $\lambda = 546,1$  nm) e azul ( $\lambda = 435,8$  nm). Essa padronização ocorreu antes mesmo do traço das curvas experimentais da Figura 3.4 (GONZALEZ; WOODS, 2010). As cores primárias podem ser adicionadas para produzir as cores *secundárias*, sintetizando o magenta (M), ciano (C) e amarelo (Y). Essa síntese é ilustrada na Figura 3.6 (a). Já na Figura 3.6 (b) estão as cores primárias de pigmentos, ou corantes. Uma cor primária de pigmento é definida pela subtração (ou absorção) de uma cor primária de luz e pela reflexão (ou transmissão) das outras duas. Um exemplo desse fenômeno pode ser visto na Figura 3.3 (b), na qual a luz amarela é produzida pela reflexão do verde e do vermelho e pela absorção do azul.

O fato de existir apenas três tipos de fotossensores sensíveis a cores na retina humana simplifica enormemente a tarefa de exibir cores em telas (SHIRLEY; MARSCHNER, 2009). Por esse motivo, as cores em telas são apresentadas em combinações ponderadas das três cores primárias vermelha, verde e azul, em constituição ao padrão RGB, o qual será apresentado posteriormente.

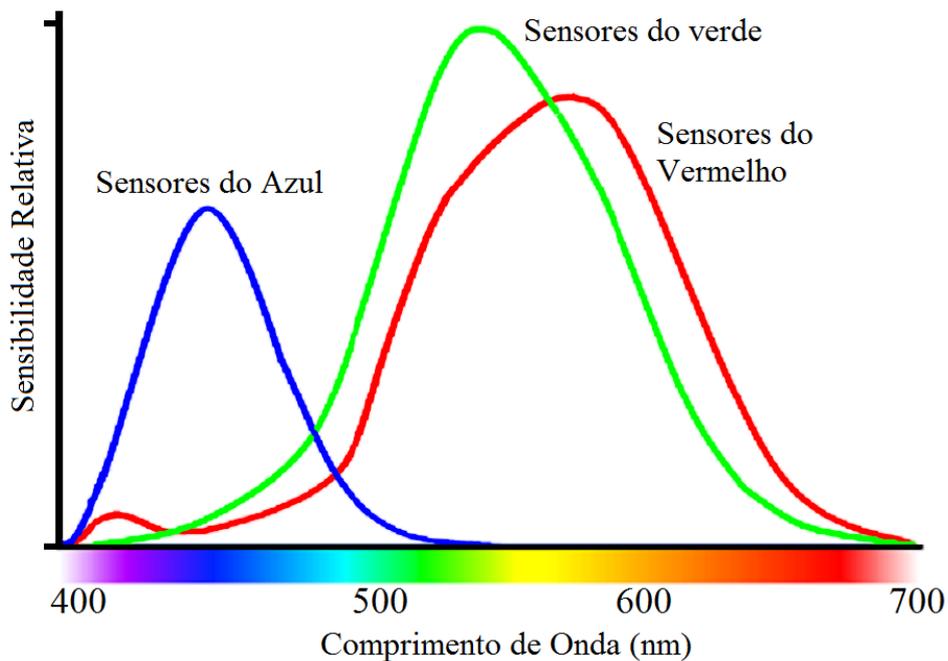


Figura 3.5 – Comparação entre os diferentes valores de sensibilidade e absorção de energia pelas células óticas no espectro de luz visível.

### 3.1.2 Espaço de Cores

Os *espaços, modelos, ou sistemas* de cores nada mais são do que a representação padronizada das cores em seu formato discretizado. Para que seja possível codificar bits em cores, os dispositivos convertem a cor, que é fisicamente um estímulo contínuo, em três dimensões. Em seguida, esse estímulo espectralmente distribuído em três dimensões é acomodado em um espaço de cor. Os modelos de cores são geralmente orientados ao *hardware*. Dentre eles, pode-se citar os padrões RGB e CMYK, utilizados em monitores e impressoras, respectivamente. Para atender aos interesses deste trabalho, são apresentados a seguir os sistemas de cores RGB, CMY e HSI. O sistema YCbCr será oportunamente descrito na Subseção 4.2.7.

#### 3.1.2.1 Espaço de Cores RGB

No modelo de cores RGB cada cor aparece em seus componentes espectrais primários de vermelho (R), verde (G) e azul (B), catalogados em três vértices da Figura 3.7 (a). Completam os vértices do cubo unitário as cores secundárias ciano (C), magenta (M), amarelo (Y), preto (K) e branco (W). O preto e o branco situam na origem e no vértice mais distante a ela,

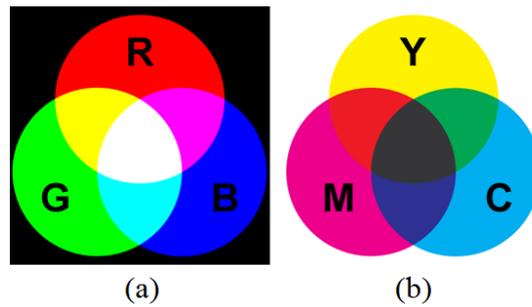


Figura 3.6 – Cores primárias e secundárias de luz (a) e de pigmentos (b).

respectivamente. Por exemplo, em uma imagem RGB, na qual cada canal de cor do pixel possui 8 bits (total 24 bits), é possível representar  $(2^8)^3 = 16.777.216$  cores.

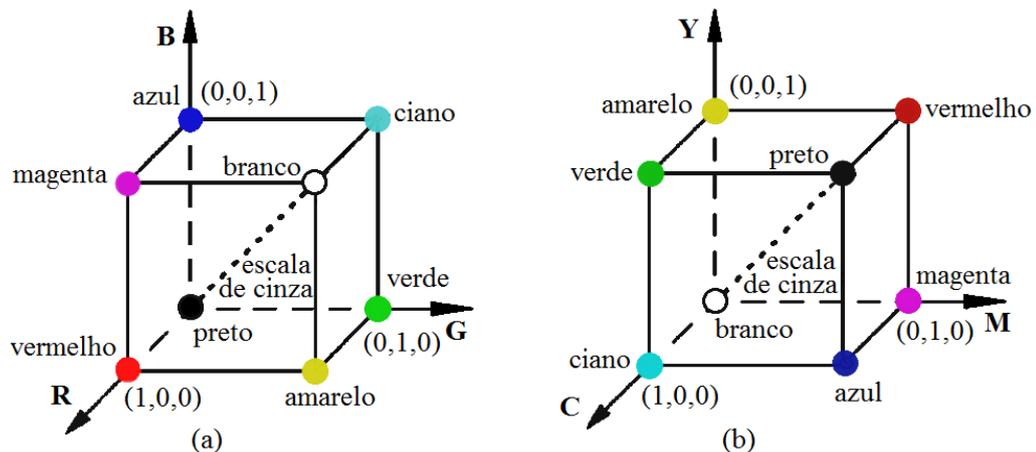


Figura 3.7 – Espaço de cores: (a) RGB e (b) CMY.

A diagonal que interliga os vértices branco e preto constitui a escala de cinza (*intensidade* monocromática), na qual as três cores primárias possuem o mesmo valor. Também é possível observar que a saturação (pureza) de uma cor aumenta com o afastamento da diagonal de intensidade (no eixo de intensidade a saturação é nula).

### 3.1.2.2 Espaço de Cores CMY e CMYK

As cores ciano, magenta e amarelo são cores secundárias de luz ou, de outra forma, cores primárias de pigmentos, que compõem o espaço de cores CMY demonstrado na Figura 3.7 (b). Por exemplo, quando sobre uma superfície amarela incide luz branca, a porção azul da luz é absorvida, enquanto que as porções verde e vermelho são refletidos, e cuja combinação

resulta no amarelo. Este fenômeno é ilustrado na Figura 3.3 (b).

Como vimos, os dispositivos que depositam os corantes sobre o papel requerem a entrada de dados no padrão CMY, ou devem realizar internamente a seguinte conversão:

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \quad (3.2)$$

Em consenso com a Figura 3.6 (b) magnitudes iguais de ciano, magenta e amarelo devem produzir o preto. Muito embora essa combinação seja factível, na prática revela-se limitada. Sendo assim, geralmente, emprega-se o padrão a quatro cores CMYK, com a inserção do corante preto ao padrão CMY.

*Nota:* Baseado na utilização do espaços de cores RGB e CMY e, em consonância com a Equação 3.2, os sistemas de cores podem ser categorizados como aditivo, ou subtrativo (SHARMA; TRUSSELL, 1997). O sistema de cores aditivo utiliza o padrão RGB e representa as cores através da combinação aditiva das cores primárias vermelha, verde e azul. Já o sistema de cores subtrativo, por outro lado, usa o modelo de cores CMY e sintetiza as cores pela subtração do espectro indesejado da luz branca (BULAN; MONGA; SHARMA, 2009).

### 3.1.2.3 Espaço de Cores HSI

Apesar da constelação RGB se aproximar muito da estrutura sensitiva de cores do olho humano, sua descrição é bastante afastada da interpretação humana de cores (GONZALEZ; WOODS, 2010). Dito de outra forma, os seres humanos descrevem um objeto em cores pela sua matiz, saturação e brilho.

O modelo de cores HSI (*hue, saturation, intensity* - matiz, saturação e intensidade<sup>22</sup>) separa a componente de intensidade das informações de cor (matiz e saturação) de uma imagem colorida. O espaço de cores HSI se organiza em torno do eixo vertical de intensidade em formato cônico, como pode ser visto na Figura 3.8 (b). O tracejado presente, tanto no espaço de cores RGB, quanto no HSI, representa o nível de cinza. Para um vetor arbitrário  $\mathbf{p}$  a intensidade do nível de cinza é determinada pelo plano perpendicular ao eixo de I, pelo ângulo  $h$  da matiz e pela distância  $s$  (em relação ao eixo I) da saturação.

O modelo de cores HSI, na perspectiva de seus atributos, pode ser integralmente obtido a partir do cubo de cores RGB (Figura 3.8 (a)) por simples deduções geométricas. As relações

<sup>22</sup> A intensidade (nível de cinza), ao contrário da noção de brilho, é facilmente mensurável.

entre os atributos do sistema RGB para o HSI são descritas como:

$$H = \begin{cases} \theta, & \text{se } B \leq G \\ 360 - \theta, & \text{se } B > G \end{cases}, \quad (3.3a)$$

$$\text{com } \theta = \cos^{-1} \left\{ \frac{0,5[(R-G) + (R-B)]}{[(R-G)^2 + (R-B)(G-B)]^{0,5}} \right\}, \quad (3.3b)$$

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)], \quad (3.3c)$$

$$I = \frac{1}{3}(R+G+B). \quad (3.3d)$$

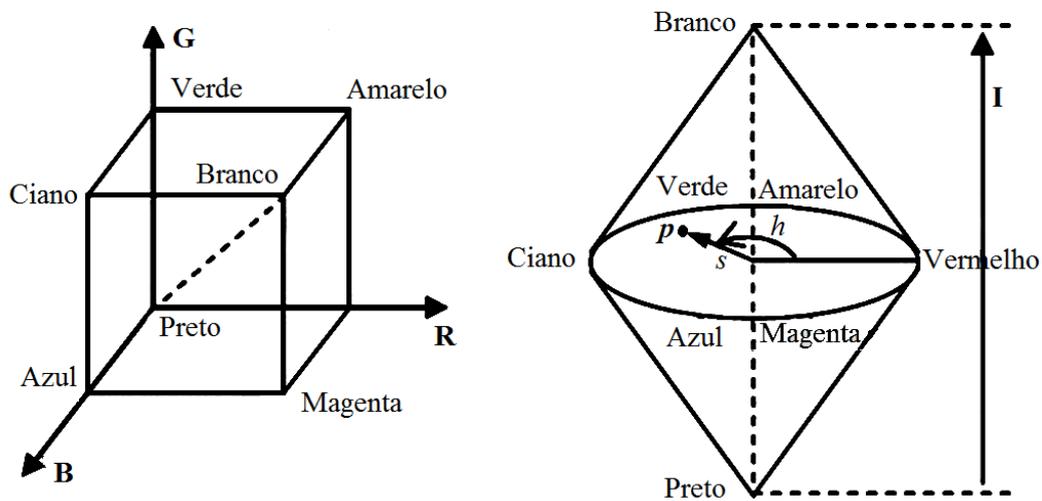


Figura 3.8 – Espaço de cores RGB (à esquerda) e HSI (à direita).

### 3.2 Separabilidade de Cores

O uso de cores para codificar bits de informação em barcodes coloridos oferece uma alternativa para o incremento da capacidade de dados, no entanto, demandam estabilidade a variações de cor nos processos de impressão e captura. Estas imperfeições na cor têm um grande impacto degenerativo quando comparadas aos processamentos de reprodução e captura dos barcodes monocromáticos (BULAN; SHARMA, 2011).

Uma das maiores dificuldades no trato de imagens com cores resultam das variações na cor RGB causadas pelos componentes do sistema de visão computacional. Nesse sentido, (TAPPAN; WRIGHT; SISTLER, 1987) identificaram e interpretaram nove diferentes fontes de erros nos sistemas de imageamento digital, com origens na fonte de luz, placa de digitalização, lentes, câmera e na combinação destes. Sem a devida calibração da cor distorcida pelos sistemas

de geração e captura da imagem, as análises que envolvem a cor podem esconder ou apresentar imprecisamente importantes características dos objetos nas imagens (CHANG; REID, 1996). Em nosso caso, os objetos em questão são os módulos/células de um barcode 2D colorido. Logo, em vez de recuperarmos a representação fidedigna da cor, estamos mais interessados na informação em bit, ou byte, embarcada em cada cor.

### 3.2.1 Interferência entre Canais de Cores

No canal de impressão-escaneamento (*print-scan*), os dados embarcados em cada canal de cor C, M e Y são extraídos no decodificador a partir do escaneamento do barcode impresso. Ao contrário dos sistemas de impressão, os dispositivos de captura, tais como câmeras digitais ou *scanners*, utilizam os canais RGB, espectralmente sensitivos, para capturar imagens coloridas. Assim, os canais C, M e Y da imagem precisam ser estimados a partir dos canais R, G e B escaneados, explorando a complementariedade entre os canais CMY e RGB. Em um cenário ideal, o escaneamento do canal vermelho somente captura informação do corante ciano, o canal verde do magenta e o azul do amarelo, conforme pôde ser visto na Equação 3.2. Em outras palavras, cada corante C, M e Y absorve sua cor complementar R, G e B, respectivamente, e transmite o resto do espectro. A extensão da absorção é determinada pela concentração do corante, sendo que o uso de diferentes concentrações sintetizam cores distintas (SHARMA; TRUSSELL, 1997).

Na prática, contudo, devido a absorções indesejadas dos corantes, os canais de cores interferem uns nos outros. A interferência entre os canais (*cross-channel*) afeta o desempenho da detecção e causa erros (BULAN; OZTAN; SHARMA, 2011). Na Figura 3.9, por exemplo, podem ser vistas as componentes de interferência no canal *print-scan*. Cada uma das fitas verticais da Figura 3.9 (a) imprime a imagem nas componentes de cores C, M e Y. O resultado do escaneamento das componentes R, G e B são mostrados nas Figuras (b)-(d), respectivamente. Observa-se que o canal verde, que idealmente captura somente a informação do corante magenta, na prática, sofre interferência do corante ciano. Similarmente, o canal azul, obtido idealmente do corante amarelo, recebe interferência significativa do canal magenta. Portanto, a interferência de cor depende tanto das características espectrais dos corantes, quanto da resposta espectral do *scanner* (BULAN; SHARMA, 2011).

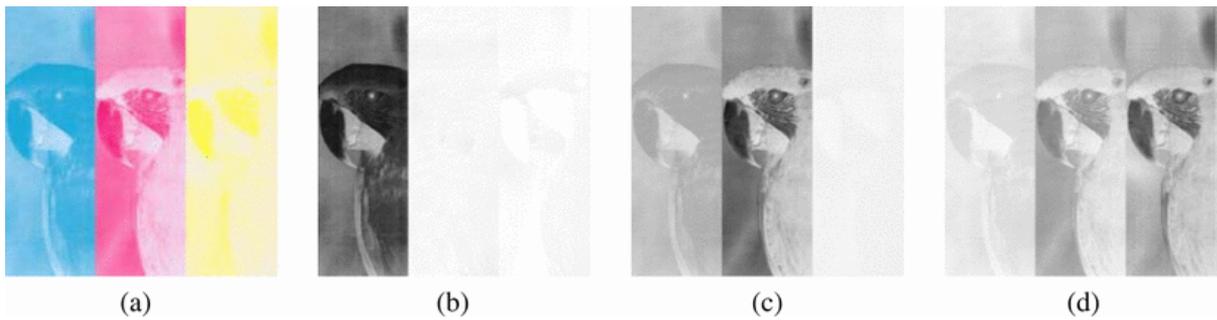


Figura 3.9 – Interferência *cross-channel* do espaço de cores CMY nos canais de escaneamento RGB: (a) CMY impresso; (b) canal R escaneado; (c) canal G escaneado; (d) canal B escaneado; (extraída de (BULAN; SHARMA, 2011)).

### 3.2.2 Distanciamento entre Cores

Para elevar a capacidade de dados de um barcode, pode-se aumentar deliberadamente a quantidade de cores codificadas. No entanto, há um *trade-off* entre o número de cores usadas para codificar informação e a robustez de leitura do barcode (KATO; TAN; CHAI, 2010), (QUERINI; ITALIANO, 2013). Quanto mais cores forem usadas em um espaço de cor, menores serão as distâncias entre elas, conduzindo a um menor contraste entre as diferentes cores. Esta redução do contraste ensejará uma dificuldade maior para distinguir as cores e de recuperar os dados originais. Os barcodes monocromáticos, por exemplo, permitem que mais células sejam codificadas em uma mesma área, quando comparados aos barcodes coloridos. Por esses motivos, o acréscimo do número de cores não conduz diretamente ao incremento da capacidade de dados do barcode. Conseqüentemente, uma cuidadosa seleção de cores codificáveis, bem como um código de detecção de cor robusto devem ser preferidos, a fim de garantir um barcode 2D otimizado para aplicações *mobile* (KATO; TAN; CHAI, 2010).

#### 3.2.2.1 Capacidade do Canal

Baseando-se no trabalho de (LIU; DOERMANN; LI, 2008) é possível estimar a capacidade do pixel utilizando a informação mútua, proveniente dos conceitos de Teoria da Informação. Neste cenário, a capacidade do canal é formalmente definida pela informação mútua:

$$I(X; Y) = \sum_{j=0}^{q-1} \sum_{i=1}^{Q-1} P(x_j) P(y_i | x_j) \log \frac{P(y_i | x_j)}{P(y_i)} \quad (3.4)$$

em que  $q$  e  $Q$  são os tamanhos dos conjuntos de símbolos (cores) de entrada e de saída, respectivamente;  $P(x_j)$  é a probabilidade da cor  $x_j$  estar sendo mostrada e  $P(y_i)$  é a probabilidade de que a cor  $y_i$  seja capturada pela câmera.  $P(y_i|x_j)$  é a possibilidade condicional de que a cor  $x_j$  seja capturada como  $y_i$ .  $P(y_i|x_j)$  é arranjado em uma matriz, chamada “matriz de transição”. Assume-se que os dados enviados através do canal de câmera são randômicos e que todos os símbolos são equiprováveis. Logo,  $P(x_j) = 1/q$  e  $P(y_i) = \sum_{j=0}^{q-1} \frac{P(y_i|x_j)}{q}$ . Portanto, a Equação 3.4 pode ser reescrita da seguinte maneira:

$$I(X; Y) = \sum_{j=0}^{q-1} \sum_{i=1}^{Q-1} \frac{1}{q} P(y_i|x_j) \log \frac{P(y_i|x_j)}{\sum_{j=0}^{q-1} \frac{P(y_i|x_j)}{q}} \quad (3.5)$$

A partir da Equação 3.5, se for conhecida a distribuição da cor capturada, torna-se possível determinar a capacidade do canal. Intuitivamente, usando um conjunto finito de cores para carregar informação, é conveniente que a diferença entre as cores capturadas seja tão distante quanto possível e que a variância da distribuição de cores seja pequena, de modo que duas cores tenham probabilidade reduzida de serem confundidas. Baseado nesses pressupostos, (LIU; DOERMANN; LI, 2008) analisam a capacidade dos canais monocromáticos e coloridos:

- **Canal Monocromático:**

A capacidade de um canal binário é afetada pela matriz de transição. Assumindo que os valores em escala de cinza capturados estão distribuídos em torno de médias  $(128 \pm d/2)$  com variância  $v$ , a capacidade do canal monocromático, baseada na Equação 3.5, é definida por  $b = \text{Capacidade}(d, v)$ , em que  $0 \leq b \leq 1$ ,  $0 \leq d \leq 255$  e  $1 \leq v \leq 100$ . O resultado é apresentado na Figura 3.10.

Como pode ser visto na Figura 3.10, a capacidade do canal é máxima quando a variância é nula e  $d$  atinge o valor 255. Portanto, deve-se buscar um elevado distanciamento entre cores e baixa variância para atingir altas capacidades (LIU; DOERMANN; LI, 2008).

- **Canal Colorido:**

Uma boa estratégia para minimizar o efeito degenerativo que atua sobre o canal da câmera é processar a escolha de  $k$  cores de modo a se maximizar a mínima distância Euclidiana entre elas. Em (LIU; DOERMANN; LI, 2008) é proposta a seleção de cores baseada no valor médio  $P_{cam}(c)$  da cor  $c$ , dentre 200 imagens capturadas por diferentes câmeras de celular. Para

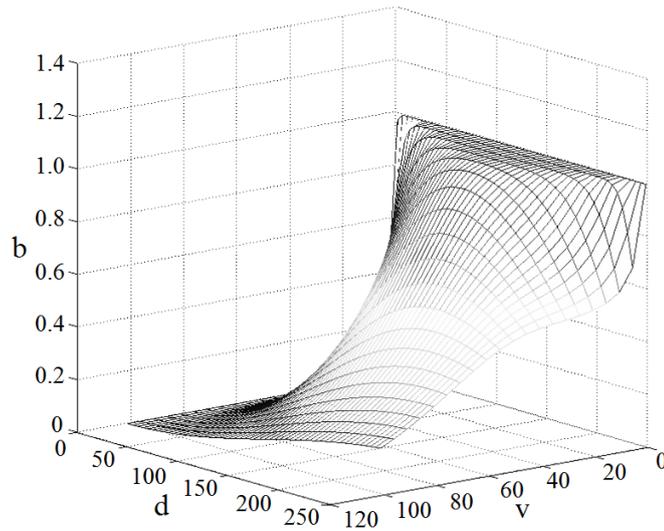


Figura 3.10 – Capacidade do canal preto e branco (adaptado de (LIU; DOERMANN; LI, 2008)).

tanto, os autores calculam a distância Euclidiana entre as cores  $c_1$  e  $c_2$  capturadas por diferentes câmeras, conforme segue:

$$dist(c_1, c_2) = \min_{cam} (|P_{cam}(c_1) - P_{cam}(c_2)|). \quad (3.6)$$

A Equação 3.6 permite selecionar as cores que estejam mais afastadas do que um limiar  $th$ , cujo valor deve decrescer até que  $k$  cores tenham sido selecionadas.

A seleção de  $k$  cores tem como objetivo maximizar a capacidade do canal. Para isso, na Figura 3.11 a capacidade do canal é avaliada para  $k$  diferentes cores, selecionadas utilizando a Equação 3.6 (linha azul), ao passo que outras  $k$  cores são randomicamente escolhidas (linha vermelha). Pode-se facilmente observar que a seleção das  $k$  cores apresenta um resultado significativamente melhor enquanto  $k < 800$  cores. Acima desse número de cores, o efeito da seleção de cores não apresenta resultados satisfatórios, uma vez que as cores estão muito próximas. Ainda, para  $k = 300$  cores o canal atinge capacidade máxima<sup>23</sup> de 7,2 bits por pixel.

### 3.2.3 Determinação da Cor em Canais Multi-Cores

Para a análise do comportamento em canais multi-cores, (MEMETI, 2012) e (LIU; DOERMANN; LI, 2008) conduzem análises similares a fim de verificar a variabilidade das cores. O primeiro cria um barcode com 4096 módulos e, em seguida, captura com uma câmera. O

<sup>23</sup> Não estão considerados o ruído e a variação da iluminação.

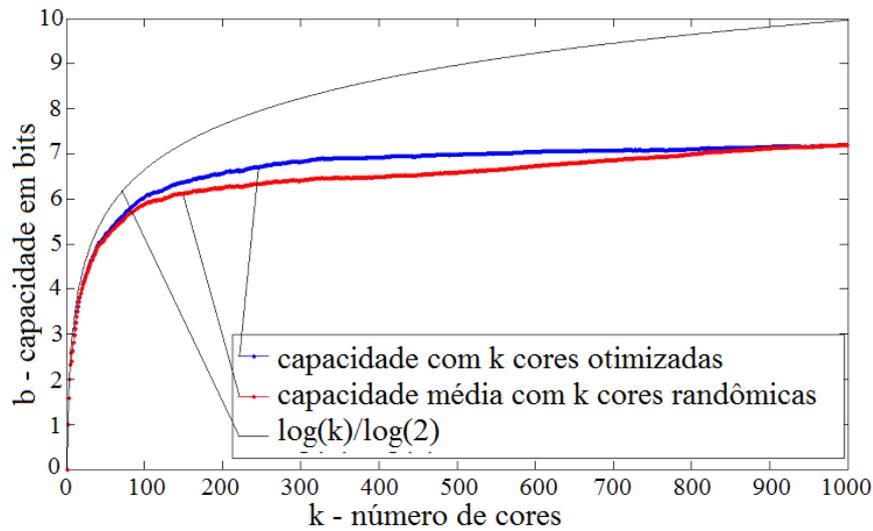


Figura 3.11 – Capacidade do canal com  $k$  cores selecionadas (adaptado de (LIU; DOERMANN; LI, 2008)).

resultado deste barcode é apresentado em oito<sup>24</sup> agrupamentos de cores na Figura 3.12 (a). Já o segundo escolhe 2000 cores randomicamente e as amostra múltiplas vezes. A distribuição de seis delas é apresentada na Figura 3.12 (b).

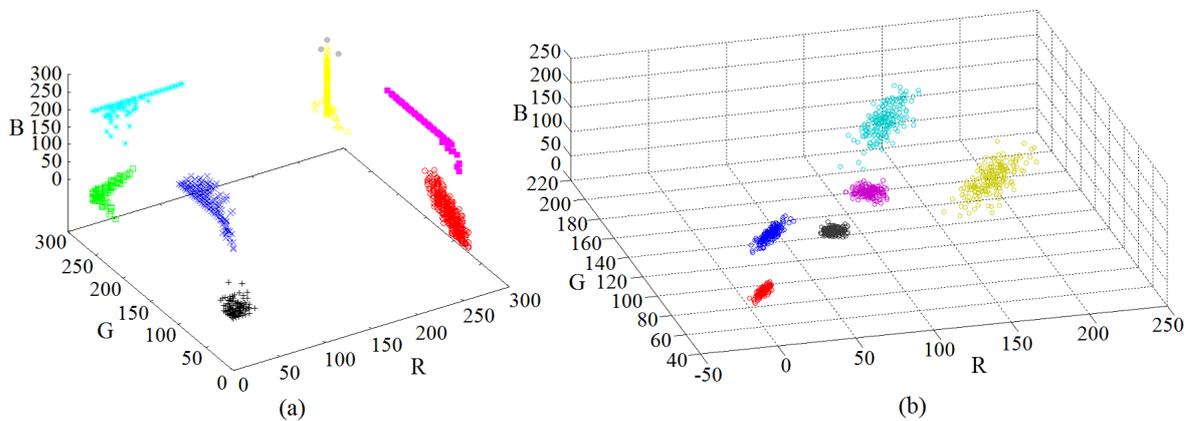


Figura 3.12 – Diagramas RGB para 8 (a) e 6 (b) cores em testes realizados por (MEMETI, 2012) e (LIU; DOERMANN; LI, 2008), respectivamente.

A Figura 3.12 aponta que uma cor pode variar na imagem capturada pela câmera, mas a distribuição é geralmente concentrada em torno da média de cor com alguma covariância (LIU; DOERMANN; LI, 2008). Os autores avançam e sugerem a utilização de um ruído Gaussiano 3D para a modelagem da degradação de cor.

<sup>24</sup> A cor branca é impressa em cinza para ser visível.

O que torna dificultosa a modelagem do canal reside no fato de que o canal de câmera não possui um parâmetro de distorção fixo (LIU; DOERMANN; LI, 2008). Alguns sensores CMOS, por exemplo, são insensíveis à variação no vermelho, enquanto que em outros o mesmo ocorre para o canal verde. Além da sensibilidade dos sensores, ocorrem variações entre diferentes câmeras. Para ilustrar, os autores apresentam uma tabela comparativa das variações médias dos valores dos canais R, G e B para cinco cores em oito câmeras diferentes.

Em virtude das variações cromáticas apresentadas na Figura 3.12 (a), a determinação da cor é o grande obstáculo na leitura dos módulos do barcode. Em geral, para a sincronização do barcode, utilizando os módulos de localização, a redução das  $16,8^{25}$  milhões das possíveis cores para apenas duas é suficiente. Todavia, para a decodificação dos bits do barcode é necessário a dedução (e distinção) de oito<sup>26</sup> cores dentre as 16,8 milhões possíveis (MEMETI, 2012). A questão é como estabelecer um *threshold* para os diferentes canais de cores que sejam capazes de distinguir os agrupamentos de cores.

Para isso, a utilização de um *threshold* sugere duas abordagens. A primeira utiliza o *threshold* fixo para cada canal de cor. Comumente, o valor estabelecido é 128, por estar na faixa central da escala de cinza. O valor médio de cada um dos módulos (com  $n$  pixels) pertencentes às respectivas matrizes dos canais RGB é determinado por:

$$R_{avg}, G_{avg}, B_{avg} = \frac{\sum_{i=1}^n R_i, G_i, B_i}{n} \quad (3.7)$$

Se o valor médio do módulo for inferior a 128, ele será atribuído para “0”, caso contrário, para “1”. A partir da determinação dos *thresholds* de cada módulo, a cor resultante da adição dos três canais estará determinada. Por consequência, o barcode colorido será determinado pela adição das matrizes dos canais R, G e B.

Apesar de simples, essa abordagem com *threshold* fixo não traz bons resultados, especialmente para imagens claras, ou escuras. Ocorre que as áreas pretas não apresentam canal de cor igual a zero, nem as áreas brancas possuem cor igual a 255. Desta maneira, o pixel não usa completamente a faixa de 0 a 255. Assim, as cores preta e branca capturadas em cada canal ( $B_{canal}$  e  $W_{canal}$ ) precisam ser convertidas para valores os absolutos 0 e 255, respectivamente, conforme ilustra o exemplo da Figura 3.13.

Torna-se claro, a partir da Figura 3.13, que quanto menor a distância entre os valores

<sup>25</sup> Cada pixel contém um valor entre 0 e 255 para os canais vermelho, verde e azul (espaço de cores RGB 24 bits). Isto resulta em 16.777.216 diferentes cores para cada pixel.

<sup>26</sup> Devido à variação tonal das cores, o número de cores é tipicamente limitada em menos de 8 cores.

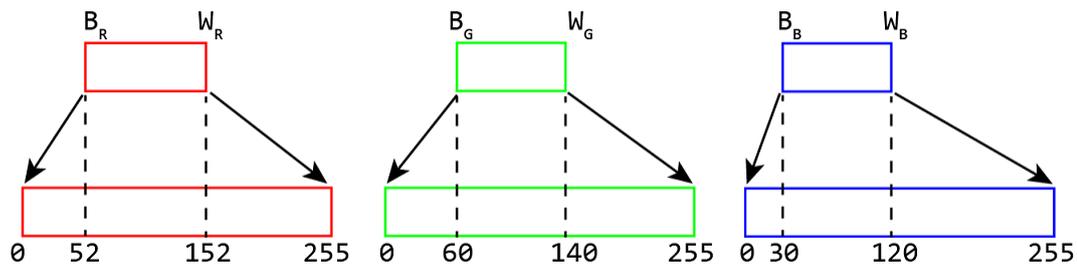


Figura 3.13 – Exemplo de mapeamento dos valores RGB preto e branco relativos para os valores absolutos (extraído de (MEMETI, 2012)).

$B_{canal}$  e  $W_{canal}$  mais difícil será a separação monocromática em cada canal. Também é possível observar a limitação imposta pela utilização do limiar fixo em 128. No canal azul, por exemplo, é impossível fazer a distinção entre brancos e pretos.

Combinadas com outras distorções<sup>27</sup>, as alterações cromáticas apresentam vários desafios (BULAN; SHARMA, 2011), a citar:

- A técnica para codificação em cada canal individual precisa ser robusta a variações tonais nos valores em escala de cinza;
- As distorções geométricas introduzidas no processo de digitalização precisam ser compensadas para permitir a sincronização entre o barcode e o aplicativo de detecção;
- As características de interferência de cores entre os canais precisam ser compreendidas na etapa de codificação, com o intuito de minimizar o efeito da interferência de cor no detector;
- No receptor, o modelo de canal *print-scan*, ou outro, utilizado para estimar os dados codificados, deveria ser capaz de capturar as características de interferência *cross-channel* para extrair completamente as informações dos sensores RGB da imagem. Isto é feito com uso de cores de referência, conforme veremos; e, por fim,
- Um código de correção de erro é necessário para reparar eventuais erros de estimação devido às distorções do canal, incluindo a interferência entre os canais de cor.

Quanto mais íntimo um código barcode for das características acima listadas mais legível (facilmente sincronizável), robusto (imune a distorções) e confiável (maior a garantia de encontrar no receptor os dados codificados) ele será.

<sup>27</sup> Distorções geométricas e ruído que são geralmente tratadas de maneira independente na literatura.

### 3.2.4 Calibração das Cores

No contraponto à robustez assegurada pelos barcodes monocromáticos está, como vimos, a baixa densidade de dados para uma mesma área. Na maioria dos barcodes monocromáticos existentes a utilização das duas cores espectralmente mais distanciadas assegura a robustez na leitura, mesmo para leitores com relativo baixo custo e resolução. Quando comparado com os barcodes monocromáticos, os barcodes coloridos são mais susceptíveis a variações introduzidas por uma impressora em particular, à tecnologia de impressão utilizada (isto é, jato de tinta, laser colorida, ou sublimação) e o substrato de impressão, tal como o papel e seu envelhecimento (KATO; TAN; CHAI, 2010). Por este motivo, o valor da cor impressa pode desviar do valor original que foi enviado em formato digital para o dispositivo de impressão. Essa distorção também pode ser oriunda da iluminação<sup>28</sup> não uniforme ao longo do barcode (ONG; CHAI; TAN, 2009).

A Figura 3.14 apresenta duas fitas com maior cromaticidade e brilho em AB do que em CD. No entanto, quando vistas como uma imagem, A e B são superfícies diferentes, enquanto que A e D emitem luz de mesma matiz, saturação e intensidade.

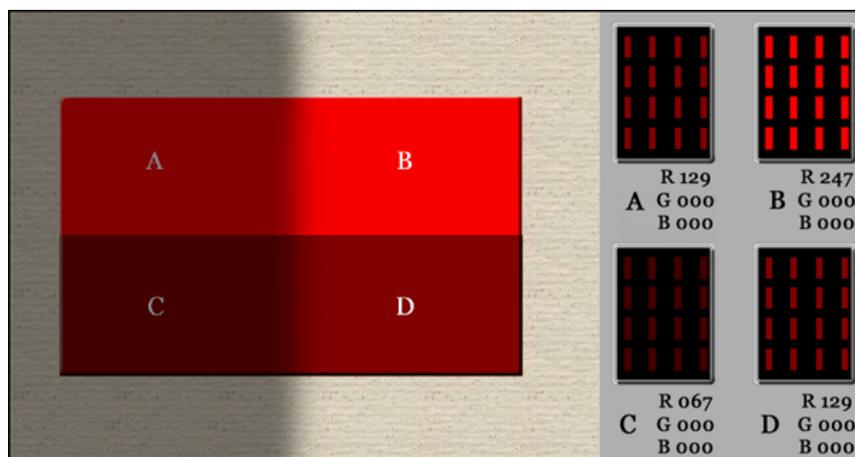


Figura 3.14 – Efeito da iluminação na alteração das cores (extraído de (BRIGS, 2012)).

Em virtude dos efeitos danosos da iluminação desigual<sup>29</sup> e da dependência do *hardware* de impressão e digitalização envolvido, uma representação precisa da cor é usualmente desafiadora. (KATO; TAN; CHAI, 2009) acrescentam, ainda, a estes dois efeitos, a conversão entre

<sup>28</sup> A iluminação desigual pode ser observada pela extração de histogramas em diferentes regiões da imagem.

<sup>29</sup> Parte de um símbolo sobre o efeito de sombreamento também causa falhas na decodificação (KATO; TAN; CHAI, 2010).

espaços de cores e o formato do arquivo em que as imagens capturadas são salvas. A utilização de *thresholding* adaptativo, em detrimento a algoritmos de *thresholding* global, apresentam resultados melhores, uma vez que um valor único para o limiar geralmente não produz bons resultados, quando o brilho (luz do fundo) da imagem varia ao longo do barcode.

No padrão HCCB, por exemplo, as variações cromáticas são superadas pela incorporação de uma paleta de referência dentro do símbolo. Isto permite a comparação entre o valor escaneado, ou amostrado e o conjunto de cores de referência, de modo que a correção de cor possa ser realizada (KATO; TAN; CHAI, 2010). Em outras palavras, as cores de referência atualizam o modelo de cores baseado no ambiente atual sobre o qual ele trabalha (LIU; DOERMANN; LI, 2008). Outras nomenclaturas são utilizadas para se referir às cores de referência, são elas: calibração de cor (LIU; DOERMANN; LI, 2008), blocos piloto (BLASINSKI; BULAN; SHARMA, 2013) e paleta de cores (QUERINI; ITALIANO, 2013).

Durante a codificação os valores RGB (ou outro espaço de cor) dos símbolos são definidos e codificados. Contudo, as distorções oriundas do canal da imagem corrompem o símbolo, deslocando-o do seu valor original. Na Figura 3.15, por exemplo, a localização da célula decodificada é  $RGB = (148, 155, 165)$  que está muito próxima da cor de referência  $(143, 145, 161)$ , porém muito aquém do valor codificado  $(255, 255, 255)$ . A comparação entre os símbolos decodificados e de referência fornece uma “média” para as cores se “auto” calibrarem em relação às condições de iluminação e ao movimento das cores devido aos meios de impressão e captura (ONG; CHAI; RASSAU, 2011). Este cenário é propício para a melhoria da robustez do barcode.

Mesmo com a utilização das cores de referência, a iluminação desigual ao longo do barcode é indesejada, uma vez que a iluminação em uma parte do barcode pode não ser a mesma que em outra parte. Neste cenário, uma solução bastante atrativa apropria-se da diversidade espacial para distribuir réplicas das cores de referência em diferentes regiões do barcode, atenuando o efeito da iluminação desigual. Além disso, não se pode criar a falsa expectativa de que as cores de referência serão capazes de salvaguardar todos os problemas de variações cromáticas na leitura do barcode. O antídoto final para problemas não resolvidos utilizando a calibração de cor está no código de correção de erro, intrinsecamente incorporado aos barcodes.

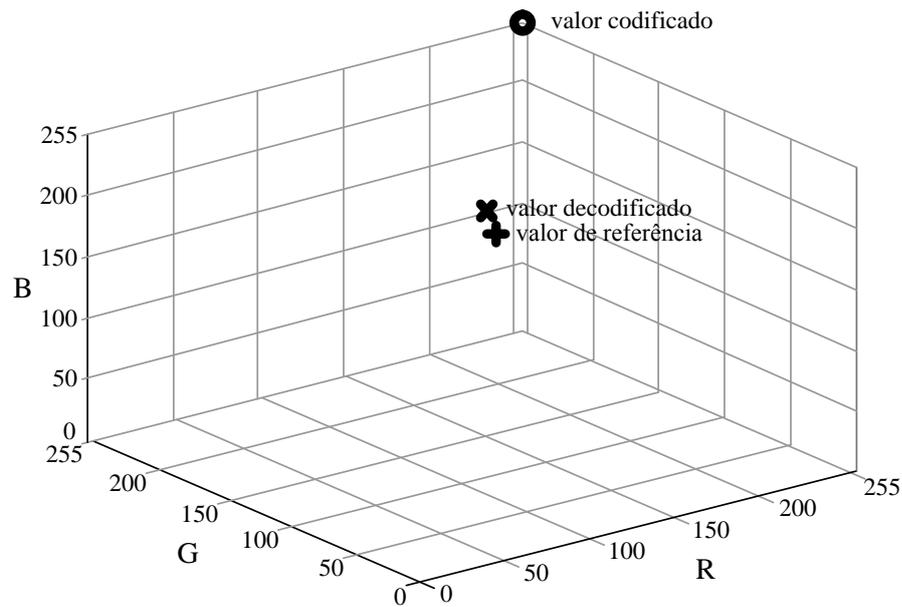


Figura 3.15 – Localização dos símbolos codificados, decodificados e da cor de referência (adaptado de (ONG; CHAI; RASSAU, 2011)).

#### 3.2.4.1 Métricas para Análise das Variações Cromáticas

Em (ONG; CHAI; TAN, 2009), o desempenho das câmeras de celulares é avaliado baseado em três indicadores:

1. Movimento da cor de teste: mede o quanto a cor capturada se moveu em relação a cor original codificada.

Para fins de comparação entre duas ou mais câmeras, a distância euclidiana ( $D$ ) entre as localizações original  $(R_1, G_1, B_1)$  e a capturada  $(R_2, G_2, B_2)$  da cor no espaço de cores RGB é determinada por

$$D = \sqrt{(R_2 - R_1)^2 + (G_2 - G_1)^2 + (B_2 - B_1)^2}. \quad (3.8)$$

Para um melhor desempenho, a distância  $D$  deve ser mínima. A Figura 3.16 exemplifica o posicionamento do símbolo RGB = (0,255,255) em 20 amostras coletadas. Observe que as amostras estão bastante afastadas do símbolo original, havendo uma maior probabilidade de um símbolo ser confundido com outro, o que já mencionamos na Figura 3.12.

2. Agrupamento de teste: mede o quão próximo está cada símbolo de um barcode da média do agrupamento de cor e não do valor original, como no item 1. Para fins ilustrativos, a

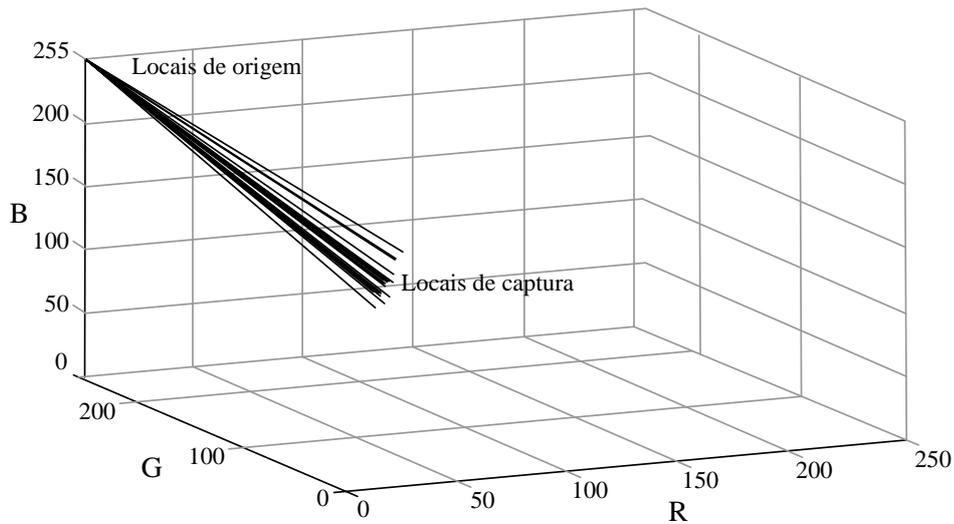


Figura 3.16 – Localização das 20 amostras do símbolo original  $RGB = (0, 255, 255)$  (adaptado de (ONG; CHAI; TAN, 2009)).

Figura 3.17 apresenta os agrupamentos dos símbolos  $RGB = (0, 255, 255)$  e  $RGB = (255, 255, 0)$  de um mesmo barcode. Como pode ser visto, a exemplo do que vimos na Figura 3.12, os símbolos capturados definem uma localização única. Contudo, é importante ressaltar que símbolos mais coesos uns com os outros repercutem em resultados melhores na decodificação. Essa presunção mitiga a possibilidade do decodificador confundir um símbolo com outro próximo.

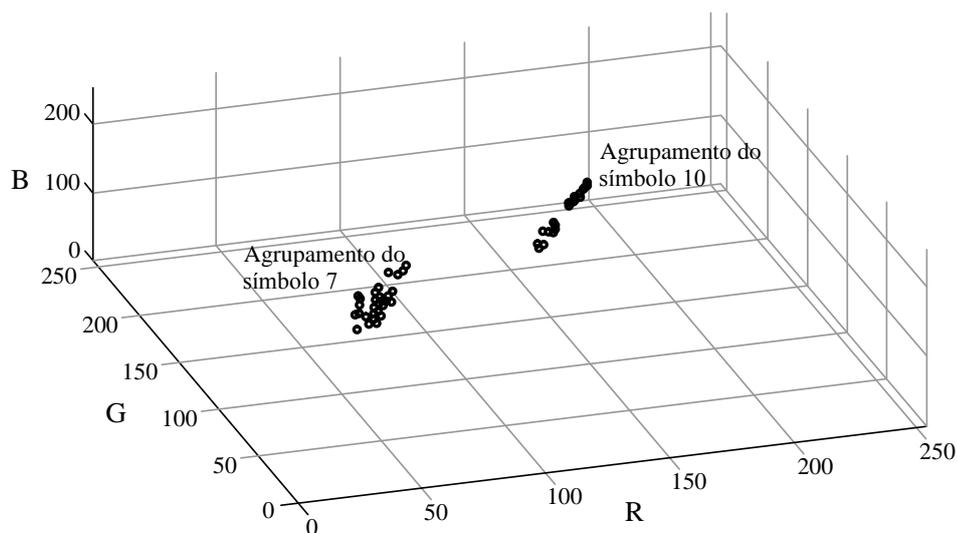


Figura 3.17 – Agrupamento dos símbolos  $RGB = (0, 255, 255)$  e  $RGB = (255, 255, 0)$  de um barcode (adaptado de (ONG; CHAI; TAN, 2009)).

Para avaliar o quão coesos (próximos) estão os símbolos é necessário estabelecer o ponto referência  $(R_{ref}, G_{ref}, B_{ref})$  que é obtido pela média de todos os  $N$  símbolos capturados  $(R_i, G_i, B_i)$  e que representam uma determinada cor. Matematicamente:

$$\{R_{ref}, G_{ref}, B_{ref}\} = \left\{ \frac{\sum_{i=1}^N R_i}{N}, \frac{\sum_{i=1}^N G_i}{N}, \frac{\sum_{i=1}^N B_i}{N} \right\}. \quad (3.9)$$

Em seguida, a distância Euclidiana entre cada símbolo capturado e o ponto de referência (média) é calculado através da Equação 3.8, resultando na medida  $x_i$ . A variabilidade em torno da média das distâncias Euclidianas,  $\bar{x}$ , de um agrupamento pertencente a uma mesma cor é determinada por:

$$\sigma^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2. \quad (3.10)$$

3. Mínima distância Euclidiana entre símbolos de teste: indica qual dos grupamentos de cor possuem os símbolos mais afastados entre si. É determinada pela distância Euclidiana (Equação 3.8) entre os pontos de referência (Equação 3.9) e as cores que compõem o barcode. Para se maximizar o desempenho da decodificação, busca-se pela máxima mínima distância Euclidiana.

A partir dos testes compilados em (ONG; CHAI; TAN, 2009), os autores concluem que as novas tecnologias de câmeras celulares, com maior capacidade em megapixels, não necessariamente traduzem melhoria na distinção de cores nos barcodes 2D. Isso muito se deve ao fato destas câmeras serem projetadas para cenas e não para captura de imagens próximas.

## 4 TÉCNICAS PARA EXPANSÃO DO ARMAZENAMENTO DE DADOS

Os códigos 2D exploram duas dimensões espaciais (vertical e horizontal) para realizar o armazenamento de dados (GRILLO et al., 2010). O desafio para esta etapa do trabalho é justamente potencializar ainda mais a capacidade de informação (*data payload*) em relação aos códigos 2D monocromáticos<sup>30</sup>. Embora, por exemplo, o QR Code seja constituído em diferentes versões e, portanto, com quantidades de dados variadas, seu tamanho é limitado à versão dez<sup>31</sup> para decodificadores embarcados em telefones *mobiles* (KIKUCHI; FUJIYOSHI; KIYA, 2013). Estes autores destacam, ainda, duas possibilidades de incremento da capacidade de informação do QR Code: uma delas utiliza técnicas de ocultação de dados (*data hiding techniques*) e a outra é baseada em intensidades multiníveis (*multilevel intensities*).

A primeira esconde a informação extra na imagem preto e branco (WANG, 2009), (CHUNG; CHEN; TU, 2009), (BULAN; SHARMA; MONGA, 2010), (HUANG; CHANG; FANG, 2011), (RUNGRAUNGSILP; KETCHAM, 2012). Embora as técnicas de ocultação distorçam ligeiramente a imagem para esconder dados extras no código, esta distorção não interfere na obtenção dos dados realizada pelo decodificador padrão. Por conseguinte, um decodificador proprietário poderá extrair a informação oculta na imagem. Para aumentar a quantidade extra de dados, essas técnicas necessitam de uma resolução maior da imagem.

A segunda, por sua vez, explora a diversidade espectral (RAMYA; JAYASHEELA, 2014) de cores, baseada em intensidades multiníveis que diretamente codificam a informação. O crescimento do número de cores permite alcançar uma alta capacidade teórica de dados, superior aos barcodes 2D monocromáticos (ONG; CHAI; TAN, 2009). Os QR Codes multiníveis são, ainda, subdivididos em duas categorias: escala de cinza (*grayscale*) e colorido (*color*). A utilização de tons de cinza permite o aumento da quantidade de dados em apenas duas vezes (KIKUCHI; FUJIYOSHI; KIYA, 2013), devido à degradação da tolerância ao ruído e às transformações geométricas. No entanto, mantém a compatibilidade com o QR padrão. Por outro lado, a codificação em cores supera a taxa de dados dos barcodes *grayscale*, sob o custo de um *codec* mais complexo.

Além dessas duas possibilidades aventadas na literatura, este trabalho elege outras duas importantes técnicas que permitem aumentar a capacidade de dados dos barcodes 2D, con-

<sup>30</sup> Isto é, de tinta única.

<sup>31</sup> Cabe salientar que a referência utilizada é de 2013 e que, portanto, acredita-se ter ocorrido uma significativa evolução nas tecnologias *mobiles* desde então.

forme pode ser visto na Figura 4.1. A utilização de técnicas “geométricas” permite expandir a capacidade de dados através da adoção de formas geométricas mais densas, tanto a nível de células quanto de símbolos. Já as técnicas de “animação”, que são mais detalhadas a seguir, multiplexam símbolos ao longo do tempo, possibilitando a transmissão de um quantidade, teoricamente, ilimitada de dados. Por óbvio, esta última técnica necessita de uma tela capaz de reproduzir sequencialmente os barcodes.

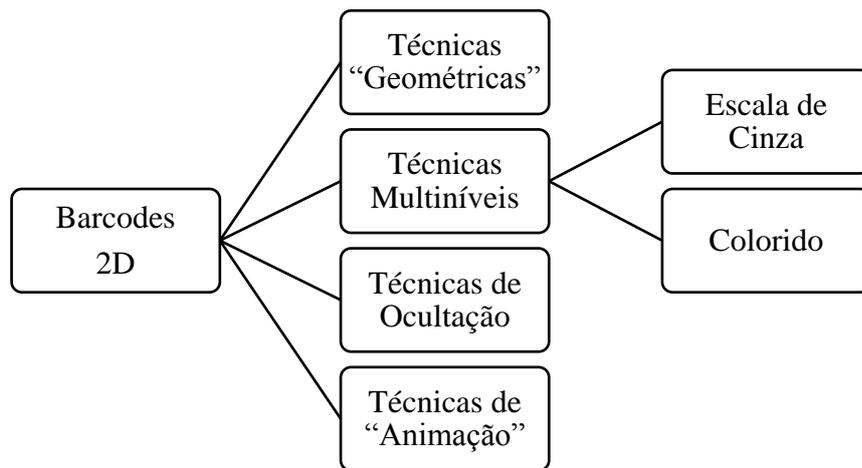


Figura 4.1 – Técnicas para incremento da capacidade de dados dos códigos de barras 2D.

Para este capítulo, em sua primeira seção, os barcodes coloridos serão tratados como uma possibilidade de aumentar a capacidade de representação de dados em uma mesma área. Outra possibilidade é reduzir o espaço codificável para carregar uma mesma informação, ou ainda, aumentar a redundância codificada e aumentar a confiabilidade. Além disso, uma suave abordagem dos barcodes animados é realizada, com o intuito de apresentar uma importante aplicação de transferência de dados por imagens. Na segunda e última seção do capítulo, os barcodes coloridos, foco especial desta dissertação, são analisados com o intuito de se averiguar quais características estes barcodes devem possuir e quais são os principais desafios a serem enfrentados.

#### 4.1 Codificação em Cores - Barcode Colorido

Como vimos, mesmo os barcodes *database*, ou orientados ao conteúdo, possuem capacidade restrita a uma “pegada”, normalmente com limitante de algumas centenas ou, em certas situações, de alguns milhares de bytes. A fim de romper com essa limitação duas abordagens

têm sido alvo de pesquisas para expandir a capacidade de informação dos barcodes 2D (LIU; DOERMANN; LI, 2008):

1. O uso de cores permite que cada pixel carregue mais informação;
2. A utilização de barcodes 2D *multi-frame* ou multiplexados no tempo.

Na mesma linha, em (LANGLOTZ; BIMBER, 2007) os autores vislumbram o aumento da resolução do barcode 2D como outra possibilidade para aumentar o volume de dados embarcados. Entretanto, concluem que a ótica utilizada nas câmeras digitais dos chamados celulares *off-the-shelf*<sup>32</sup> possuem claras limitações.

Uma vez que as câmeras de telefones celulares capturam, inerentemente, imagens coloridas, em vez de monocromáticas, a maneira mais natural de aumentar a capacidade dos barcodes 2D para aplicações móveis é, justamente, explorar a “diversidade do espectro” que os sensores vermelho, verde e azul oferecem sob o ponto de vista de comunicação (BULAN; BLASINSKI; SHARMA, 2011). Além disso, (KATO; TAN; CHAI, 2010) acrescenta que, além do aumento da capacidade de dados, o uso de imagens coloridas são visualmente mais “apela-tivas” e melhor apreciadas pelo público em geral, especialmente em campanhas publicitárias. O primeiro barcode colorido proposto, o ColorCode atentou para este fato, com a utilização de *eyecatching symbols*, ou símbolos atraentes aos olhos.

Todavia, aquém do aumento da capacidade de dados e do maior apelo visual, dois aspectos limitantes devem ser considerados: o primeiro está relacionado ao custo computacional mais elevado no trato de símbolos coloridos. Já o segundo considera crítica a reprodução<sup>33</sup> das cores originais a partir de um barcode capturado. Por estes motivos, muitos pesquisadores preferem trabalhar com barcodes preto e branco.

De um modo geral, a operação sobre cores pode ser tratada como uma terceira dimensão inserida aos barcodes que, por consequência, passam a ser chamados de barcodes 3D - *three dimensional barcode* (NURWONO; KOSALA, 2009), (DEAN; DUNN, 2012), (BLASINSKI; BULAN; SHARMA, 2013). No entanto, a terminologia “barcode 3D” é utilizada para nomear os barcodes em auto relevo (*bumpy barcode*). Um *bumpy barcode* é um barcode 1D, ou 2D, que é diretamente gravado por altos e baixos na superfície do material para ser lido sem o uso de contraste visual entre diferentes cores (COMMUNICATIONS, 2014). Diante desse impasse

<sup>32</sup> Produtos que estão comercialmente disponíveis para uso geral.

<sup>33</sup> Leia-se, para nosso interesse, a extração de bits. Diz-se que a reprodução das cores, nesse contexto, é dependente do dispositivo e de seus parâmetros.

optamos por suprimir a terminologia “barcode 3D” e passamos a utilizar outras expressões encontradas na literatura. Muitos autores preferem chamar de “Barcode Colorido”, “Barcode Multinível”, “Barcode 2D Multicolorido”, “Código de Barras Colorido”, ou ainda, “Barcode 2D Colorido”. Neste trabalho, essas terminologias serão empregadas intercambiavelmente para fins de enriquecimento textual.

Na literatura o interesse por barcodes 2D tem se acentuado ao longo dos anos em consonância com a proliferação do uso de *smartphones*. Mesmo assim, até o ano de 2009, em (KATO; TAN; CHAI, 2009) foram apurados apenas três barcodes 2D coloridos, quais sejam: *ColorCode* (TACK-DON et al., 2006), *High Capacity Color Barcode* (HCCB) e *Paper Memory* (PM) (ONODA; MIWA, 2009). Por sua vez, este trabalho identificou, além dos já referidos barcodes coloridos, os seguintes (em ordem cronológica): *Data Storage Device and Encoding/Decoding Method* (TAN et al., 2008), *High Capacity Colored Two Dimensional Codes* (GRILLO et al., 2010), *CQR Codes: Colored Quick-Response Codes* (Vizcarra Melgar et al., 2012) e *A New Color QR Code Forward Compatible with the Standard QR Code Decoder* (KIKUCHI; FUJIYOSHI; KIYA, 2013).

O que estes códigos têm em comum está no esquema de codificação da cor, o que torna possível aumentar a capacidade de dados de um barcode 2D, desde que *mais do que um bit* possa ser codificado em um única célula de dados (KATO; TAN; CHAI, 2010). Teoricamente, usando  $n$  cores é possível aumentar a capacidade de pixel para  $\log_2(n)$  bits (GRILLO et al., 2010). Sob esta mesma ótica, sistemas de impressão e captura de imagem que tipicamente utilizam a representação RGB em 24 bits podem, em teoria, extrair 24 bits de informação de um único pixel (LIU; DOERMANN; LI, 2008).

Embora os códigos 2D possuam maior capacidade<sup>34</sup>, inovações que permitam aumentar ainda mais sua densidade de dados possuem considerável interesse de investigação, pois possibilitam (BULAN; BLASINSKI; SHARMA, 2011):

1. A redução<sup>35</sup> da área de impressão utilizada;
2. Adicionar informação com a mesma área de impressão;
3. O aumento da robustez; ou

<sup>34</sup> Em contrapartida aos códigos de barras 1D.

<sup>35</sup> Uma área de impressão reduzida desafia a separabilidade das células, especialmente para câmeras que não possuem foco automático. Já em barcodes maiores a representação da cor ao longo do símbolo pode variar, uma vez que a iluminação em uma parte do barcode pode não ser a mesma da outra parte (ONG; CHAI; RASSAU, 2010).

#### 4. Alguma combinação destas.

##### 4.1.1 Codificação de Dados por Matrizes de Pontos Modulados

Em (BULAN; SHARMA, 2011) propõe-se um barcode colorido de alta capacidade utilizando o espaço de cores CMY e o canal *print-scan*. Cada canal monocromático da impressora carrega dados independentemente, aumentando a capacidade sobre realizações em preto e branco. Os canais C, M e Y transmitem dados através de um arranjo periódico de pontos em forma elíptica, com modulação em sua orientação para codificar dados, conforme mostra a Figura 4.2 (a). Esta técnica é baseada no método de ocultação proposto em (BULAN; SHARMA; MONGA, 2010). Na imagem escaneada, apresentada na Figura 4.2 (b), as características espectrais complementares dos corantes ciano, magenta e amarelo da impressora e os canais vermelho, verde e azul do dispositivo de escaneamento são exploradas no receptor.

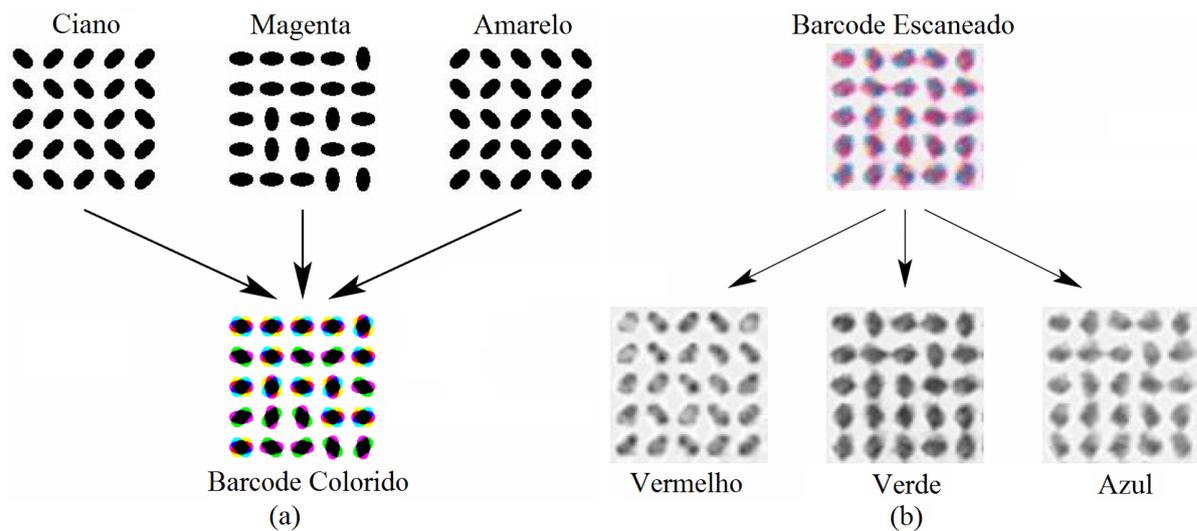


Figura 4.2 – (a) Método de Codificação por modulação de pontos elípticos e o barcode resultante. (b) Imagem escaneada com as distorções inerentes ao canal *print-scan* e a decomposição nas componentes R, G e B (adaptado de (BULAN; SHARMA, 2011)).

Desde que as variações nas operações de impressão e escaneamento não comprometam a orientação dos pontos elípticos, esta técnica é inevitavelmente robusta contra típicas alterações tonais da impressora e do *scanner*. Além disso, os autores destacam que os módulos em formato de pontos são menos sensíveis a variações tonais na resposta da impressora.

#### 4.1.2 Barcodes 4D

Aquém das técnicas *hiding techniques* e *multilevel intensities* utilizadas para aumentar a capacidade de dados, conservando-se intactas as dimensões do barcode 2D, pode-se considerar o uso de dimensões adicionais, tais como o tempo e as propriedades geométricas do barcode (KATO; TAN; CHAI, 2010). Em virtude da utilização da dimensão tempo, na qual é concebida a possibilidade teórica de codificar uma quantidade ilimitada de dados (DOERMANN, 2008), alguns pesquisadores se empenham para combinar à grandeza tempo uma geometria eficiente capaz de maximizar a capacidade de dados por unidade de área. Aditadas a estas duas possibilidades, o aprimoramento técnico e o avanço tecnológico dos mecanismos de captura e processamento de imagem, traça-se um cenário bastante promissor para o incremento da taxa de dados.

Sem a pretensão de nos aprofundarmos no trato de imagens animadas, faremos uma breve explanação baseada na Figura 4.3, na qual é apresentada simplificada o conceito de transferência de dados por imagem. Alguns autores preferem chamar os barcodes animados de barcodes 4D, no qual além das três dimensões já conhecidas, acrescenta-se a quarta dimensão, o tempo (LANGLOTZ; BIMBER, 2007). No entanto, alguns trabalhos com barcodes animados foram desenvolvidos suprimindo-se a dimensão cor, a citar (DOERMANN, 2008).

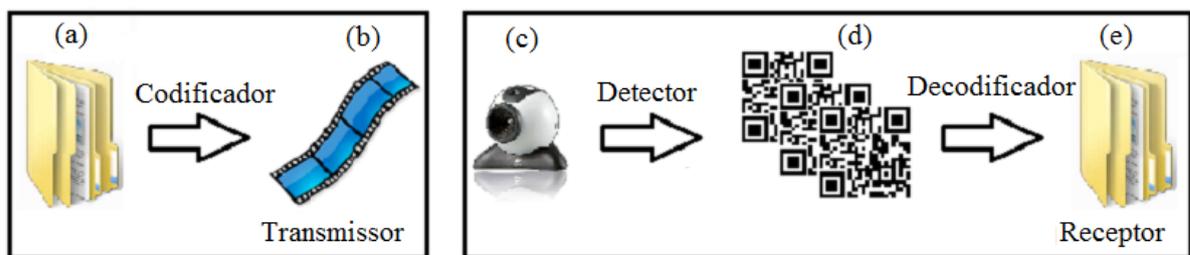


Figura 4.3 – Codificação e decodificação de imagens animadas (adaptado de (MEMETI, 2012)).

O transmissor da Figura 4.3 precisa ser capaz de converter os dados (a) em uma sequência codificada de barcodes 2D gerados (b). O receptor, por sua vez, precisa ser equipado com uma câmera digital (c) e um *software* para decodificar a animação. No receptor as sequência de barcodes 2D precisam ser detectadas (d) pela câmera para depois ter seu conteúdo acessado e, se necessário, recuperado no decodificador (e).

A comunicação baseada em câmera é uma classe da chamada Comunicação por Luz Visível (*Visible Light Communication - VLC*), em que a informação é modulada através da luz,

transmitida a partir de emissores ópticos<sup>36</sup> e recebida por elementos fotossensores no receptor (ASHOK et al., 2014). O canal de comunicação tela-câmera (*screen-camera channel*) possui uma análise análoga ao canal RF MIMO (*radio frequency multiple-input multiple-output*), no qual cada pixel do sensor da câmera é considerado uma antena receptora, ao passo que cada elemento emissor de luz é considerado uma antena transmissora. A comunicação baseada em câmera é caracterizada pela transmissão altamente direcional entre transmissor e receptor, com perdas de multipercurso desprezíveis. Em contrapartida, este canal sofre amplamente com as distorções perspectivas.

Comparado com outros canais de dados, tais como CDMA/GPRS, cabos, *Bluetooth*, *WiFi* e infravermelho, a comunicação visual não necessita de *hardware* adicional, ou plano de dados (DOERMANN, 2008). Este cenário de câmera onipresente, ou de câmera pervasiva, permite sua utilização não apenas para capturar imagens, mas também para comunicação de informação (ASHOK et al., 2014).

## 4.2 Barcodes 2D Coloridos Existentes na Literatura

Na seção inicial deste capítulo foram listados os barcodes coloridos analisados por este trabalho. Nesta seção, suas características são melhor esclarecidas, sem muita preocupação com a reprodução dos resultados colhidos por seus autores. Preocupa-nos, sim, as características que tornam esses barcodes robustos e com maior capacidade de dados.

### 4.2.1 ColorCode

O *ColorCode* foi proposto em 2000 por Han e seu time na Universidade de Yonsei na Coreia do Sul. Esse foi o primeiro barcode 2D a utilizar cor para codificar dados em aplicações celulares<sup>37</sup> (KATO; TAN; CHAI, 2010). Tornou-se, pois, inspiração para futuros trabalhos na área de “etiquetas” visuais, mais conhecidas como *visual tags*.

O *ColorCode* é, talvez, o barcode visualmente mais atraente proposto até o presente momento, sendo flexível no desenho e na forma (veja a Figura 4.4). Sendo assim, sua preocupação não é definitivamente o aumento da capacidade de dados. Neste cenário, embora se conheça o potencial de codificação dos barcodes coloridos, o *ColorCode* é um barcode indexado

<sup>36</sup> Por exemplo, LEDs e LCDs.

<sup>37</sup> O *ColorCode* não foi o primeiro barcode colorido a codificar dados, mas foi pioneiro no desenvolvimento de aplicações com câmeras baratas CCD ou CMOS e de baixa resolução, tais como *webcams* e câmeras integradas a celulares (KATO; TAN, 2007).

(direcionado a um banco de dados) que codifica apenas 10 dígitos. Em virtude disso, o símbolo do ColorCode possui células bastante largas, de modo que o decodificador necessita que apenas 40% de cada célula seja ocupado por dados. Esta característica permite que o ColorCode possa ser incorporado em *designs* gráficos nos 60% de espaço restante. Desta maneira, o ColorCode pode realizar duas tarefas simultaneamente (KATO; TAN; CHAI, 2010): como etiqueta visual e propaganda.



Figura 4.4 – *Designs* do ColorCode (extraído de (KATO; TAN; CHAI, 2010)).

#### 4.2.1.1 Estrutura do Símbolo

O padrão *ColorCode* possui  $5 \times 5$  células em uma combinação de quatro cores, quais sejam: vermelho, verde, azul e preto. É constituído de pelo menos duas áreas: dados e paridade. Além dessas duas áreas, o símbolo pode conter as áreas de controle e referência. A distribuição dessas áreas no símbolo pode ser vista na Figura 4.5.

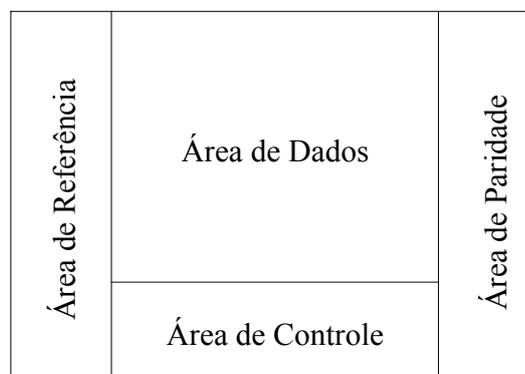


Figura 4.5 – Estrutura do símbolo do ColorCode (adaptado de (TACK-DON et al., 2006)).

O ColorCode utiliza cores de referência para superar problemas relacionados às variações tonais. Deste modo, a cor de cada célula é determinada em relação ao valor da cor de referência (KATO; TAN; CHAI, 2009), situada à esquerda da Figura 4.5. As células de controle guardam a informação da localização e propriedades das células de referência e a sequência para decodificação da área de dados (KATO; TAN; CHAI, 2010).

A área de paridade possui verificadores/corretores de erro para localizar qualquer reconhecimento incorreto da cor. O resultado da operação XOR em cada linha e coluna deve ser o valor da célula de paridade da respectiva linha, ou coluna.

#### 4.2.2 HCCB

O *High Capacity Color Barcode* (HCCB) foi desenvolvido pela *Microsoft* para aumentar a capacidade de dados utilizando símbolos coloridos em geometria triangular. Através da geometria triangular é possível elevar a densidade de dados em até três vezes quando comparado ao barcode DataMatrix (KATO; TAN; CHAI, 2010), conforme pode ser visto na Figura 4.6. Sua principal aplicação está atribuída aos IDs (*identity document*) biométricos, em substituição aos caros RFID, ou *smart cards*.

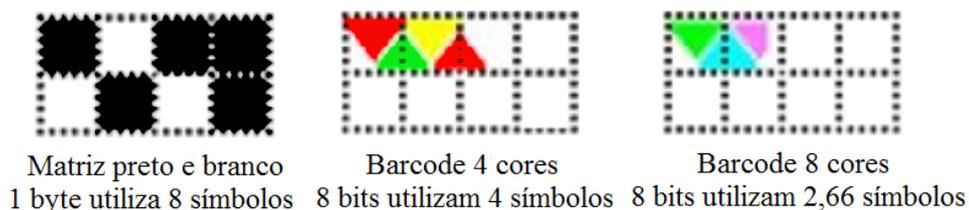


Figura 4.6 – Incremento da capacidade de dados do HCCB (adaptado de (MICROSOFT, 2006)).

A geometria triangular das células permite duas importantes vantagens: aumento da capacidade de dados e amostragem mais precisa da cor. A primeira pode ser facilmente observada através da Figura 4.6, com reduzida ocupação física quando comparado a uma célula quadrada. A segunda, por sua vez, permite uma redução dos efeitos de borramento, ou *anti-aliasing*, já que o triângulo possui apenas três lados susceptíveis à interferência da vizinhança. O HCCB pode embarcar quatro cores (preto, vermelho, verde e amarelo) no símbolo, conforme pode ser visto na Figura 4.7.

O número de células em cada linha é sempre um múltiplo inteiro do número linhas, estas que pode variar (PARIKH; JANCKE, 2008). O símbolo é cercado por uma borda preta ainda

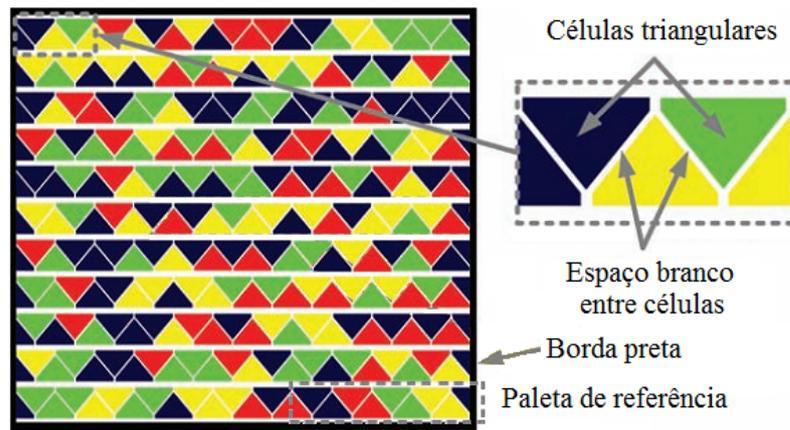


Figura 4.7 – Estrutura do símbolo do HCCB (adaptado de (KATO; TAN; CHAI, 2010)).

mais circundada por uma fina banda branca. Esses padrões atuam como pontos de referência do barcode. A borda preta na parte inferior do símbolo, mais espessa que as outras três, é designada para definir a orientação do símbolo.

A região codificável do símbolo do HCCB é composta pelos dados a serem codificados justapostos com os valores da verificação de redundância cíclica (CRC, *cyclic redundancy check*), com o código *Reed-Solomon* e com a paleta de referência. Estas cores de referência estão localizadas nas últimas oito células da parte inferior direita do símbolo, cuja distribuição ocorre na seguinte ordem: preto, vermelho, verde e amarelo, cada qual composta por duas células. O tamanho do símbolo do HCCB é determinado pela quantidade de dados a ser codificada, incluindo CRC e *Reed-Solomon*, pela paleta de referência e pelo número de cores usadas.

O processo de decodificação<sup>38</sup> inicia-se supondo que um ponto está no interior do barcode. Ela prossegue em quadrados até reconhecer a borda branca, entorno do símbolo. Depois de localizado o barcode o algoritmo busca a borda preta mais espessa a fim de proceder a orientação do símbolo. Em seguida, é realizada a localização dos cantos, das linhas (horizontalmente separadas por linhas brancas) e das células. Após a delimitação das células ocorre a amostragem do centroide de cada célula, que está mais afastada de suas fronteiras e, conseqüentemente, será menos afetada pelas cores vizinhas. Por fim, a atribuição da cor é baseada na paleta de referência e utiliza o algoritmo *mean-shift*. A correção de um eventual erro na atribuição da cor é realizada por códigos CRC e *Reed-Solomon*.

A utilização de mais cores no HCCB é limitada pelo tamanho de cada célula e pela

<sup>38</sup> Maiores detalhes podem ser encontrados em (PARIKH; JANCKE, 2008).

fidedignidade do símbolo reconstruído (KATO; TAN; CHAI, 2010). Quando o tamanho da célula é pequeno torna-se bastante provável que ela seja afetada pelas cores circundantes. A fidedignidade da imagem produzida por dispositivos de baixa resolução é geralmente degradada por borramento, prejudicando a sincronização entre as células (ONG; CHAI; TAN, 2008). Com a intenção de minimizar tal efeito negativo e aumentar a robustez, o HCCB usa espaços brancos de separação entre as células coloridas.

#### 4.2.3 PM Code

O *Paper Memory Code* (PM Code), desenvolvido pela *Content Idea of Asia Co.*, utiliza estrutura de barcodes em camadas. O PM Code é constituído pela sobreposição de camadas de código (*code layers*) que resultam em uma camada superficial (*surface layer*). Cada camada de código é constituída de uma matriz 2D de células coloridas e espaços. A camada superficial é resultante da adição das cores de todas as camadas de código (KATO; TAN; CHAI, 2010), conforme pode ser visto na Figura 4.8. A unificação das camadas de código na camada superficial inclui um código de índice de informação, que contém os valores RGB de cada camada de código.

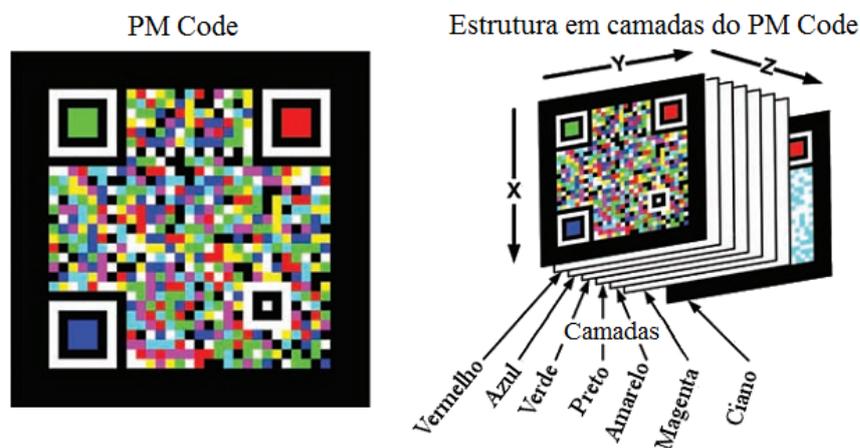


Figura 4.8 – Símbolo do PM Code e sua estrutura em camadas (adaptado de (KATO; TAN; CHAI, 2010)).

A cor em cada célula da camada de superfície será constituída pela própria cor da camada de código, ou pela cor resultante da adição de todas as camadas de código (KATO; TAN; CHAI, 2009). Quando a cor resultante é idêntica à cor usada em uma das camadas, ela deverá ser convertida para uma cor designada de acordo com o algoritmo de conversão de cor do PM

Code, no qual estão envolvidos dos espaços de cores RGB e HSI.

No processo de conversão do PM Code os valores RGB da cor resultante são convertidos para o espaço de cores HSI, de acordo com o algoritmo acima referido. Depois, em operação inversa, os valores HSI são convertidos para RGB. Desta maneira, o algoritmo de conversão de cor, juntamente com o índice de informação na camada de superfície, habilita o *software* de decodificação a detectar a presença, ou ausência de células de cores em cada camada de código, resultando na decodificação total do símbolo PM Code.

Na composição da camada de superfície as camadas de código, cada qual com sua cor no padrão RGB [0, 255], são adicionadas, resultando em valores RGB que podem ultrapassar a faixa permitida para imagens 24 bits. Desta forma, ao invés de saturar o valor no respectivo canal em 255, o algoritmo promove a conversão do espaço de cores RGB para o HSI, afim de diferenciar a cor resultante em cada célula da camada de superfície. Em seguida, os valores HSI são novamente convertidos para RGB.

Por exemplo, na Figura 4.9 (a) e (b), a soma das camadas de código resultam em RGB (255, 255, 0) e (510, 255, 0), respectivamente. Estes valores são convertidos pelo algoritmo do PM Code para o valor HSI (120, 100, 51) e (0, 15, 100), nesta ordem. Observe, a partir da Figura 4.9, que este algoritmo permite a recuperação dos dados no decodificador, entretanto, o *software* de leitura deve ter uma capacidade refinada para reconhecimento de cor, uma vez que precisa distinguir uma ampla faixa de cores (KATO; TAN; CHAI, 2010).

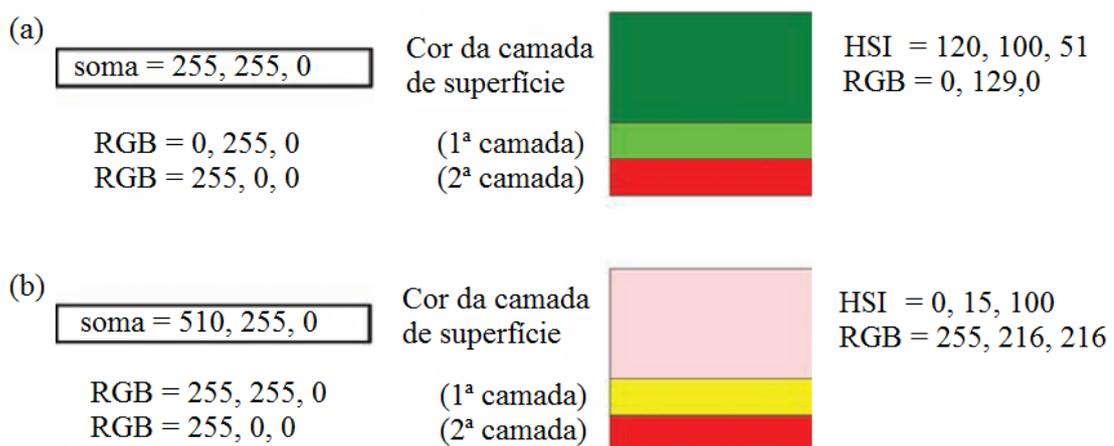


Figura 4.9 – Algoritmo PM Code de conversão de cor (adaptado de (KATO; TAN; CHAI, 2010)).

Tanto o HCCB quanto o PM Code são classificados como *database barcodes*, ao contrário do ColorCode, que trabalha como *index barcode*. Além disso, os dois primeiros foram

desenvolvidos para aumentar a capacidade de dados dentro de um dado espaço, em oposição ao ColorCode, cujo principal objetivo é ser visualmente atraente. Ao que pese o uso de recursos limitados em câmeras de celulares, os códigos HCCB e PM Code permitem a utilização de um número limitado de cores, tais como quatro, ou no máximo oito, pois nenhum deles foi desenvolvido especificamente para dispositivos móveis (KATO; TAN; CHAI, 2009).

#### 4.2.4 MMCC

O *Mobile Multi-Color Composite* (MMCC) exemplificado na Figura 4.10 (a) com  $30 \times 30$  células, consiste em quatro partes principais (ONG; CHAI; RASSAU, 2011): padrão de localização, cores de referência, cabeçalho (barcode 1D) e área de dados (barcode 2D), conforme Figura 4.10 (b). No cabeçalho podem se incluídas informações sobre a versão e formato do símbolo, a posição das cores de referência, o nível de correção de erro e as configurações de segurança. Dependendo da quantidade de informação, pode-se utilizar matrizes 2D para a codificação no cabeçalho.

Na Figura 4.10 (a) o barcode 1D é utilizado para embarcar a informação do cabeçalho. Ao lado do cabeçalho estão dispostas as células das cores de referência que estão circundadas por uma borda branca com quatro pixels de largura, com o intuito de minimizar a interferência das cores vizinhas e da borda preta do padrão de localização. A mesma borda branca é aplicada para promover a separação entre a área de dados e o padrão de localização.

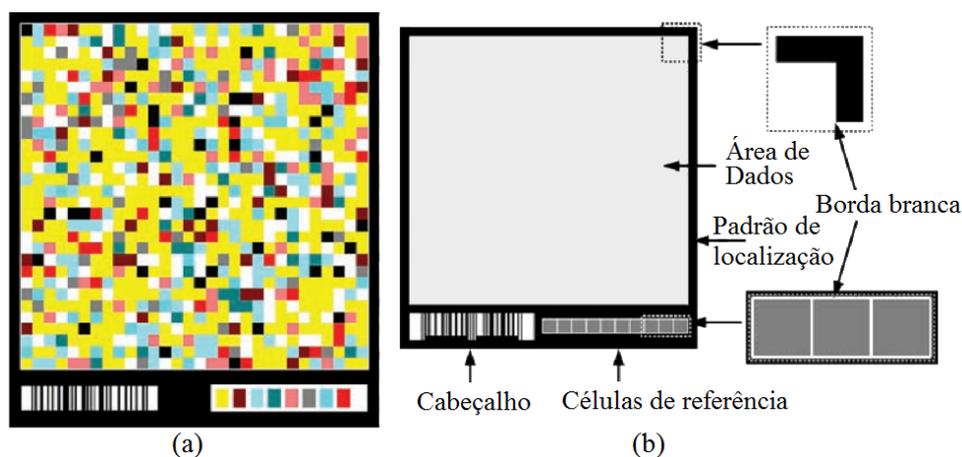


Figura 4.10 – Estrutura do barcode MMCC (adaptado de (KATO; TAN; CHAI, 2010)).

Tanto os símbolos quanto as células do código MMCC foram previstas com o intuito de

aumentar<sup>39</sup> a densidade de dados dentro de um dado espaço. Uma vez que a tela de um celular é usualmente quadrada ou retangular, a forma do MMCC é, também, quadrada, permitindo utilizar o espaço disponível de forma maximizada. Além disso, a célula do MMCC é quadrada, permitindo uma melhor ocupação do espaço disponível no interior do símbolo, já que as células podem ser arranjadas de modo que não seja perdido espaço entre elas, ou no interior do símbolo. Por exemplo, no HCCB há perda nas laterais direita e esquerda do barcode, devido ao formato triangular da célula.

A área de dados inclui a informação a ser codificada e detector/corretor de erro, isto é, o *Reed-Solomon* com capacidade de correção padrão de 22%. Primeiramente, o decodificador extrai a informação do cabeçalho e, em seguida, recupera os valores das cores das células de referência. Como o padrão de localização é o primeiro objeto a ser localizado no barcode pelo *software* de leitura, é assegurada a amostragem das cores de referência, em sua correta localização.

O MMCC permite utilizar até dez cores para codificar e decodificar informação de maneira robusta. Por meio desta característica é possível utilizar caracteres decimais (0 ~ 9), para compor cada célula, conforme detalha a Tabela 4.1. Os valores das cores estão normalizados no intervalo [0, 1].

Tabela 4.1 – Símbolos do MMCC.

Decimal	Cor	Valor RGB
0	Amarelo	(1, 1, 0)
1	Branco	(1, 1, 1)
2	Azul céu	(0,5, 1, 1)
3	Ciano	(0, 1, 1)
4	Cor areia	(1, 0,5, 0,5)
5	Cinza	(0,5, 0,5, 0,5)
6	Cinza escuro	(0, 0,5, 0,5)
7	Vermelho	(1, 0, 0)
8	Marrom	(0,5, 0, 0)
9	Preto	(0, 0, 0)

#### 4.2.5 HCC2D

O *High Capacity Colored Two Dimensional Codes* (HCC2D) foi desenvolvido para aumentar a densidade de dados utilizando cores sobre o QR Code padrão, preservando a forte

<sup>39</sup> O MMCC realiza uma prévia compressão dos dados antes de proceder a codificação, permitindo aumentar a capacidade do código.

robustez e correção de erro do QR Code<sup>40</sup> sendo, portanto, compatível com este. Ademais, o HCC2D alcança capacidade de dados similar ao código HCCB. A Figura 4.11 apresenta amostras do HCC2D com quatro cores (a) e com 16 cores (b).

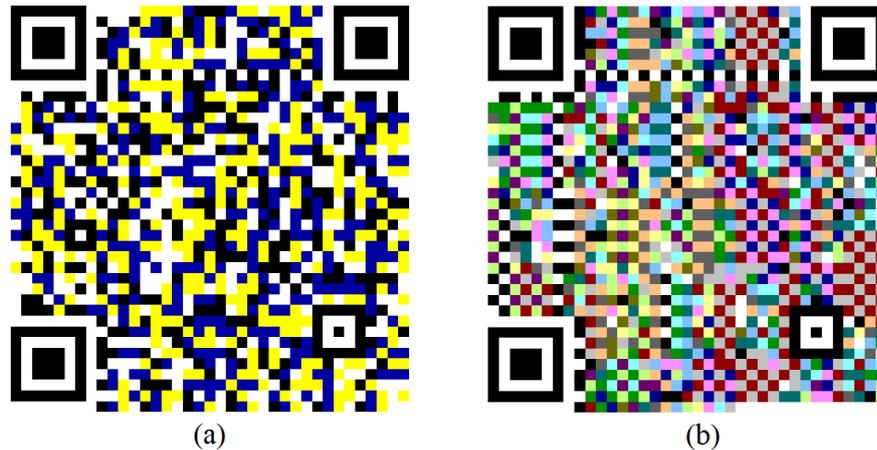


Figura 4.11 – HCC2D versão 5, nível de correção de erro H: (a) 4 cores; (b) 16 cores (extraído de (GRILLO et al., 2010)).

Para assegurar robustez, mecanismos similares aos disponíveis no QR Code padrão são especificados. Em particular, o código de correção de erro *Reed-Solomon* permite corrigir os módulos que representam mais do que um bit (GRILLO et al., 2010). O HCC2D preserva todas as funções padrão (*functions patterns*) do QR Code mantendo, assim, inteira compatibilidade com ele. As modificações mais importantes em relação ao QR Code padrão estão reunidas na área de dados e de correção de erros. A diferença mais notável está nos módulos que podem ser em diferentes cores, segundo a paleta de cores, que é composta de pelo menos quatro cores.

Durante a leitura do QR Code somente a informação de brilho é levada em conta, enquanto que no código HCC2D é necessário lidar com distorções cromáticas durante a fase de decodificação. Desde que o barcode seja composto de células coloridas, o decodificador precisa conhecer a paleta de cores completa para decodificar o símbolo. No HCC2D os campos de paletas de cores são replicadas no entorno do símbolo, tal como apresentado na Figura 4.12. Assim, a paleta de cores que não respeitar a sequência esperada de cores será descartada. Se um número mínimo de paletas de cores for reconhecida, pode-se construir a referência de cores obtida pela média de cada cor. O problema de identificação das cores no decodificador é resolvido utilizando a mínima distância Euclidiana entre os vetores em algum espaço de cores.

<sup>40</sup> O QR Code padrão em duas cores é considerado o caso mais simples do HCC2B.

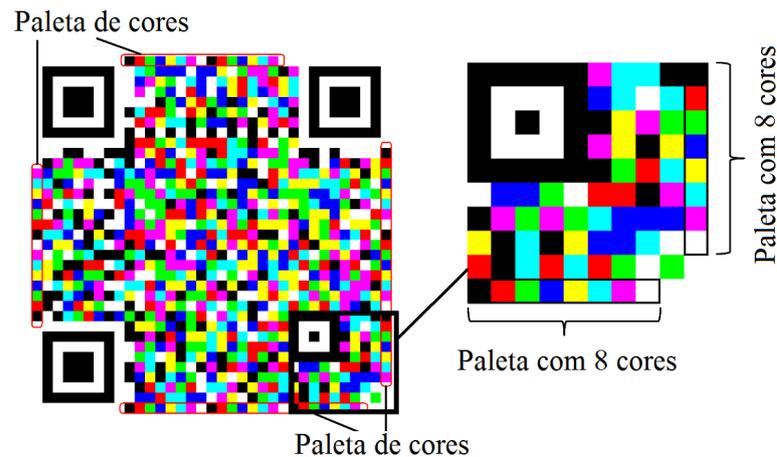


Figura 4.12 – Distribuição espacial das quatro paletas com oito cores (adaptado de (QUERINI et al., 2011)).

Observe que as cores de referência não estão muito próximas aos três padrões de localização e estão distantes umas das outras, assegurando robustez a distorções locais. Além disso, as cores na paleta são ordenadas em ordem decrescente de intensidade de brilho. Em seguida, a paleta é dividida em duas metades, sendo que a segunda metade é ordenada em ordem inversa em relação à primeira, a fim de aumentar a distância de brilho. Um outro detalhe é que as paletas de cores representam um *overhead* de apenas duas linhas e duas colunas para um mesmo símbolo. Deste modo, o *overhead* percentual reduz com o aumento da versão.

No QR Code padrão, a recuperação da informação de formato e de versão é uma etapa crucial da decodificação, e que pode conduzir a erros de leitura. Desta maneira, não há vantagem significativa na representação dessas funções do padrão em colorido, mesmo porque requerem pouco espaço de armazenamento (QUERINI; ITALIANO, 2013).

Já é de conhecimento que  $n$  cores podem aumentar a capacidade por módulo para  $BpM = \log_2(n)$  bits. Desta maneira, o código HCC2D permite aumentar a capacidade de dados do padrão ISO/IEC 18004:2006 (conforme Tabela 4.2), por um fator  $BpM$ , resultando na Tabela 4.3.

É notável a importância do mascaramento de dados no QR Code padrão com o nítido viés de alcançar um balanço apropriado entre módulos brancos e pretos, para evitar padrões similares àqueles de localização, alinhamento e temporização. Neste mesmo sentido, o código HCC2D necessita de robustez semelhante. No entanto, devido a codificação de mais de um bit por módulo, a opção de mascaramento padrão não é mais útil, pois as cores possuem brilhos diferentes, podendo levar a confusões com as funções padrão do QR Code. Para isso, o código

Tabela 4.2 – Capacidade de dados do QR Code padrão (ISO, 2006a).

Versão	Módulos Totais	Módulos do Padrão	Módulos Inf. de Versão e Formato	Módulos de Dados	Capacidade de Dados ( <i>codewords</i> )	Bits Restantes
1	21	202	31	208	26	0
2	25	235	31	359	44	7
3	29	243	31	567	70	7
4	33	251	31	807	100	7
5	37	259	31	1079	134	7
6	41	267	31	1383	172	7
7	45	390	67	1568	196	0

HCC2D é considerado uma matriz binária, cujos valores em brilho da metade superior são considerados brancos e os demais pretos. A identificação da melhor máscara utiliza o procedimento do QR Code padrão, baseado naquelas quatro regras já vistas.

Tabela 4.3 – Capacidade de dados do HCC2D (GRILLO et al., 2010).

Versão	Capacidade de Dados ( <i>codewords</i> )			Bits Restantes		
	4 cores (2 BpM)	8 cores (3 BpM)	16 cores (4 BpM)	4 cores (2 BpM)	8 cores (3 BpM)	16 cores (4 BpM)
1	52	78	104	0	0	0
2	88	132	176	14	21	28
3	140	210	280	14	21	28
4	200	300	400	14	21	28
5	268	402	536	14	21	28
6	344	516	688	14	21	28
7	392	588	784	14	21	28

Uma vez que a melhor máscara (monocromática) foi identificada as cores da região de codificação precisam ser modificadas por meio de uma operação XOR, conforme pode ser visto na Figura 4.13. Por exemplo, a operação de XOR entre a cor azul (01) do HCC2D e o preto da máscara (00) resulta em “01” que é a cor azul, ou seja, a cor azul é mantida.

#### 4.2.6 CQR Codes

Os *Colored Quick-Response Codes* (CQR Codes) utilizam a estrutura do QR Code para armazenar/transmitir informação empregando cinco diferentes cores<sup>41</sup>: vermelho, verde, azul,

<sup>41</sup> Um segundo modelo de CQR Code, evoluído do primeiro, utiliza a mesma estrutura da versão 8 do QR Code, porém emprega nove cores: preto, branco, vermelho, verde, azul, ciano, amarelo, magenta e cinza (MELGAR, 2013).

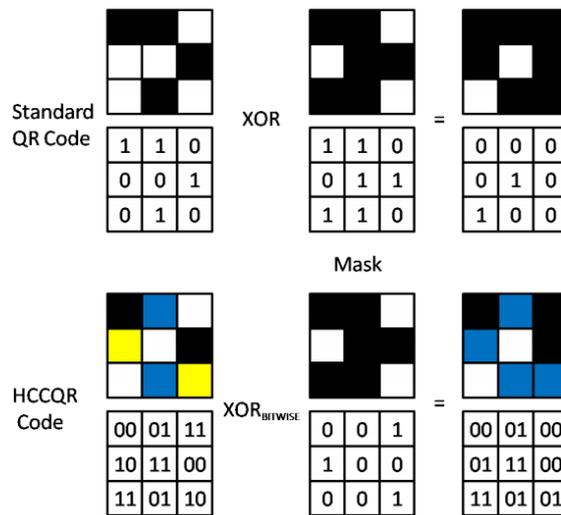


Figura 4.13 – Exemplo de aplicação de máscara no QR Code padrão e no HCC2D (adaptado de (GRILLO et al., 2010)).

preto<sup>42</sup> e branco, permitindo duplicar (dois bits por módulo) a capacidade do QR Code tradicional (Vizcarra Melgar et al., 2012). Essa classe de códigos também considera o uso do código *Reed-Solomon*, o que torna possível uma taxa teórica de correção de erro de 38,41%.

#### 4.2.6.1 Estrutura do Símbolo

O CQR Code é composto por  $49 \times 49$  módulos, distribuídos em duas regiões: região de codificação e funções padrão, conforme Figura 4.14 (a). As cores da região de codificação são representadas por “00” (vermelho), “01” (verde), “10” (azul) e “11” (branco) e codificam os bits de informação e de redundância do *Reed-Solomon*. Já a cor das funções padrão é preta sobre o fundo branco, determinando a zona vazia, padrões de localização e separadores. Observe que todas as demais estruturas presentes no QR Code padrão são suprimidas.

A partir dos 2401 módulos (versão 8 do QR Code), 192 estão na região padrão, enquanto que os restantes 2209 estão na região de codificação. Cada módulo representa dois bits, portanto, 4418 bits estão disponíveis, dos quais 1024 bits são dados e os demais 3392<sup>43</sup> são bits de redundância do *Reed-Solomon*. Os autores apresentam o armazenamento e a transmissão de dados criptografados em assinaturas digitais como possíveis aplicações dessa configuração com 1024 bits.

<sup>42</sup> A cor preta é destinada somente para utilização nos três módulos de localização.

<sup>43</sup> Um módulo não é utilizado, devido as características do *Reed-Solomon* empregado.

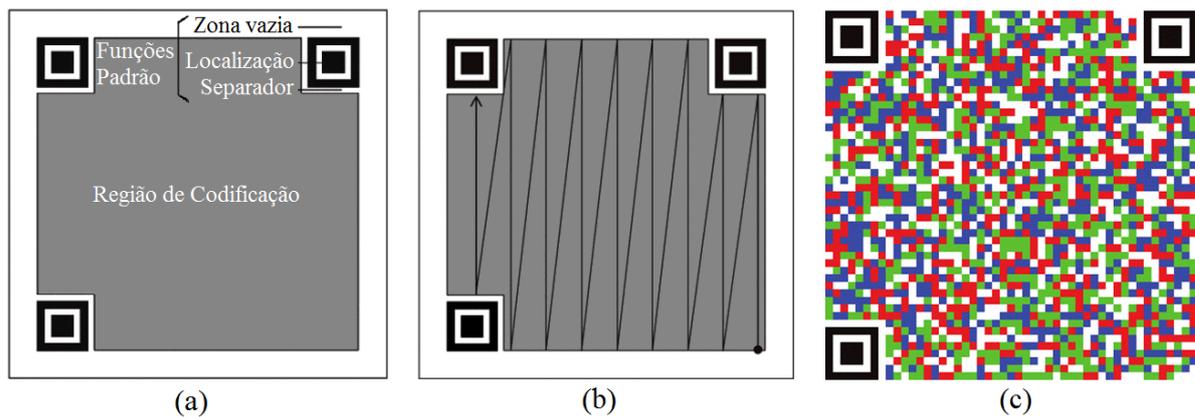


Figura 4.14 – Estrutura do CQR Code (adaptado de (Vizcarra Melgar et al., 2012)).

#### 4.2.6.2 Codificação e Decodificação do CQR Code

Durante a etapa de codificação a informação a ser transmitida é binarizada e, em seguida, os bits de redundância são adicionados. A cada sequência de dois bits uma cor das quatro possíveis é atribuída. Por fim, as cores são mapeadas na sequência apresentada na Figura 4.14 (b). O barcode produzido é, então, ilustrado na Figura 4.14 (c).

O processo de decodificação se inicia quando o CQR Code é detectado. A correção de rotações indesejadas no barcode é realizada pelos padrões de localização. Para isso, a imagem é convertida para escala de cinza e, depois, binarizada por um limiar  $th$  definido pela média das duas maiores modas ( $lm_1$  e  $lm_2$ ) do histograma, isto é,  $th = (lm_1 + lm_2)/2$ . Para este efeito, os autores propõem que  $lm_2$  possa ser estabelecido quando  $c_2 = 2 \times c_1$ , em que  $c_1$  define  $lm_1$ , conforme pode ser visto na Figura 4.15.

Uma vez realizada a binarização, seguida da detecção dos padrões de localização da imagem e, conhecendo-se a versão do barcode (fixa e igual a 8), é possível determinar o tamanho de cada módulo do barcode. Por fim, uma grade regular de amostragem sobre o barcode permite detectar a cor mais frequente em cada módulo.

É importante ressaltar que o CQR Code aqui apresentado não possui imunidade a distorções causadas pela inclinação do sensor da câmera. Assim, apenas imagens rotacionadas com a correção da orientação auferida pelos padrões de localização é realizado pelo CQR Code.

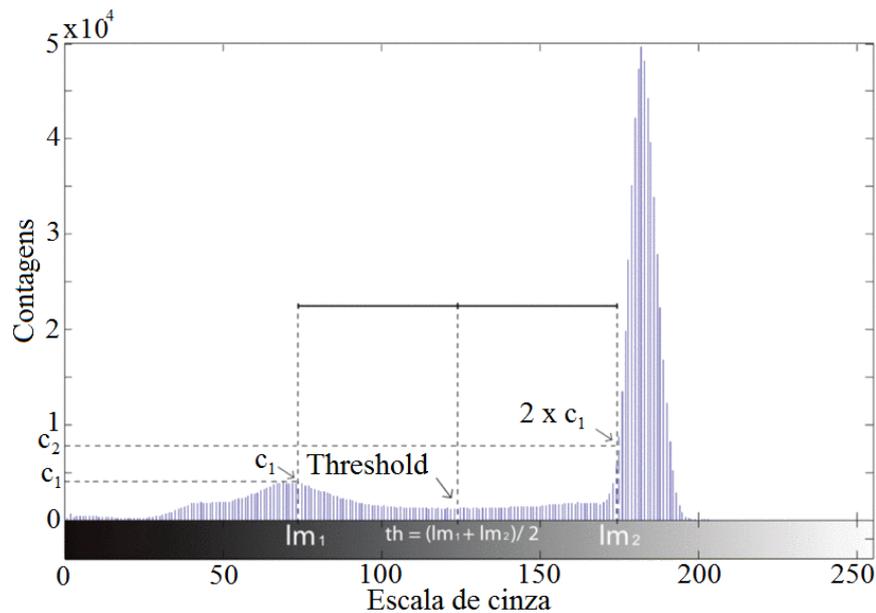


Figura 4.15 – Histograma típico usado para calcular  $th$  (adaptado de (Vizcarra Melgar et al., 2012)).

#### 4.2.7 QR Code Colorido Compatível com o Decodificador QR Padrão

Em (KIKUCHI; FUJIYOSHI; KIYA, 2013) é proposto um novo QR Code colorido compatível com o decodificador QR padrão para aumentar a informação codificada. Este método multiplexa três QR Codes monocromáticos no espaço de cores YCbCr, de modo que o canal Y possa ser decodificado pelo decodificador padrão do QR Code, enquanto que a decodificação dos canais Cb e Cr é realizado por um decodificador proprietário. A escolha do espaço de cores YCbCr está atrelada à possibilidade de o canal Y poder ser extraído, por diferença de luminância, diretamente do QR Code colorido multiplexado, utilizando o decodificador padrão do QR Code.

##### 4.2.7.1 Codificação

No algoritmo de codificação são contempladas as seguintes etapas (KIKUCHI; FUJIYOSHI; KIYA, 2013):

1. Os dados de entrada são divididos em três porções, geradas a partir do codificador bicolor padrão do QR Code da Figura 2.9.
2. Dois níveis de intensidade para os módulos preto e branco em cada canal de cor do

YCbCr,  $(Y_1, Y_2)$ ,  $(C_{b1}, C_{b2})$  e  $(C_{r1}, C_{r2})$  são selecionados, de tal modo que estejam tão mais afastados quanto possível. Observe que os valores dos módulos preto e branco não são configurados para os valores mínimo (16) e máximo (236), respectivamente, no espaço de cores YCbCr<sup>44</sup>. O procedimento deste item tem o interesse de selecionar as cores que estejam mais afastadas<sup>45</sup> entre si, tanto no espaço de cores YCbCr quanto no RGB.

3. Os três QR Codes no espaço YCbCr são convertidos para RGB de acordo com as seguintes relações:

$$\begin{cases} R = 1,164(Y - 16) + 1,596(Cr - 128) \\ G = 1,164(Y - 16) - 0,391(Cb - 128) - 0,813(Cr - 128) \\ B = 1,164(Y - 16) + 2,018(Cb - 128). \end{cases} \quad (4.1)$$

A Figura 4.16 ilustra os três QR Codes (versão 3 e nível de correção de erro H) que são utilizados para construir o barcode colorido. O decodificador padrão do QR Code será capaz de ler a informação do canal Y (Figura 4.16 (a)) diretamente a partir do barcode colorido (Figura 4.16 (d)).



Figura 4.16 – Exemplo de QR Code colorido multiplexado a partir dos canais YCbCr independentes (adaptado de (KIKUCHI; FUJIYOSHI; KIYA, 2013)).

<sup>44</sup> Equivale ao intervalo  $[0, 255]$  no espaço de cores RGB para imagens oito bits.

<sup>45</sup> Em (KATO; TAN; CHAI, 2009) foi realizado um estudo similar com o intuito de selecionar as cores mais afastadas conjuntamente nos espaços de cores RGB e YCbCr.

#### 4.2.7.2 Decodificação

O processamento da decodificação ocorre em duas etapas. A primeira é realizada baseada no decodificador padrão do QR Code (veja a Figura 2.13), utilizando apenas a luminância. Nesta etapa, o canal de cor  $Y$  é decodificado a partir do barcode colorido. Na segunda etapa, por sua vez, a informação de crominância dos outros dois canais de cor são decodificadas com a utilização de um decodificador proprietário. A imagem capturada no formato RGB é convertida para o espaço de cores YCbCr, conforme segue:

$$\begin{cases} Y = 0,257R + 0,504G + 0,098B + 16 \\ Cb = -0,148R - 0,291G + 0,439B + 128 \\ Cr = 0,439R - 0,368G - 0,071B + 128. \end{cases} \quad (4.2)$$

Em seguida, cada canal de cor YCbCr é decodificado isoladamente usando o decodificador padrão do QR Code.

A principal inovação proposta em (KIKUCHI; FUJIYOSHI; KIYA, 2013) é, justamente, manter a compatibilidade com o decodificador padrão, ancorando-se, para isso, na informação de luminância do QR Code Colorido. Isso habilita que os três QR Codes na etapa de codificação podem ser totalmente diferentes, tanto em versão como nível de correção de erro. Desta maneira, a informação mais importante é codificada no canal  $Y$ , pois este pode ser decodificado pelo padrão QR Code. Os canais  $Cb$  e  $Cr$  transmitem, então, informações complementares.

## 5 CÓDIGO DE BARRAS COLORIDO PROPOSTO

Este capítulo apresenta o *framework* desenvolvido para essa dissertação, o qual também é adequado para prosseguir a exploração futura deste trabalho. Inicialmente, o barcode colorido proposto, baseado no padrão QR Code, é devidamente apresentado. Em seguida, são realizadas algumas análises relacionadas às distorções geométricas e cromáticas, intrínsecas à transmissão de dados por códigos de barras coloridos. Para a análise das distorções cromáticas, um modelo de canal genérico, que emula as alterações tonais em barcodes coloridos, é utilizado. Com o intuito de minimizar essas distorções na cor, algumas técnicas de correção cromática são propostas, baseadas em cores de referência.

### 5.1 QR Code Colorido Proposto

Diante de todas as assimetrias relacionadas à cor retratadas no Capítulo 3, e das estruturas dominantes dos barcodes coloridos existentes detalhadas no Capítulo 4, neste capítulo as características atribuídas mais importantes destas referências são sintetizadas. Neste contexto, propomos um barcode colorido, o QR Code Composto (*Composite QR Code - CQC*), baseado no padrão QR Code, com as seguintes principais características:

- Utiliza oito cores (vermelho, verde, azul, ciano, magenta, amarelo, preto e branco), o que permite um armazenamento de dados com capacidade três vezes superior ao QR Code preto e branco;
- Utiliza os canais R, G e B para codificar três barcodes monocromáticos que, reciprocamente, podem ser decodificados pelo decodificador padrão do QR Code;
- O canal G do barcode colorido pode ser diretamente extraído pelo decodificador padrão, por diferença de luminâncias;
- Possui características totalmente compatíveis com o padrão QR Code, em especial, a robustez;
- Apresenta cores de referência, ativas na compensação de desvios cromáticos.

A escolha do QR Code é baseada em sua elevada disseminação em aplicativos *mobiles* pelo mundo, alta imunidade a erros, fácil detecção (legibilidade) e ampla versatilidade atribuída as suas 40 versões.

O CQC consiste em três QR Codes sobrepostos em diferentes canais de cores. A correção das distorções cromáticas é realizada por cores de referências justapostas e circundantes ao barcode. Uma vez executada as correções tonais, as três camadas sobrepostas são separadas, amostradas, binarizadas e decodificadas, segundo o algoritmo padrão do QR Code.

Para o desenvolvimento e análise do CQC uma plataforma composta pelo codificador do padrão QR Code e pela combinação aditiva dos canais R, G e B foi desenvolvida em linguagem Matlab, conforme ilustra a Figura 5.1. As distorções mais comuns no canal de imagem, para avaliar o desempenho do barcode proposto, também foram implementadas em código *M-file*. A correção cromática é realizada utilizando três métricas, a saber: mínimas distância Euclidiana e Mahalanobis e *k-means*. Já o decodificador possui métodos implementados sobre a biblioteca *Zxing*<sup>46</sup>. O intercâmbio entre o codificador e o decodificador é realizado em *workspace* Matlab.

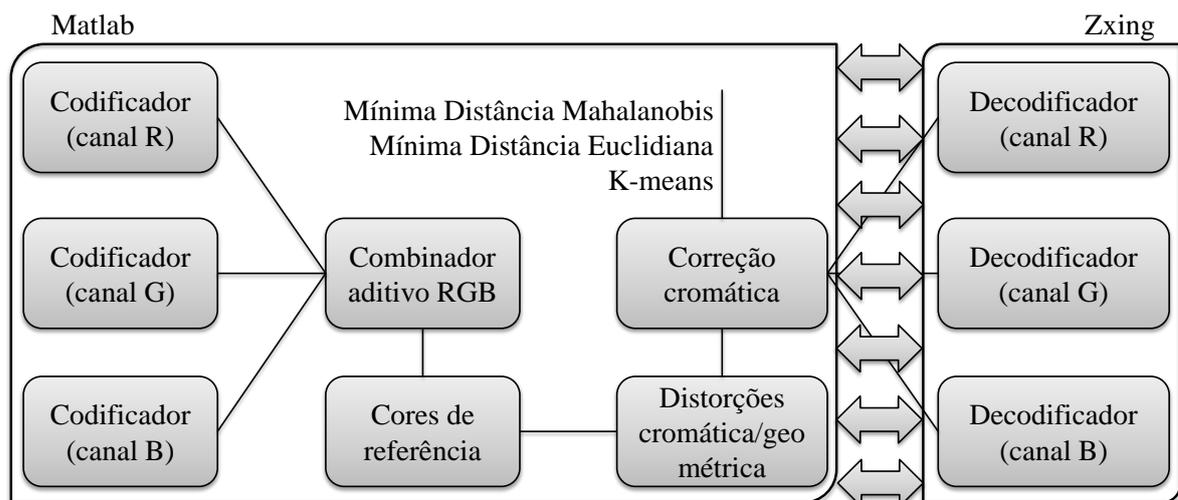


Figura 5.1 – Esquema de codificação do CQC.

A seguir, o desenvolvimento do CQC é seccionado conceitualmente em três etapas: codificação, detecção e decodificação.

### 5.1.1 Codificação

O código CQC propõe transformar três barcodes monocromáticos do QR Code em um barcode colorido. Para isso, utiliza de maneira sobreposta e aditiva as três cores primárias

<sup>46</sup> A *Zxing* é uma biblioteca de processamento de imagens para barcodes 1D/2D voltada ao desenvolvimento em Android, utilizando a linguagem Java.

vermelho, verde e azul. A informação a ser transmitida é dividida em três partes,  $d_1$ ,  $d_2$  e  $d_3$ , sendo cada uma delas alocada independentemente nos canais R, G e B, respectivamente, conforme ilustrado pela Figura 5.2. Essa composição de camadas do QR Code permite triplicar a capacidade de dados do padrão bicolor do QR Code, em consonância com a Tabela 4.3. Os dados em cada canal de cor utilizam o código corretor de erro do padrão monocromático, permitindo a correção dos erros na decodificação.

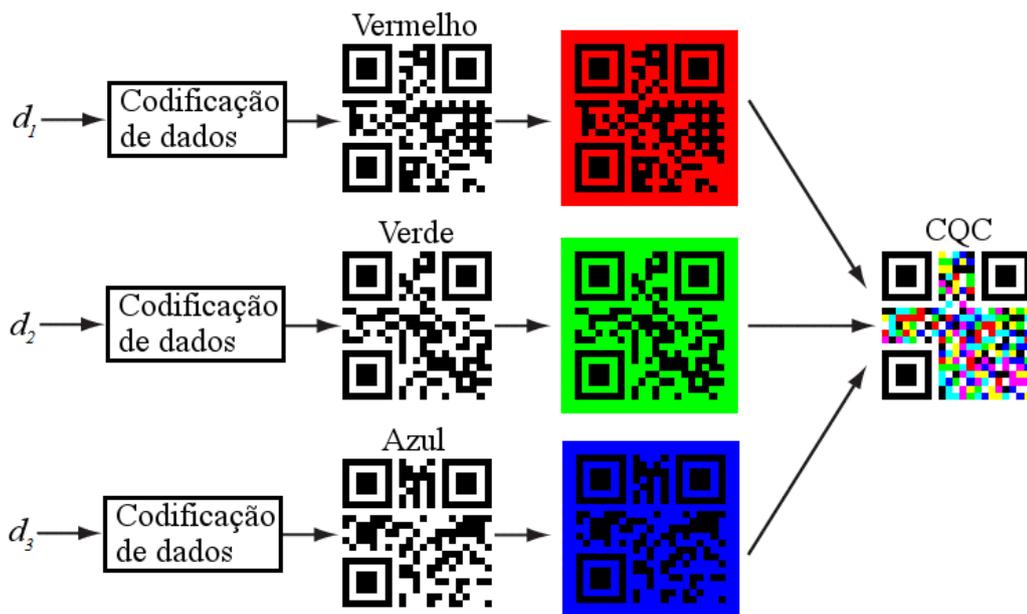


Figura 5.2 – Esquema de codificação do CQC.

As informações de versão e formato do padrão QR Code podem ser mantidas nas cores preta e branca, mesmo na composição do barcode CQC. Para isso, as versões dos QR Codes devem ser as mesmas, o que é bastante provável já que os dados podem ser divididos em três partes iguais. Além disso, é necessário que o nível de correção de erro e máscara utilizada nos três barcodes sejam iguais. No entanto, obter-se três máscaras idênticas é extremamente complicado, pois, como vimos, quatro penalidades decidem pela melhor máscara. Embora o uso de máscaras idênticas possa ser vantajoso na detecção das informações de formato, deve-se avaliar quanto a decodificação, baseada no padrão QR Code, será penalizada em função da utilização de uma mesma máscara nos três barcodes.

### 5.1.2 Detecção

Da mesma forma que em (KIKUCHI; FUJIYOSHI; KIYA, 2013), uma importante inovação deste trabalho defronte aos códigos PM Code, HCCB, HCC2D, MMCC e CQR, reside na abordagem dos canais de cores como independentes, sem qualquer mapeamento de cores. No trabalho citado, os autores procedem de maneira similar, porém multiplexando os três canais YCbCr, cada qual com três QR Codes independentes, como já foi ilustrado na Figura 4.16.

Vinculada à proposição do parágrafo anterior, o CQC possibilita que o decodificador padrão do QR Code detecte um dos três canais RGB, decodificando parcialmente o barcode colorido (em princípio, 1/3 da informação armazenada). O QR Code bicolor do canal verde é diretamente decodificado pelo decodificador padrão no CQC, pois ele somente utiliza da variação de luminância da imagem, conforme foi detalhado na Equação 2.5. Desta maneira, o restante da informação nos canais vermelho e azul pode ser extraída pelo decodificador do CQC. Essa proposição considera a utilização disseminada do decodificador padrão do QR Code, minimizando o prejuízo para o usuário que ainda não está equipado com o decodificador do CQC. Em outras palavras, os usuários de aplicativos baseados no decodificador padrão ainda estão habilitados a extrair a informação armazenada no canal G do código CQC.

Estas considerações oportunizam que o canal verde (aquele capaz de ser decodificado pelo QR Code padrão) carregue a informação principal do CQC, deixando, assim, para os canais vermelho e azul informações complementares. Deste modo, diferentes versões e níveis de correção de erro do QR Code podem ser utilizados. Na Figura 5.3, por exemplo, os canais vermelho e azul estão na versão e nível de correção de erro 1-L, ao passo que o canal verde armazena dados com a configuração 4-H.

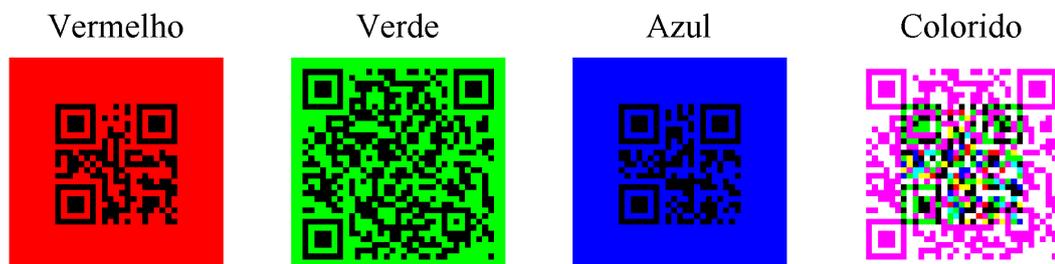


Figura 5.3 – Canal verde carrega a informação principal do barcode, enquanto que os canais vermelho e azul trazem as informações complementares.

A detecção do CQC baseada no padrão do QR Code detecta o barcode a partir dos valo-

res de luminância. No entanto, a luminância não varia tão somente com o valor da intensidade do espaço de cores HSI, mas também com a matiz e a saturação da cor. Neste sentido, é fácil perceber que determinadas matizes são mais brilhantes<sup>47</sup> do que outras. Neste sentido, as cores amarela, ciano, verde com luminâncias de 94%, 87% e 80%, respectivamente, além do próprio branco (luminância de 100%) são mapeadas para branco pelo decodificador padrão. Já as cores azul, vermelha e magenta com luminâncias de 44%, 54% e 70%, nesta ordem, além do preto (luminância nula) são convertidas para o preto. Observe que esse mapeamento coincide exatamente com o canal verde que será extraído do barcode colorido diretamente pelo decodificador padrão.

Na prática, devido ao mapeamento de cada matiz em um determinado nível de cinza o limiar de binarização para mapear as quatro cores para branco e as outras quatro para preto pode variar, junto com qualquer oscilação da matiz e saturação da cor. Os testes realizados mostraram que o limiar estabelecido pelo QR Code (Equação 2.5) fica muito próxima da luminância do magenta. Desta forma, o magenta pode, dependendo das condições de brilho da imagem, ser mapeado incorretamente para o branco. Neste caso, tendências mais escurecidas<sup>48</sup> de imagem tendem a produzir resultados satisfatórios, pois trazem o magenta para baixo do valor limiar. Uma solução para minimizar este problema é mapear a combinação do vermelho com o azul (isto é, magenta) para um tom mais escuro, por exemplo, de (1 0 1) para (0,95 0 0,95). Nestas condições há que se avaliar qual a deterioração, em termos de distanciamento entre cores, inserida por esta operação.

#### 5.1.2.1 Cores de Referência

As técnicas de codificação de canal, em especial o *Reed-Solomon*, são capazes de corrigir erros e recuperar os dados originais, sob a exigência de que a redundância inserida na etapa de codificação seja suficiente. No entanto, quanta redundância é necessária para um barcode colorido? Obviamente, isto depende de uma série de fatores, incluindo o algoritmo usado para a classificação da cor. O acréscimo de robustez ao código de barra, pela redução da sensibilidade tonal do barcode, permite aumentar a *goodput* da região de codificação. No caso do CQC, é possível reduzir o nível de correção de erro se nele for implementado um algoritmo que o torne menos sensível a variações de cor.

<sup>47</sup> A luminância é uma medida da percepção do brilho.

<sup>48</sup> Um teste interessante que comprova isso (caso alguma imagem do CQC colorido desta dissertação não seja decodificado) é diminuir o brilho da tela.

A utilização das cores de referência no código CQC está associada à necessidade de minimizar variações tonais nas cores do barcode, maximizando sua robustez. Mesmo que não se possa garantir a efetividade, é possível afirmar que o balanceamento de cores pode mitigar o efeito da iluminação desigual, ou sombreamento do barcode. Nestas condições, quanto melhor for a distribuição espacial das cores de referência no código de barra mais robusto ele será.

Atentos a esses requisitos, propõe-se<sup>49</sup> a distribuição das oito cores codificáveis de maneira replicada entorno do CQC, conforme mostra a Figura 5.4. Observe que as cores de referência não estão próximas aos módulos de detecção do padrão QR Code, bem como apresentam-se afastadas umas das outras (em lados opostos), assegurando robustez a distorções locais. Ademais, quanto maior a versão do QR Code maior será a quantidade de amostras da *string* de oito cores em torno do CQC. O *overhead* propiciado pelas paletas de cores é de apenas duas linhas e duas colunas, em qualquer uma das versões do QR Code.

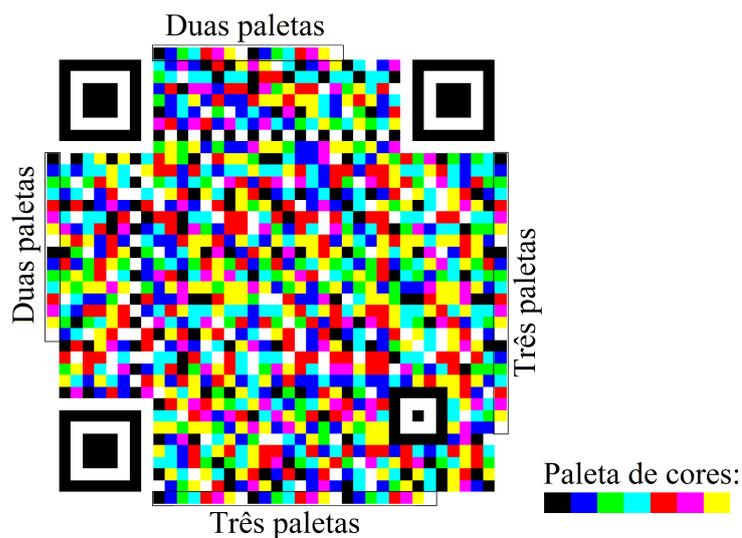


Figura 5.4 – Ilustração da paleta de cores.

A partir da utilização das cores da paleta de cores a nuvem de pontos constituintes de cada agrupamento, ou *cluster*, deve ser rotulada<sup>50</sup>. Para isso, neste trabalho, foram utilizadas três métricas que identificam a menor distância entre a cor amostrada e a cor de referência. São elas:

<sup>49</sup> No Apêndice A é apresentada uma outra proposta para calibração dos módulos de cor do QR Code baseado em blocos piloto. No entanto, esta técnica ainda carece de uma análise mais rigorosa.

<sup>50</sup> Este procedimento consiste em converter os valores dos módulos analógicos para *strings* digitais.

- Mínima Distância Euclidiana:

A distância Euclidiana, já referida na Equação 3.8, é a mais simples e popular medida de distância. Ela atribui um rótulo de classe que minimiza a distância entre os dados não rotulados e as classes<sup>51</sup> no espaço RGB de características. A mínima distância representa máxima similaridade.

- *K-means*:

O *k-means* é um método de classificação que secciona  $n$  cores capturadas em  $k$  rótulos<sup>52</sup> de cor, onde cada cor capturada é classificada para o rótulo mais próximo a ela. Este seccionamento resulta em uma divisão das cores capturadas em um Diagrama de *Voronoi*<sup>53</sup>. A convergência do *k-means* ocorre de maneira iterativa<sup>54</sup>, de tal modo que a média dos rótulos de cor é atualizada a cada iteração em busca de seus centroides.

- Mínima Distância Mahalanobis:

A distância euclidiana é extremamente sensível às variações de escala das grandezas envolvidas, bem como cega à correlação entre elas. Neste sentido, medidas consideradas *outliers* podem ser equivocadamente classificadas. A fim de suplantar essas limitações a distância de Mahalanobis  $d(\vec{x}, \vec{y})$  é baseada na correlação  $\Sigma$  entre dois vetores  $\vec{x}$  e  $\vec{y}$ , de acordo com a seguinte equação:

$$d(\vec{x}, \vec{y}) = \sqrt{(\vec{x} - \vec{y})^T \Sigma^{-1} (\vec{x} - \vec{y})} \quad (5.1)$$

*Nota:* Se a matriz  $\Sigma$  for identidade, a distância de Mahalanobis coincide com a distância Euclidiana.

### 5.1.3 Decodificação

Os dispositivos de captura dos barcodes são tipicamente fotossensíveis às cores vermelho, verde e azul, de modo que cada um desses canais de cores reproduzem a presença, ou a ausência da respectiva cor, conforme ilustra a Figura 5.5. Estas cores amostradas, com componentes nos três canais de cor, tendem a formar agrupamentos tais como os vistos na Figura 3.12. Estes *clusters* de cores devem ser rotulados, por meio das mínimas distância Euclidiana e de Mahalanobis, conforme vimos.

<sup>51</sup> Isto é, cores de referência.

<sup>52</sup> Se o número de cores é oito, então,  $k = 8$  (agrupamentos de cor).

<sup>53</sup> Diagrama de *Voronoi* é um tipo especial de decomposição em um dado espaço de cor, no qual a distância de uma cor em relação a um rótulo não é maior que sua distância para os demais rótulos de cor.

<sup>54</sup> Na prática, pode-se fixar um limiar de convergência, ou um número máximo de iterações.

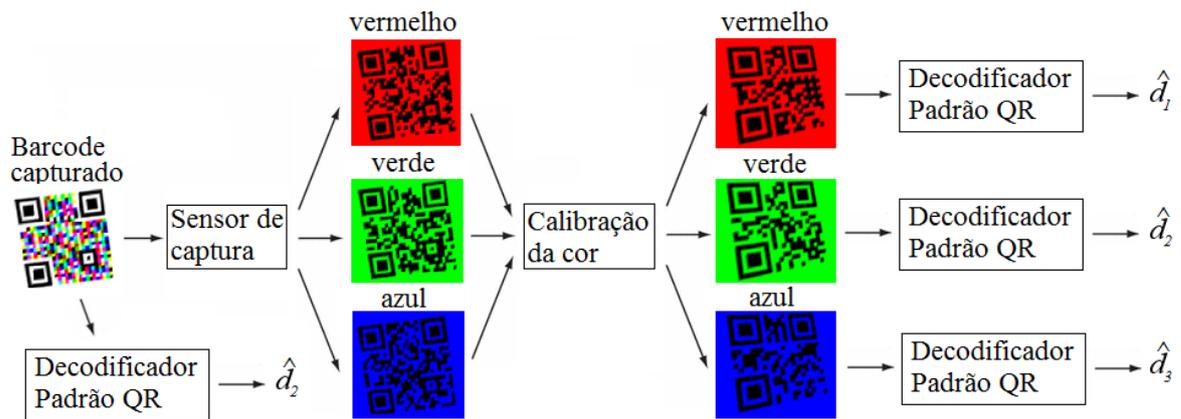


Figura 5.5 – Esquema de detecção e decodificação do CQC.

Durante o processo de digitalização do barcode colorido, o decodificador precisa conhecer previamente as cores de referência para realizar a correção das distorções cromáticas. Desta forma, através das cores de referência, os módulos coloridos da região de codificação são classificados para uma das oito cores (classes, ou etiquetas) do CQC. Em cenários reais uma porcentagem de cores é classificada de maneira equívoca, surgindo erros de bits que produzem uma “suave” diferença entre a *bitstream* codificada e aquela decodificada. Torna-se, então, evidente que quanto melhor o resultado sintetizado pelo algoritmo de classificação menor a quantidade de redundância requerida pelo código corretor de erro do barcode.

Por fim, os canais de cores calibrados são enviados separadamente para o decodificador padrão do QR Code que procede a correção dos erros e detecção dos dados  $\hat{d}_1$ ,  $\hat{d}_2$  e  $\hat{d}_3$ . Da mesma forma, o decodificador padrão pode detectar o dado  $\hat{d}_2$  diretamente a partir do CQC, com já foi apurado anteriormente.

Esta metodologia que utiliza códigos de barras sobrepostos na síntese de barcodes coloridos habilita o reuso de otimizações e melhorias propostas para os barcodes monocromáticos. Por exemplo, a utilização de técnicas de *thresholding* adaptativo para lidar com variações nos níveis de iluminação, e diferentes esquemas de correção de erro desenvolvidos para tratar de erros típicos em barcodes são diretamente aplicáveis a cada um dos canais R, G e B. Em linhas gerais, pode-se expandir essa metodologia para qualquer barcode, não somente ao QR Code.

## 5.2 Análise das Distorções Geométricas

Esta seção tem o objetivo de verificar quais os limites paramétricos de distorções às quais o decodificador padrão do QR Code é capaz de estimar a informação armazenada. Para isso, a métrica utilizada é a *synchronization success rate* (SSR), ou taxa de sucesso na sincronização. Ela considera que o decodificador será capaz de extrair as informações armazenadas se três condições ocorrerem: a) os módulos de localização forem encontrados na imagem; b) a informação de formato e versão forem lidas com sucesso; e c) a quantidade de erros identificada é inferior à capacidade de correção de erro do *Reed-Solomon*.

A Figura 5.6 ilustra diferentes tipos de distorções geométricas emuladas em computador. As imagens apresentadas correspondem aos limites verificados no decodificador padrão do QR Code para cada uma das distorções. A dimensão da imagem utilizada foi de  $120 \times 120$  pixels.

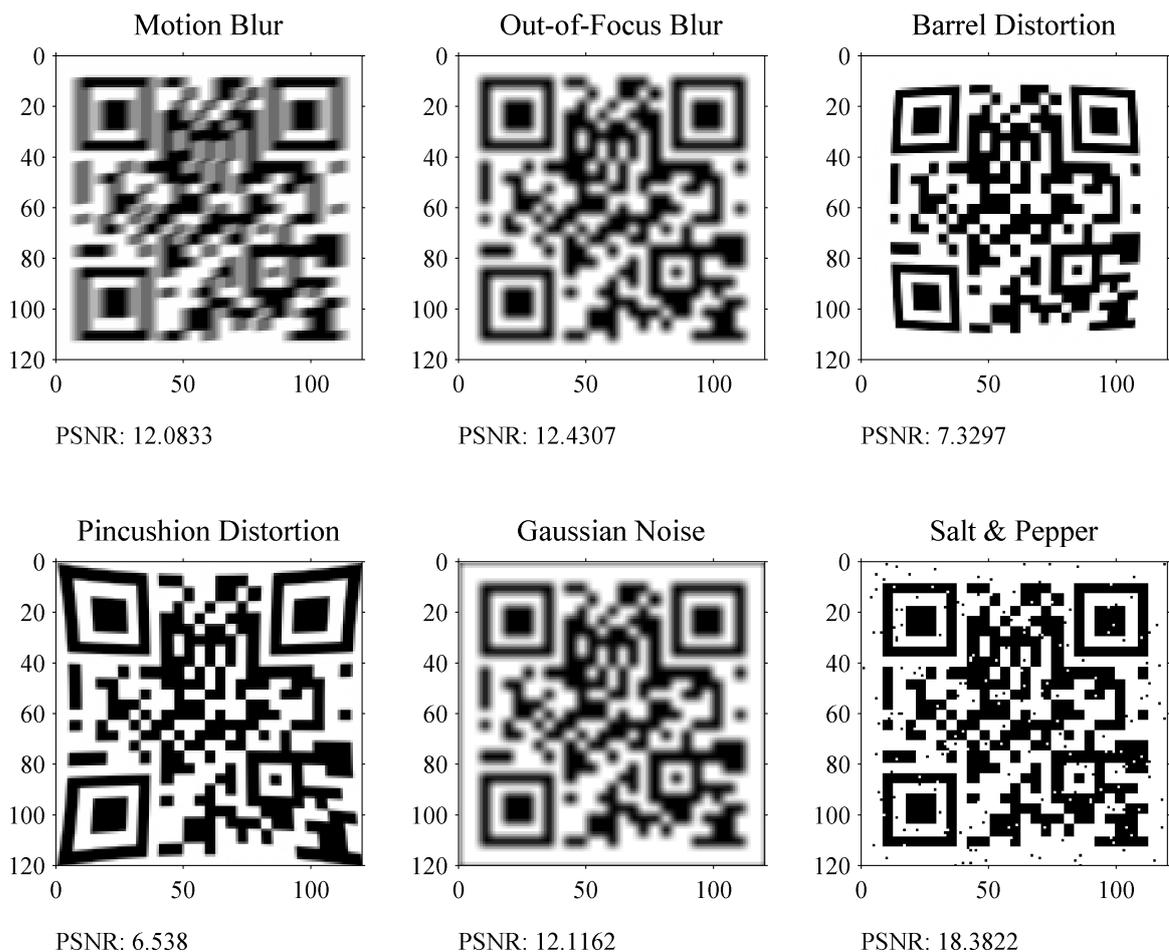


Figura 5.6 – Limites paramétricos de distorções no QR Code para uma imagem de  $120 \times 120$  pixels.

A medida utilizada para mensurar o dano causado às imagens da Figura 5.6 é a *peak signal to noise ratio* (PSNR). Um valor elevado de PSNR representa uma imagem menos agredida por distorções, ou mais fidedigna<sup>55</sup> em relação à imagem monocromática<sup>56</sup> de referência. Nesta análise, porém, quanto maior o valor da PSNR obtida melhor é a robustez do decodificador padrão do QR Code a este tipo de distorção. Portanto, com base na Figura 5.6, pode-se concluir que o QR Code é mais robusto a distorções do tipo *salt & pepper*, ou ruído impulsivo<sup>57</sup>. Em seguida, as distorções de natureza visualmente similar *motion blur*, *out-of-focus blur* e ruído Gaussiano, apresentaram tolerâncias similares. O decodificador apresentou-se menos robusto às distorções radiais, comumente verificadas nas lentes das câmeras.

A PSNR pode ser definida pelo erro médio quadrático (MSE), que é uma medida estatística do erro usado para determinar a qualidade da imagem (KATO; TAN; CHAI, 2010). O MSE para duas imagens  $I$  e  $K$  de mesma dimensão,  $m \times n$ , é definido pela relação:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} \| I(i, j) - K(i, j) \|^2, \quad (5.2)$$

em que uma das imagens é a imagem de referência.

Portanto, a PSNR pode ser descrita como:

$$PSNR = 10 \cdot \log_{10} \left( \frac{MAX_I^2}{\sqrt{MSE}} \right) = 20 \cdot \log_{10} \left( \frac{MAX_I}{\sqrt{MSE}} \right) \quad (5.3)$$

em que  $MAX_I^2$  é o valor máximo de um pixel da imagem. Se a imagem for de 8 bits *greyscale*, ou 24 bits RGB, o valor máximo será 255.

Considerando-se que o CQC proposto não altera as estruturas de localização do QR Code, não foram verificadas alterações nos três canais monocromáticos R, G e B, no que tange ao sucesso da sincronização. Na proposta veiculada no Apêndice A deve-se avaliar os efeitos da redução de contraste (em termos de luminância máxima e mínima) no desempenho da SSR. Estima-se que a SSR apresentará um desempenho inferior, em função das novas características cromáticas dos módulos de localização.

<sup>55</sup> Se duas imagens forem iguais a PSNR será infinita.

<sup>56</sup> Em imagens coloridas, PSNRs elevadas representam cores mais precisas.

<sup>57</sup> O ruído impulso não é, de fato, uma distorção geométrica. Contudo, é colocado nesta seção, para fins comparativos.

### 5.3 Análise das Distorções Cromáticas

#### 5.3.1 Modelagem do Canal de Cor

A estratégia comumente utilizada na literatura para analisar as distorções cromáticas do canal, envolvendo qualquer meio de impressão/digitalização<sup>58</sup> de barcodes consiste na reprodução do cenário real, especificando os dispositivos envolvidos e analisando os resultados auferidos. Em contraponto a esta eficaz estratégia, propomos neste trabalho uma modelagem genérica das variações tonais sintetizadas pelos dispositivos envolvidos na comunicação por imagem, sem nos preocuparmos especificamente com este ou aquele dispositivo. A partir de trabalhos propostos na literatura e previamente ilustrados na Figura 3.12, adota-se a modelagem das cores baseada na matriz simétrica de covariância:

$$\Sigma = \begin{bmatrix} \sigma_{rr} & \sigma_{rg} & \sigma_{rb} \\ \sigma_{gr} & \sigma_{gg} & \sigma_{gb} \\ \sigma_{br} & \sigma_{bg} & \sigma_{bb} \end{bmatrix}, \quad (5.4)$$

em que os elementos da diagonal principal  $\sigma_{rr}$ ,  $\sigma_{gg}$  e  $\sigma_{bb}$  são as variâncias amostrais dos canais vermelho, verde e azul, respectivamente; e os elementos simétricos  $\sigma_{rg}$  e  $\sigma_{gr}$ ,  $\sigma_{br}$  e  $\sigma_{rb}$  e  $\sigma_{bg}$  e  $\sigma_{gb}$  denotam a variação conjunta entre as cores vermelha e verde, azul e vermelha e verde e azul, nesta ordem.

No gráfico de dispersão da Figura 5.7 estão plotados exemplos de distribuição das oito<sup>59</sup> cores, considerando variações tonais de cada uma delas. A dispersão das cores obedecem a uma regra de formação estabelecida pela matriz de covariância da Equação 5.4. Observe que determinadas cores têm comportamentos que se aproximam de uma distribuição Gaussiana, ou Normal<sup>60</sup>. Por outro lado, algumas cores possuem variação conjunta em uma determinada direção, caracterizadas por um nuvem de pontos cuja geometria se assemelha a um elipsoide.

Quanto maior a tendência do canal de imagem produzir distribuições tonais próximas à Gaussiana, maior a facilidade de distinção entre as cores. Essa característica pode ser melhor observada na Figura 5.8, na qual cada uma das faces do cubo da Figura 5.7 são plotadas. Quanto mais coesa uma mesma cor estiver (menor a variabilidade em torno da média, definida pela Equação 3.10), mais distante ela estará das demais, facilitando a decodificação.

<sup>58</sup> Aqui estão compreendidos diferentes tipos de canais, considerando como dispositivos de saída a tela, ou impressora, e como dispositivos de entrada o *scanner* e a câmera.

<sup>59</sup> Observe a existência de um maior número de pontos das cores branca e preta, pois estes módulos estão presentes em maior quantidade no barcode. Ainda a cor branca é impressa como cinza, para fins de visualização.

<sup>60</sup> Uma distribuição com forte tendência Normal é caracterizada por uma matriz de covariância que se aproxima de uma matriz diagonal.

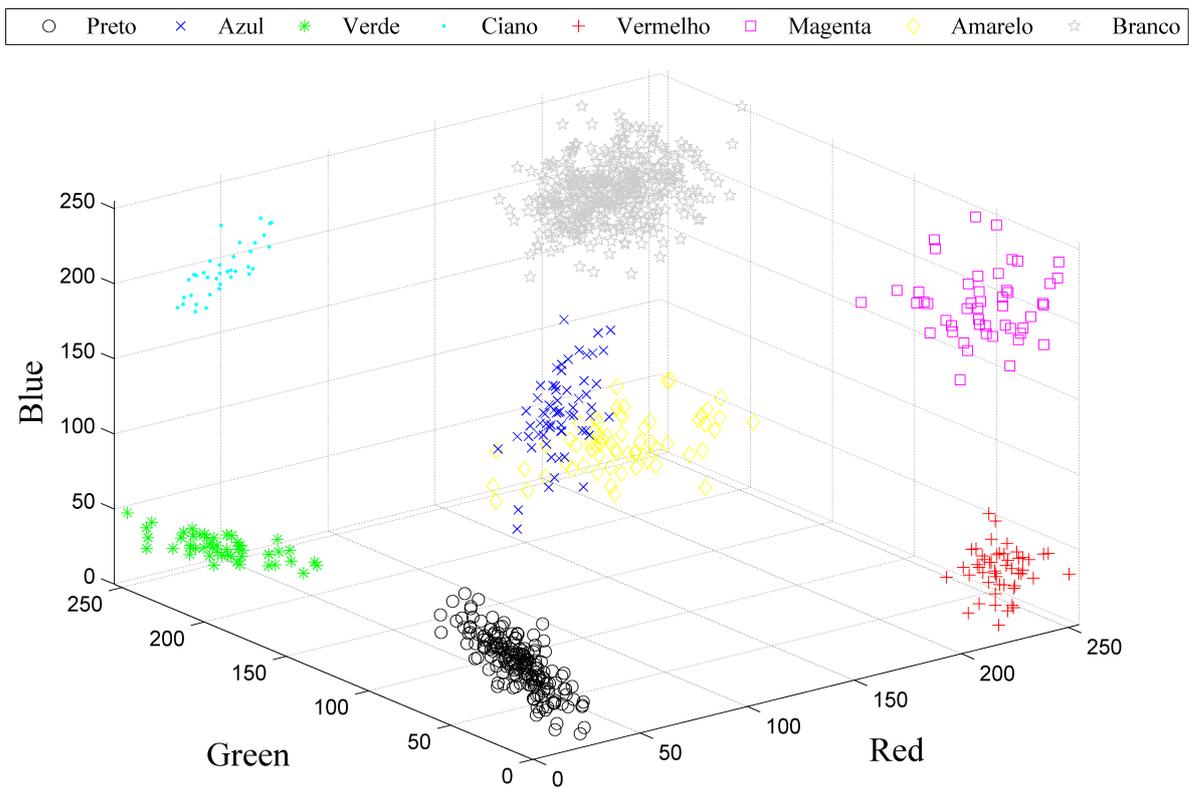


Figura 5.7 – Gráfico de dispersão das oito cores utilizadas na síntese do código CQC.

O resultado da inserção das variantes tonais<sup>61</sup> no QR Code pode ser visto na Figura 5.9. Na primeira linha de barcodes da Figura 5.9 estão os três canais RGB inicialmente codificados e o código colorido resultante. Já na segunda linha, os canais de cor são submetidos ao efeito do canal, produzindo variações cromáticas. Elas produzem uma imagem, cujas cromaticidades são distorcidas em relação a imagem codificada. Observe que o cenário proposto na Figura 5.9 pode ser considerado como um *corner* de simulação, uma vez que dificilmente os módulos do QR Code irão variar tão brutalmente em relação à imagem codificada. Neste sentido, as cores do fundo, notadamente branco, também variam.

No Apêndice B é apresentado um outro exemplo, ilustrando uma dispersão maior dos dados em torno de seu valor médio.

<sup>61</sup> Neste estágio, estamos apenas interessados nas variações cromáticas percebidas na imagem, sem nenhuma preocupação com qualquer outro tipo de distorção.

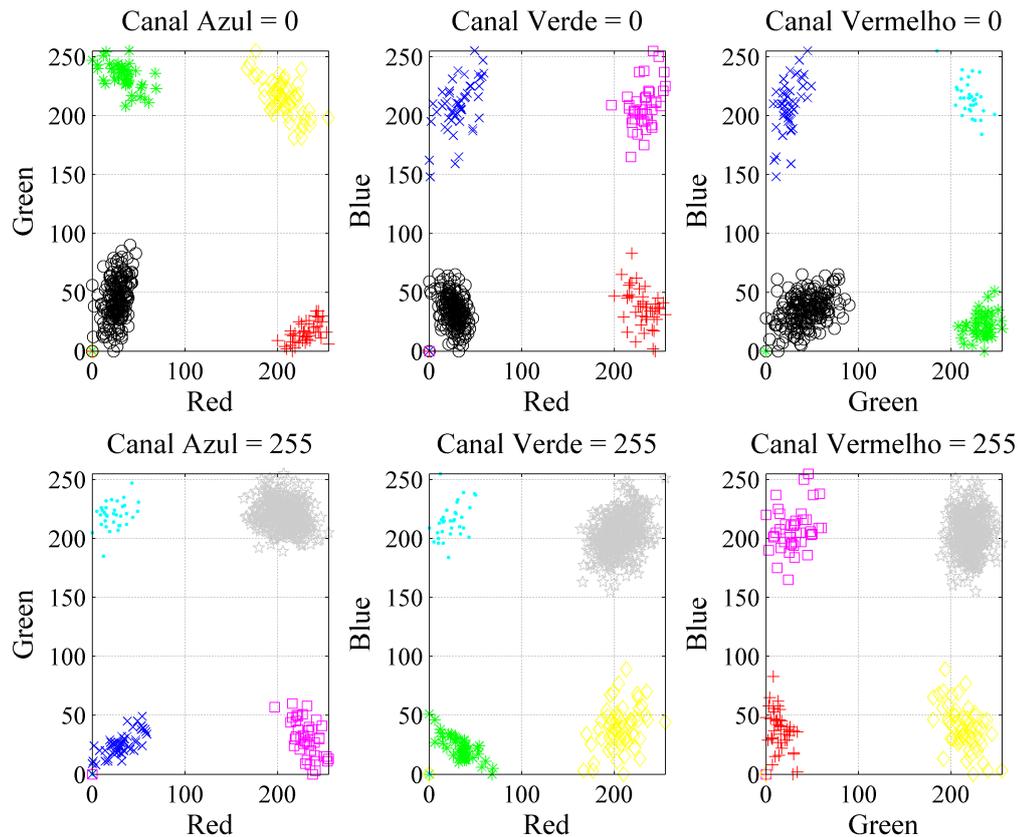


Figura 5.8 – Dispersão das cores sob a vista de cada uma das faces do cubo RGB.

### 5.3.2 Análise da BER

O desempenho do CQC, baseado nas cores de referência, é avaliado utilizando como métrica a *bit error rate* (BER). A BER é a porcentagem de erro de bits na imagem extraída comparada com a imagem codificada, previamente à aplicação dos mecanismos de correção, isto é, do *Reed-Solomon*. No eixo das abcissas a referência adotada é a PSNR da imagem capturada. É importante salientar que, mesmo restringindo os experimentos desta subseção ao CQC, a maioria dos resultados podem ser generalizados para qualquer outro barcode colorido, sob os efeitos do canal considerado. Além disso, mesmo que outros fatores interfiram nas análises apresentadas (por exemplo, *hardware* envolvido, rotinas intrínsecas do CQC, distorções geométricas, etc.), seus impactos são mantidos constantes durante as simulações, de modo que o desempenho relativo dos algoritmos avaliados podem ser considerados mais gerais.



Figura 5.9 – QR Code submetido aos efeitos destrutivos que agem sobre o canal, determinando variações tonais.

### 5.3.2.1 Desempenho das Técnicas de Classificação

A Figura 5.10<sup>62</sup> ilustra o comparativo entre as taxas de erro de bit da imagem sem correção tonal e empregando a mínima distância Euclidiana como métrica para a classificação. Observe que há uma melhora do desempenho em aproximadamente<sup>63</sup> 1,5 dB, às custas de *overhead* computacional que é “mínimo”<sup>64</sup>.

Já a Figura 5.11 apresenta a curva de desempenho utilizando a mínima distância Mahalanobis. Esta é a métrica que apresenta o melhor desempenho (de aproximadamente 2 dB em relação a imagem sem correção) para agrupamentos de cores que apresentam distorções cujas distribuições obedecem às matrizes de covariância. Todavia, necessita-se conhecer as matrizes de covariância de cada *cluster* de cor. Na simulação ilustrada na Figura 5.11, a estimação das matrizes de covariância é considerada perfeita, pois são exatamente as matrizes utilizadas para produzir os agrupamentos de cada cor.

Do ponto de vista prático, deve-se adotar algum algoritmo de classificação que se aproxime do resultado obtido com a mínima distância Mahalanobis. Em (QUERINI; ITALIANO,

<sup>62</sup> As imagens utilizadas possuem tamanho  $43 \times 43$  pixels (cada pixel corresponde a um módulo) do QR Code. O nível de correção de erro é Q e a versão 5, com três módulos de *quiet zone* em cada lateral do barcode.

<sup>63</sup> Para uma definição mais precisa dos ganhos envolvidos seriam necessários mais um ou dois pontos para BERs menores, a fim de garantir uma derivada constante. No entanto, o elevado tempo das simulações limitou a estatística para o ponto em que já se pode verificar uma derivada constante.

<sup>64</sup> Sabe-se que a mínima distância Euclidiana possui o menor custo computacional quando comparada a outras técnicas de classificação (QUERINI; ITALIANO, 2013). Cabem, todavia, análises que estimem quantitativamente esses valores.

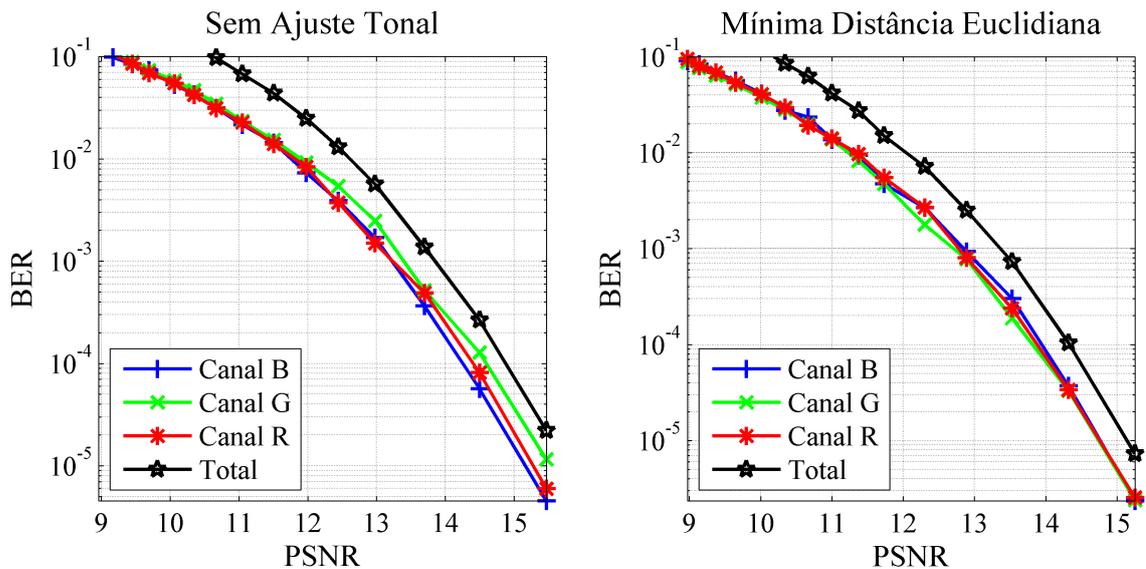


Figura 5.10 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando a mínima distância Euclidiana.

2013) são analisados cinco algoritmos de classificação, a saber: mínima distância Euclidiana, *logistic model tree* (LMT), *k-means*, *Naive Bayes* e *support vector machines* (SVM). Como esperado, o algoritmo de menor complexidade computacional é a mínima distância Euclidiana, seguida do algoritmo *k-means*. Entretanto, o *k-means* apresenta um desempenho superior aos demais em termos de taxa de sucesso<sup>65</sup> de leitura do barcode.

Considerando-se os resultados obtidos em (QUERINI; ITALIANO, 2013), este trabalho opta por analisar o desempenho do algoritmo *k-means* com o intuito de averiguar a robustez do CQC. O algoritmo *k-means* possui como entrada *k* rótulos de cor (oito), previamente conhecidas. Além disso, considera-se como *input* inicial (*seeds*) do algoritmo os valores das oito cores de referência, amostradas na paleta de cores. O resultado comparativo de BER entre o barcode sem a correção tonal e utilizando o *k-means* é apresentado na Figura 5.12. Foram consideradas 100 iterações do *k-means*.

Em termos de BER, os canais R, G e B apresentam desempenhos similares para o modelo de distorção cromática proposto. Contudo, em sensores reais os desempenhos de BER tendem a ser diferentes devido as suas características construtivas focadas na captura de imagens pessoais.

Na Figura 5.13 é apresentado o comparativo dos desempenhos das mínimas distância

<sup>65</sup> Os barcodes foram submetidos a condições reais de impressão (quatro pontos por módulo) e escaneamento (resolução 600 dpi). O tamanho de impressão foi de uma polegada.

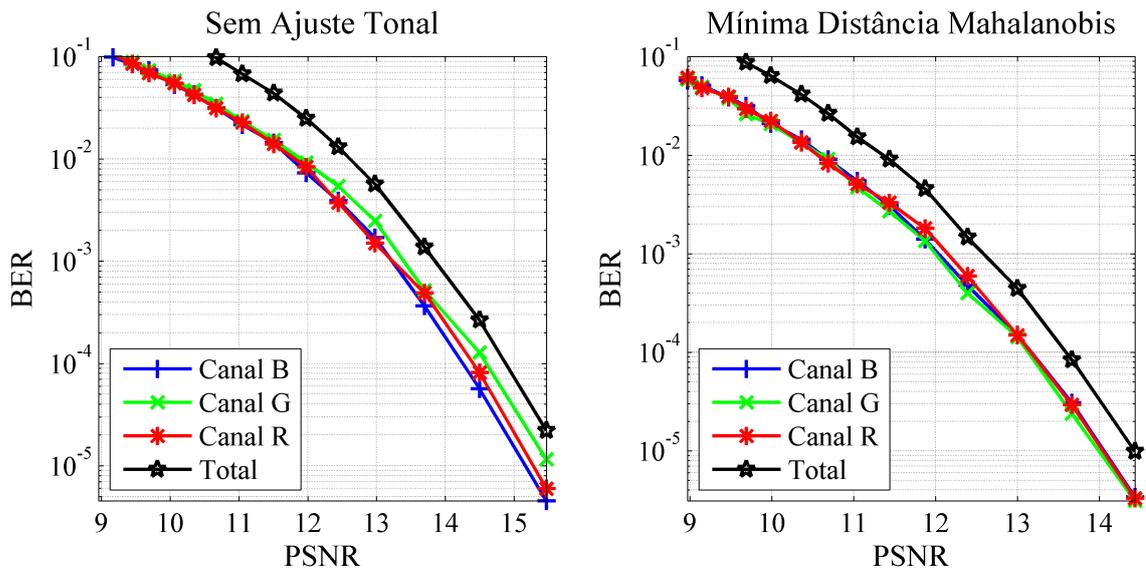


Figura 5.11 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando a mínima distância Mahalanobis.

Euclidiana, *k-means* e Mahalanobis para dois tipos de canais: Gaussiano e com covariância. No canal com covariância é considerado uma variação conjunta e aleatória entre os canais R, G e B, conforme já ilustrado na Figura 5.7. Já no canal Gaussiano, considera-se que a matriz de covariância é diagonal, ou seja, apenas com os termos das variâncias próprias de cada canal cor.

Naturalmente, para o canal Gaussiano, a mínima distância Mahalanobis coincide com a mínima distância Euclidiana. No entanto, o *k-means* apresenta o melhor resultado entre as três técnicas de classificação. Atribui-se esta melhoria ao fato do *k-means* iterativamente atualizar o valor das cores de referência, permitindo a cada iteração melhorar a classificação anterior, quando comparadas às mínimas distância Euclidiana e Mahalanobis, estas que não atualizam os valores das cores de referência.

Já para o canal com covariância, a taxa<sup>66</sup> de erro de bit para o barcode imageado e processado, empregando a mínima distância Euclidiana, apresentou um desempenho aproximado de apenas 0,7 dB inferior à mínima distância Mahalanobis. O *k-means*, por sua vez, melhorou o desempenho da mínima distância Euclidiana em menos de 0,2 dB. Logo, devido a dificuldade em se estimar de maneira fidedigna a matriz de covariância (necessária para a obtenção do desempenho ótimo da mínima distância de Mahalanobis), a mínima distância Euclidiana apresenta um desempenho considerado satisfatório.

<sup>66</sup> Para efeito de correção de erro, considera-se que o *Reed-Solomon* necessita, no mínimo, o dobro de redundância (*redundancy rate*) em relação a BER.

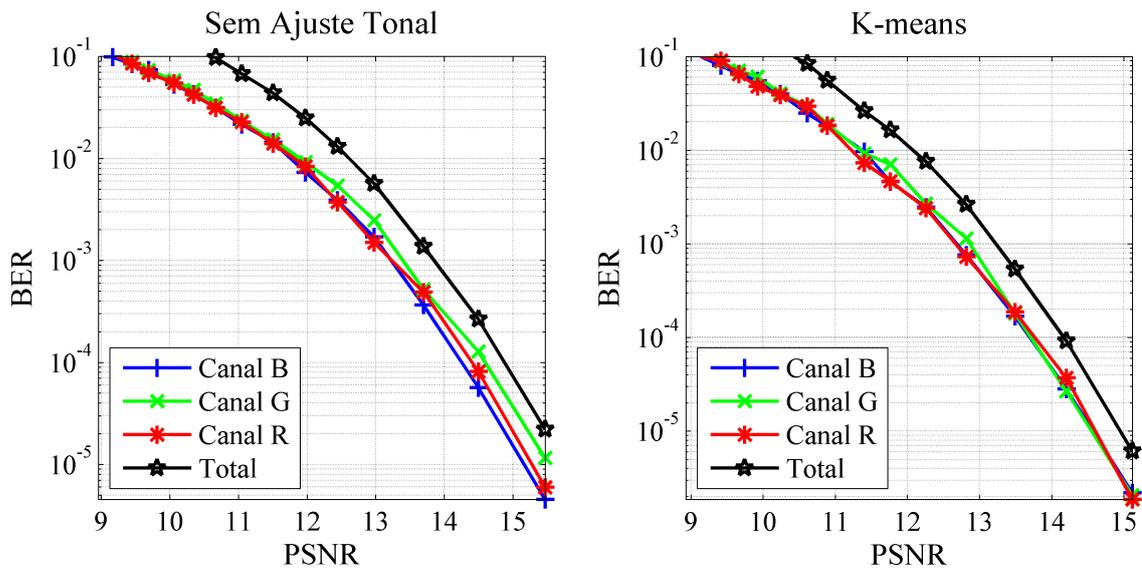


Figura 5.12 – Comparativo entre os desempenhos de BER da imagem sem ajuste de cor e utilizando o algoritmo *k-means*.

O *k-means* melhora o desempenho em relação à mínima distância Euclidiana, pois ele processa iterativamente a minimização das distâncias Euclidianas entre cada uma das  $n$  cores capturadas e a média de cada *cluster* (a cada iteração a média se aproxima do centroide do *cluster*, caso o algoritmo esteja convergindo). Por sua vez, a mínima distância Euclidiana, calcula a diferença de cada uma das  $n$  cores capturadas *sempre* em relação às cores de referência que podem estar afastadas do centroide do *cluster*. Uma vez que as cores de referência são observações presentes dentro de cada *cluster* de cor, com poucas<sup>67</sup> iterações o *k-means* apresenta um desempenho melhor que a mínima distância Euclidiana.

### 5.3.2.2 Desempenho do Aumento da Resolução

Um outro fator que afeta o desempenho de BER é resolução das imagens. Na Figura 5.14 é realizado um comparativo entre as taxas de erro de bit das imagens com resolução  $43 \times 43$  (um pixel por módulo) e  $86 \times 86$  (quatro pixels por módulo). Verificou-se uma melhoria no desempenho de aproximadamente 0,25 dB para a curva sem correção tonal. Atribui-se esta melhoria ao efeito positivo que as resoluções maiores possuem na determinação dos níveis de luminância máximo e mínimo na decodificação do QR Code. Se as distorções que os pixels de um módulo sofrem forem mais próximas<sup>68</sup> (na prática os pixels constituintes de um módulo

<sup>67</sup> Um estudo nesse sentido pode minimizar o número de iterações do *k-means*, reduzindo o custo computacional.

<sup>68</sup> No modelo desenvolvido cada pixel sofre um distorção definida pela matriz de covariância.

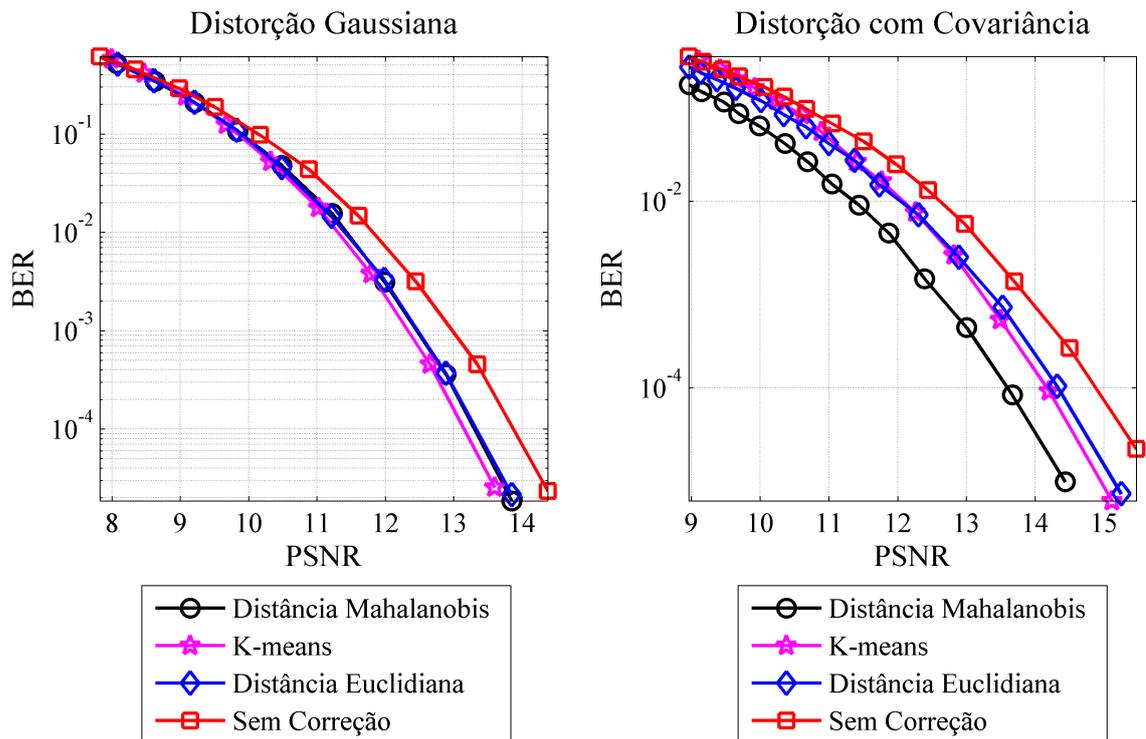


Figura 5.13 – Comparativo entre as técnicas de calibração de cor: sob distorção Gaussiana (gráfico à esquerda) e com covariância entre os canais R, G e B (gráfico à direita).

possuem distorções semelhantes) os resultados podem ser ainda melhores.

Para a resolução de quatro pixels por módulo do barcode, as cores de referência são calculadas a partir das médias<sup>69</sup> dos quatro pixels que compõem os módulos de cada cor. Este valor médio permite atingir um desempenho ainda superior ao verificado no barcode sem correção tonal para uma imagem de mesmo tamanho. Desta forma, nas condições apresentadas, a melhoria no desempenho da imagem com o dobro de resolução é de pouco mais de meio<sup>70</sup> Decibel quando comparada à imagem com resolução de um pixel por módulo. Este fato revela uma tendência de afastamento<sup>71</sup> (no eixo das abcissas) entre as curvas que utilizam as cores de referência e a curva sem correção tonal.

À medida que a resolução da imagem aumenta, a melhoria de desempenho atribuída ao *k-means*, em relação à mínima distância Euclidiana, vai se extinguindo. Esse fenômeno ocorre, pois a utilização de mais pixels por módulo permite que as cores de referência, calculadas pelas

<sup>69</sup> Uma outra abordagem consiste em amostrar o pixel central do módulo, minimizando os efeitos de *aliasing*, especialmente para imagens com alta resolução (DOERMANN, 2008). O fenômeno *aliasing* ocorre devido a correlação entre pixels de módulos vizinhos.

<sup>70</sup> Tanto para a mínima distância Euclidiana e Mahalanobis.

<sup>71</sup> Resoluções superiores a 86 × 86 pixels permitiriam uma análise mais precisa, bem como a possibilidade de saturação deste efeito.

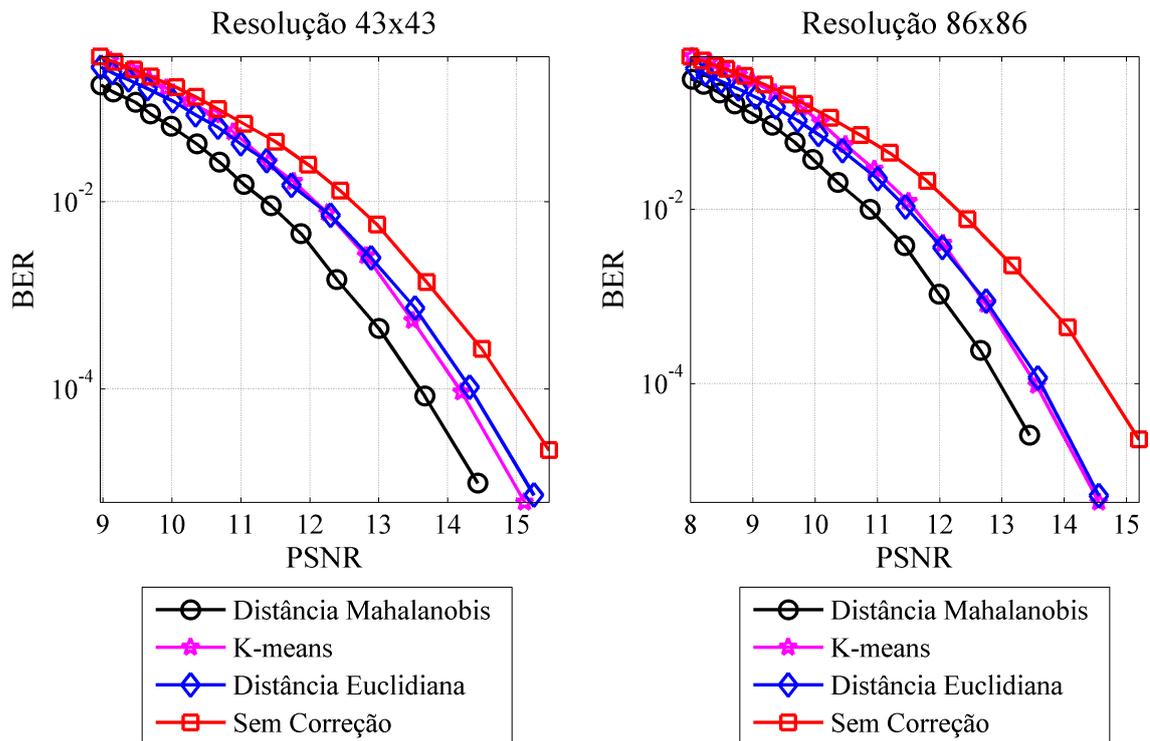


Figura 5.14 – BER para as resoluções de  $43 \times 43$  e  $86 \times 86$  pixels, utilizando códigos de barra sem correção tonal e os algoritmos de classificação *k-means*, mínima distância Euclidiana e Mahalanobis.

média desses pixels, aproximem o valor da cor de referência do centroide do agrupamento de cor, “emulando”, de certo modo, uma iteração do *k-means*.

### 5.3.3 Análise da FRR

A análise do desempenho geral de leitura do CQC pode ser realizada pela *first read rates* (FRR<sup>72</sup>) que avalia percentualmente a taxa de sucessos de leitura, em relação ao total de leituras. Matematicamente é expressa por (KATO, 2010):

$$\text{FRR} = \frac{\text{Número de Sucessos de Leitura}}{\text{Número de Tentativas de Leitura}}. \quad (5.5)$$

As Figuras 5.15 e 5.16 ilustram o desempenho da FRR para as distorções cromáticas mensuradas em função da PSNR. Naturalmente, a FRR aumenta com o incremento da PSNR. Além disso, a FRR cresce na medida que a técnica empregada se torna mais apurada apresentando, portanto, melhores resultados para a mínima distância Mahalanobis. Esse resultado é o reflexo do desempenho já verificado de BER, pois quanto menor a BER para um dado valor de

<sup>72</sup> A FRR é uma métrica que generaliza a SSR (já referida em Seção 5.2).

PSNR maiores são as chances do barcode ser sincronizado e decodificado.

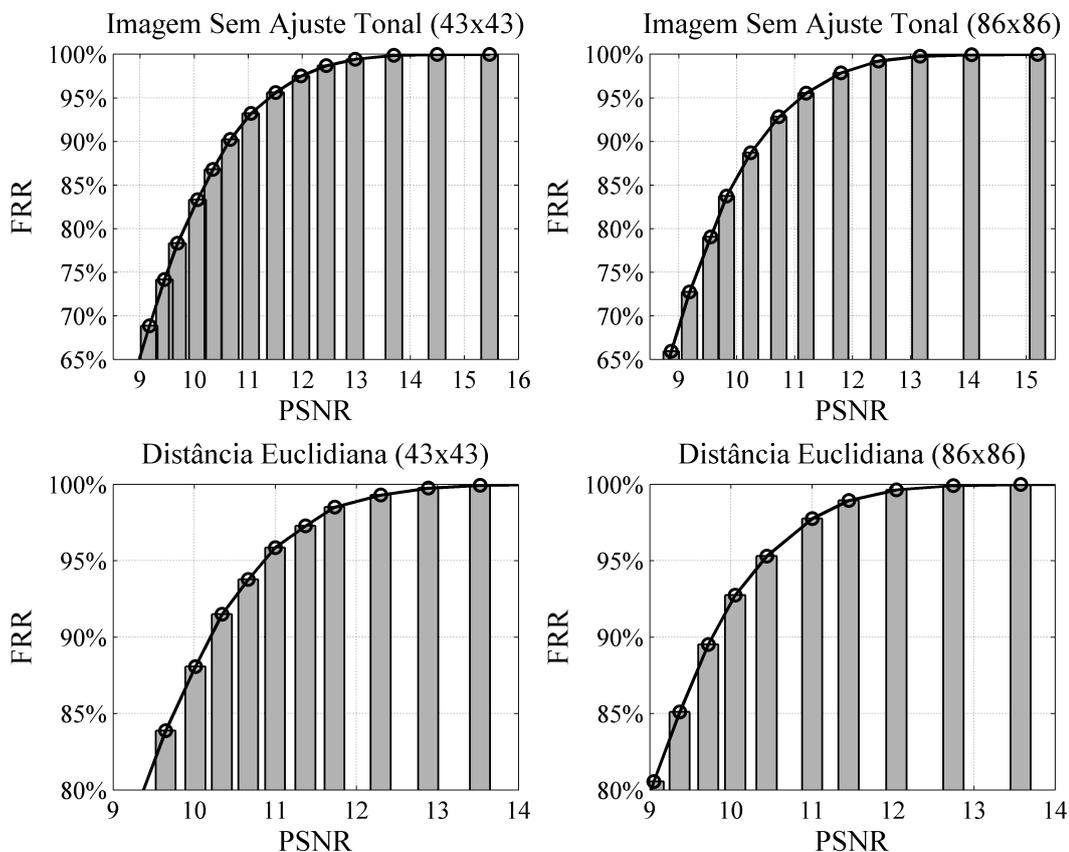


Figura 5.15 – FRR para as resoluções de  $43 \times 43$  e  $86 \times 86$  pixels, utilizando as imagens sem correção tonal e corrigida, empregando a mínima distância Euclidiana.

Adicionalmente, nestas figuras estão dispostos lado a lado as FRRs para as resoluções de imagem  $43 \times 43$  e  $86 \times 86$ . A partir da inserção de mais pixels na imagem há uma melhoria do desempenho da FRR, pois o decodificador se torna mais capaz de fazer a sincronização do barcode através do contraste de luminâncias de cada canal monocromático.

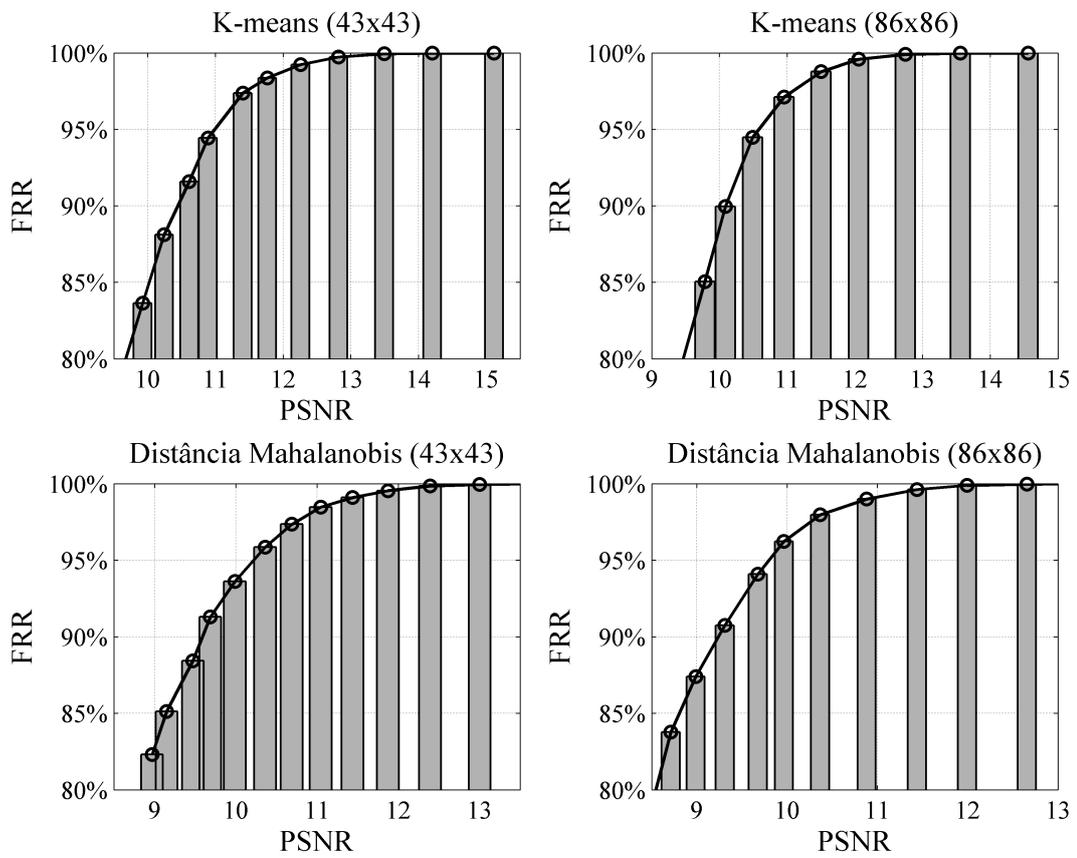


Figura 5.16 – FRR para as resoluções de  $43 \times 43$  e  $86 \times 86$  pixels, utilizando as imagens corrigidas, empregando o algoritmo *k-means* e a mínima distância Mahalanobis.

## 6 CONCLUSÃO

O *framework* desenvolvido neste trabalho contemplou a implementação do codificador do padrão QR Code, de modelos que simulam as interferências e distorções que atuam sobre o canal de imagem e a proposta de um novo barcode colorido, compatível com o padrão QR Code. Com esta proposta, buscou-se aumentar a taxa de dados e a imunidade às variações cromáticas, mantendo a mesma robustez do QR Code. Além disso, o CQC possui características bastantes atrativas que merecem serem destacadas: i) compatibilidade com o padrão QR code, permitindo a reutilização de melhorias e aprimoramentos já propostas para o barcode monocromático; ii) utilização de oito cores que possibilitam triplicar a capacidade de armazenamento em relação ao QR Code monocromático; iii) o canal verde pode ser “lido” diretamente no barcode; iv) possui capacidade de correção tonal.

As paletas de cores, que foram posicionadas no contorno do barcode proposto, permitem realizar a correção tonal e, ao mesmo tempo, buscar uma melhor rejeição aos efeitos causados pela iluminação. Um modelo emulador genérico de distorção cromática, baseado em matrizes de covariâncias, foi elaborado com o intuito de avaliar o desempenho das cores de referência na correção tonal das distorções produzidas. A principal métrica utilizada foi a taxa de erro de bit. A partir dela, observou-se que a mínima distância Euclidiana apresenta um desempenho satisfatório a apenas 0,5 dB do desempenho ótimo da mínima distância de Mahalanobis.

A utilização da métrica de Mahalanobis teve um desempenho considerado ótimo para o modelo de canal proposto neste trabalho, pois considerou a estimação perfeita da matriz de covariância. No entanto, devido a dificuldade em se estimar a matriz de covariância por meio de alguma técnica de classificação das cores, pode-se considerar que mínima distância Euclidiana obteve um resultado satisfatório, aditado ao seu baixo custo computacional. Para futuras implementações, pode-se adotar a métrica Mahalanobis como referência de desempenho a ser superado.

Nas simulações empregando as mínimas distâncias Euclidiana e Mahalanobis, os resultados de BER foram comparativamente melhores do que na imagem de mesmo tamanho, mas sem correção tonal. Além disso, os testes realizados indicam que imagens com resoluções maiores determinam desempenhos superiores em termos de BER e FRR. Essa melhoria é atribuída às cores de referência que passam a considerar mais pixels no cálculo da média que determina a cor de referência. Devido a este resultado, pode-se estimar que o afastamento, em dB, entre

as curvas das mínimas distâncias Euclidiana e Mahalanobis tenda a aumentar.

Os resultados obtidos mostram que a taxa de redundância requerida para uma determinada técnica de classificação de cores está atrelada a taxa de erro de bit. No caso do *Reed-Solomon*, com capacidade de correção igual a 50% do número de blocos de redundância, a taxa de redundância mínima será o dobro da BER. Desta forma, quanto mais eficaz for a técnica de classificação, menos taxa de redundância será necessária e, portanto, maior será a *goodput* do *payload*.

A imunidade do QR Code a diferentes tipos de interferências que atuam sobre o canal de imagem foi avaliada, delimitando os limites de decodificação do padrão. Observou-se maior robustez do QR Code ao ruído impulsivo, em detrimento às distorções radiais. A métrica de avaliação considerou o sucesso, ou o fracasso na sincronização do QR Code. Como a distorção do tipo sal e pimenta cria pontos de inversão de bit na imagem, e as demais distorções alteram efetivamente a geometria da imagem, pode-se afirmar que o QR Code é menos robusto às distorções geométricas.

## 6.1 Possibilidades de Investigações Futuras

No desenvolvimento deste trabalho surgiram algumas importantes possibilidades de investigação que complementam o atual trabalho. São elas:

- Implementação em Matlab do decodificador padrão do QR Code (permitirá uma melhor gestão sobre as análises);
- Análise do *overhead* computacional inserido pelas cores de referência;
- Simulação dos efeitos de iluminação desigual e de baixo contraste;
- Determinação da BER para imagens com resoluções maiores, considerando duas abordagens para as cores de referência: amostragem do pixel central do módulo, ou média dos pixels do módulo;
- Análise da redução de robustez em função da diminuição de luminância da cor magenta no CQC para permitir a leitura do canal verde, mesmo com decodificador padrão baseado na diferença de luminâncias;
- Validação da proposta em um canal de imagem real;

- Implementação de outros algoritmos de classificação, buscando o melhor *trade-off* entre custo computacional e ganho de desempenho;
- Alteração da métrica BER para ByER (*byte error rate*), pois os barcodes 2D utilizam códigos corretores de erro de bloco (tal como o *Reed-Solomon*), em vez de códigos corretores de bits;
- Análise da perda de desempenho do CQC em função da escolha das melhores máscaras para os barcodes R, G e B (segundo o decodificador do QR Code), e a escolha aleatória das máscaras;
- Avaliação da compressão JPEG (formato usualmente utilizado para salvar as imagens digitais), já que esta afeta a exatidão das imagens reproduzidas;
- Derivação analítica de uma expressão de BER para o modelo de distorções cromáticas;
- Determinação da taxa de dados em função da distorção perspectiva.

Outras possibilidades mais ousadas tratam da utilização de barcodes coloridos animados para a transferência de arquivos por imagem, e a proposição de uma nova geometria de barcodes.

## REFERÊNCIAS

- ABOWD, G. D. et al. Towards a Better Understanding of Context and Context-Awareness. In: HUC '99 PROCEEDINGS OF THE 1ST INTERNATIONAL SYMPOSIUM ON HANDHELD AND UBIQUITOUS COMPUTING, London, UK. **Anais...** Springer Berlin Heidelberg, 1999. p.304–307.
- ALFTHAN, J. **Robust Detection of Two-Dimensional Barcodes in Blurry Images**. 73p. Dissertação (Mestrado) — Royal Institute of Technology. 2008.
- ASHOK, A. et al. Capacity of pervasive camera based communication under perspective distortions. In: IEEE INTERNATIONAL CONFERENCE ON PERVASIVE COMPUTING AND COMMUNICATIONS (PERCOM), 2014. **Anais...** IEEE, 2014. p.112–120.
- BELUSSI, L.; HIRATA, N. Fast QR Code Detection in Arbitrarily Acquired Images. In: SIBGRAPI CONFERENCE ON GRAPHICS, PATTERNS AND IMAGES, 2011., Maceio, Alagoas. **Anais...** IEEE, 2011. p.281–288.
- BLAHUT, R. E. **Algebraic Codes for Data Transmission**. 1<sup>a</sup>.ed. [S.l.]: Cambridge University Press, 2003. 498p.
- BLASINSKI, H.; BULAN, O.; SHARMA, G. Per-colorant-channel color barcodes for mobile applications: an interference cancellation framework. **IEEE transactions on image processing : a publication of the IEEE Signal Processing Society**, [S.l.], v.22, n.4, p.1498–1511, Apr. 2013.
- BRIGGS, D. **The Dimensions of Colour**. [Online; acessado em 13/01/2015], <http://www.huevaluechroma.com/012.php>.
- BRYANT, L. et al. **Emerging technologies for learning (volume 2)**. [S.l.]: British Educational Communications and Technology Agency (BECTA), 2007., 2007. 64–79p.
- BULAN, O.; BLASINSKI, H.; SHARMA, G. Color QR Codes: increased capacity via per-channel data encoding and interference cancellation. In: COLOR AND IMAGING CONFERENCE FINAL PROGRAM AND PROCEEDINGS, 19., San Jose, California. **Anais...** Society for Imaging Science and Technology, 2011. p.156–159.

BULAN, O.; MONGA, V.; SHARMA, G. High Capacity Color Barcodes Using Dot Orientation and Color Separability. In: SPIE ELECTRONIC IMAGING SYMPOSIUM, San Jose, CA, USA. **Anais...** [S.l.: s.n.], 2009. v.7254, p.725417–725417–7.

BULAN, O.; OZTAN, B.; SHARMA, G. High capacity image barcodes using color separability. In: SPIE 7866, COLOR IMAGING XVI: DISPLAYING, PROCESSING, HARDCOPY, AND APPLICATIONS. **Proceedings...** [S.l.: s.n.], 2011. v.7866, p.78660N–78660N–9.

BULAN, O.; SHARMA, G. High capacity color barcodes: per channel data encoding via orientation modulation in elliptical dot arrays. **IEEE transactions on image processing : a publication of the IEEE Signal Processing Society**, [S.l.], v.20, n.5, p.1337–50, May 2011.

BULAN, O.; SHARMA, G.; MONGA, V. Orientation modulation for data hiding in clustered-dot halftone prints. **IEEE transactions on image processing: a publication of the IEEE Signal Processing Society**, [S.l.], v.19, n.8, p.2070–84, Aug. 2010.

CHANG, J. H. An introduction to using QR codes in scholarly journals. **Science Editing**, [S.l.], v.1, n.2, p.113–117, Aug. 2014.

CHANG, Y. C.; REID, J. F. RGB calibration for color image analysis in machine vision. **IEEE transactions on image processing: a publication of the IEEE Signal Processing Society**, [S.l.], v.5, n.10, p.1414–22, Jan. 1996.

CHANG, Y.-H.; CHU, C.-H.; CHEN, M.-S. A General Scheme for Extracting QR Code from a Non-uniform Background in Camera Phones and Applications. In: NINTH IEEE INTERNATIONAL SYMPOSIUM ON MULTIMEDIA (ISM 2007). **Anais...** IEEE, 2007. p.123–130.

CHUNG, C.-H.; CHEN, W.-Y.; TU, C.-M. Image Hidden Technique Using QR-Barcode. **2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing**, [S.l.], p.522–525, Sept. 2009.

COMMUNICATIONS, A. **Specifications For Popular 2D Bar Codes**. [Online; acessado em 03/01/2015], <http://www.adams1.com/stack.html>.

DEAN, T.; DUNN, C. **Quick layered response (qlr) codes**. [S.l.: s.n.], 2012.

DOERMANN, D. VCode—Pervasive Data Transfer Using Video Barcode. **IEEE Transactions on Multimedia**, [S.l.], v.10, n.3, p.361–371, Apr. 2008.

EMARKETER. **Smartphone Users Worldwide Will Total 1.75 Billion in 2014.** [Online; acessado em 17/10/2014], <http://www.emarketer.com/Article/Smartphone-Users-Worldwide-Will-Total-175-Billion-2014/1010536>.

FALAS, T.; KASHANI, H. Two-Dimensional Bar-Code Decoding with Camera-Equipped Mobile Phones. In: FIFTH ANNUAL IEEE INTERNATIONAL CONFERENCE ON PERVASIVE COMPUTING AND COMMUNICATIONS WORKSHOPS (PERCOMW'07). **Anais...** IEEE, 2007. p.597–600.

GAO, J. Z.; PRAKASH, L.; JAGATESAN, R. Understanding 2D-BarCode Technology and Applications in M-Commerce - Design and Implementation of A 2D Barcode Processing Solution. In: ANNUAL INTERNATIONAL COMPUTER SOFTWARE AND APPLICATIONS CONFERENCE - VOL. 2 - (COMPSAC 2007), 31. **Anais...** IEEE, 2007. v.2, p.49–56.

GARATEGUY, G. J. et al. QR images: optimized image embedding in qr codes. **IEEE transactions on image processing: a publication of the IEEE Signal Processing Society**, [S.l.], v.23, n.7, p.2842–53, July 2014.

GONZALEZ, R. C.; WOODS, R. E. **Processamento Digital de Imagens**. 3<sup>a</sup>.ed. São Paulo: Pearson Prentice Hall., 2010.

GRILLO, A. et al. High Capacity Colored Two Dimensional codes. **Proceedings of the International Multiconference on Computer Science and Information Technology**, [S.l.], p.709–716, Oct. 2010.

HUANG, H.-C.; CHANG, F.-C.; FANG, W.-C. Reversible data hiding with histogram-based difference expansion for QR code applications. **IEEE Transactions on Consumer Electronics**, [S.l.], v.57, n.2, p.779–787, May 2011.

ISO. **ISO/IEC 18004:2006 - information technology – automatic identification and data capture techniques – qr code 2005 bar code symbology specification**. 2006. 114p.

ISO. **ISO/IEC 15438:2006 - information technology – automatic identification and data capture techniques – pdf417 bar code symbology specification**. [S.l.]: International Organization for Standardization, 2006., 2006. 102p.

ISO. **ISO/IEC 15420:2009 information technology – automatic identification and data capture techniques – ean/upc bar code symbology specification**. [S.l.: s.n.], 2013.

K. T. Gribbon, C. T. Johnston, D. G. B. A real-time FPGA implementation of a barrel distortion correction algorithm with bilinear interpolation. In: IN PROCEEDINGS OF THE IMAGE AND VISION COMPUTING NEW ZEALAND, New Zealand. **Anais...** [S.l.: s.n.], 2003. p.408–413.

KAO, W.-C.; CHIU, C.-H.; YANG, Y.-Y. Automatic ringing artifact detection in restoring blurred face images. In: IEEE INTERNATIONAL MIDWEST SYMPOSIUM ON CIRCUITS AND SYSTEMS, 2009. **Anais...** IEEE, 2009. p.770–773.

KATO, H. Performance of a color 2D barcode as a pervasive computing tool. In: ISPACS 2010 - 2010 INTERNATIONAL SYMPOSIUM ON INTELLIGENT SIGNAL PROCESSING AND COMMUNICATION SYSTEMS, PROCEEDINGS. **Anais...** IEEE, 2010. p.1–4.

KATO, H.; TAN, K. T. Pervasive 2D Barcodes for Camera Phone Applications. **IEEE Pervasive Computing**, [S.l.], v.6, n.4, p.76–85, Oct. 2007.

KATO, H.; TAN, K. T.; CHAI, D. Novel colour selection scheme for 2D barcode. In: INTERNATIONAL SYMPOSIUM ON INTELLIGENT SIGNAL PROCESSING AND COMMUNICATION SYSTEMS (ISPACS), 2009. **Anais...** IEEE, 2009. p.529–532.

KATO, H.; TAN, K. T.; CHAI, D. **Barcodes for Mobile Devices**. United States: Cambridge University Press., 2010.

KIKUCHI, M.; FUJIYOSHI, M.; KIYA, H. A new color QR code forward compatible with the standard QR code decoder. **2013 International Symposium on Intelligent Signal Processing and Communication Systems**, [S.l.], p.26–31, Nov. 2013.

LANGLOTZ, T.; BIMBER, O. Unsynchronized 4D Barcodes: coding and decoding time-multiplexed 2d colorcodes. In: INTERNATIONAL CONFERENCE ON ADVANCES IN VISUAL COMPUTING - VOLUME PART I, 3., Berlin, Heidelberg. **Proceedings...** Springer-Verlag, 2007. p.363–374. (ISVC'07).

LIN, J.-A.; FUH, C.-S. 2D Barcode Image Decoding. In: MATHEMATICAL PROBLEMS IN ENGINEERING. **Anais...** Hindawi Publishing Corporation, 2013. p.10.

LIU, N. et al. Two-dimensional bar code out-of-focus deblurring via the Increment Constrained Least Squares filter. **Pattern Recognition Letters**, [S.l.], v.34, n.2, p.124–130, Jan. 2013.

LIU, X.; DOERMANN, D.; LI, H. A camera-based mobile data channel: capacity and analysis. In: PROCEEDING OF THE 16TH ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA - MM '08, New York, New York, USA. **Anais...** ACM Press, 2008. p.359–368.

LIU, Y.; LIU, M. Automatic Recognition Algorithm of Quick Response Code Based on Embedded System. In: SIXTH INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS DESIGN AND APPLICATIONS, Jinan. **Anais...** IEEE, 2006. v.2, p.783–788.

MELGAR, M. E. V. **Geração e Leitura de QR Codes Coloridos**. 94p. Dissertação (Mestrado) — Universidade de Brasília. 2013.

MEMETI, J. **Data Transfer Using a Camera and a Three-Dimensional Code**. 127p. Tese (Doutorado) — University of Zurich. 2012.

MENG, J.; YANG, Y. Application of Mobile 2D Barcode in China. In: INTERNATIONAL CONFERENCE ON WIRELESS COMMUNICATIONS, NETWORKING AND MOBILE COMPUTING, 2008. **Anais...** IEEE, 2008. p.1–4.

MICROSOFT. **High Capacity Color Barcode Technology - Microsoft Research**. [Online; acessado em 04/02/2015], <http://research.microsoft.com/en-us/projects/hccb/about.aspx>.

MOGHADDAM, M. E. A Mathematical Model to Estimate Out of Focus Blur. In: INTERNATIONAL SYMPOSIUM ON IMAGE AND SIGNAL PROCESSING AND ANALYSIS, 2007., Istanbul. **Anais...** IEEE, 2007. p.278–281.

NURWONO, K. A. H.; KOSALA, R. Color quick response code for mobile content distribution. In: INTERNATIONAL CONFERENCE ON ADVANCES IN MOBILE COMPUTING AND MULTIMEDIA - MOMM '09, 7., New York, New York, USA. **Proceedings...** ACM Press, 2009. p.267.

OHBUCHI, E.; HANAIZUMI, H. Barcode Readers using the Camera Device in Mobile Phones. In: INTERNATIONAL CONFERENCE ON CYBERWORLDS, 2004. **Anais...** IEEE, 2004. p.260–265.

ONG, S. K.; CHAI, D.; RASSAU, A. The use of alignment cells in MMCC barcode. In: INTERNATIONAL SYMPOSIUM ON COMMUNICATIONS AND INFORMATION TECHNOLOGIES, 2010., Tokyo. **Anais...** IEEE, 2010. p.640–645.

ONG, S. K.; CHAI, D.; RASSAU, A. Adaptive use of thresholding and multiple colour space representation to improve classification of MMCC barcode. In: MULTIMEDIA AND EXPO (ICME), 2011 IEEE INTERNATIONAL CONFERENCE ON, Barcelona, Spain. **Anais...** IEEE, 2011. p.1–6.

ONG, S. K.; CHAI, D.; TAN, K. T. The Use of Border in Colour 2D Barcode. In: IEEE INTERNATIONAL SYMPOSIUM ON PARALLEL AND DISTRIBUTED PROCESSING WITH APPLICATIONS, 2008. **Anais...** IEEE, 2008. p.999–1005.

ONG, S. K.; CHAI, D.; TAN, K. T. Performance of mobile camera on colour 2D barcode. In: INTERNATIONAL SYMPOSIUM ON INTELLIGENT SIGNAL PROCESSING AND COMMUNICATION SYSTEMS (ISPACS), 2009., Kanazawa, Japan. **Anais...** IEEE, 2009. p.497–500.

ONODA, T.; MIWA, K. **Layered two-dimensional code, creation method thereof, and read method.** [S.l.]: Google Patents, 2009., 2009. n.US 20090166418.

OpenStax College. **Sensory Perception.** [S.l.]: OpenStax-CNX, 2013.

PARIKH, D.; JANCKE, G. Localization and Segmentation of A 2D High Capacity Color Barcode. In: IEEE WORKSHOP ON APPLICATIONS OF COMPUTER VISION, 2008. **Anais...** IEEE, 2008. p.1–6.

QUERINI, M. et al. 2D Color Barcodes for Mobile Phones. In: INTERNATIONAL JOURNAL OF COMPUTER SCIENCE AND APPLICATIONS (IJCSA). **Anais...** [S.l.: s.n.], 2011. p.136–155.

QUERINI, M.; ITALIANO, G. Color classifiers for 2D color barcodes. In: COMPUTER SCIENCE AND INFORMATION SYSTEMS (FEDCSIS), 2013 FEDERATED CONFERENCE ON, Kraków, Poland. **Anais...** IEEE, 2013. p.611–618.

RAMYA, M.; JAYASHEELA, M. Improved Color QR Codes for Real Time Applications with High Embedding Capacity. **International Journal of Computer Applications**, [S.l.], v.91, n.8, p.8–12, 2014.

MURAKAMI, H. et al. (Ed.). **Ubiquitous Computing Systems.** Berlin, Heidelberg: Springer Berlin Heidelberg., 2005. 74–89p. (Lecture Notes in Computer Science, v.3598).

RUNGRAUNGSILP, S.; KETCHAM, M. Data hiding method for QR code based on watermark by compare DCT with DFT domain. **2012 International Conference on Computer and Communication Technologies**, [S.l.], p.154–158, 2012.

RUSS, J. C. **The Image Processing Handbook, Sixth Edition**. Sixth Edit.ed. [S.l.]: CRC Press, 2011.

SAHA, D.; MUKHERJEE, A. Pervasive computing: a paradigm for the 21st century. **Computer**, [S.l.], v.36, n.3, p.25–31, Mar. 2003.

SCRATCHAPIXEL. **Lesson 5: colors and digital images**. [Online; acessado em 12/12/2014], <http://www.scratchapixel.com/old/lessons/3d-basic-lessons/lesson-5-colors-and-digital-images/light/>.

SHARMA, G.; TRUSSELL, H. J. Digital color imaging. **IEEE transactions on image processing: a publication of the IEEE Signal Processing Society**, [S.l.], v.6, n.7, p.901–32, Jan. 1997.

SHIRLEY, P.; MARSCHNER, S. **Fundamentals of Computer Graphics**. 3 Edition.ed. [S.l.]: A K Peters/CRC Press, 2009.

SOLOMON, C.; BRECKON, T. **Fundamentals of Digital Image Processing: a practical approach with examples in matlab**. Integral.ed. [S.l.]: John Wiley e Sons, 2011. 344p.

SOON, T. J. QR Code. **Synthesis Journal**, [S.l.], p.59–78, 2008.

SöRÖS, G.; FLÖRKEMEIER, C. Blur-resistant joint 1D and 2D barcode localization for smartphones. In: INTERNATIONAL CONFERENCE ON MOBILE AND UBIQUITOUS MULTIMEDIA - MUM '13, 12., New York, New York, USA. **Proceedings...** ACM Press, 2013. p.1–8.

TACK-DON, H. et al. **Machine readable code image and method of encoding and decoding the same**. [S.l.]: Google Patents, 2006., 2006. n.US 7,020,327.

TAN, K. T. et al. **Data Storage Device and Encoding/Decoding Method**. 2008. n.WO 2010031110.

TAPPAN, J. H.; WRIGHT, M. E.; SISTLER, F. E. Error sources in a digital image analysis system. **Computers and Electronics in Agriculture**, [S.l.], v.2, n.2, p.109–118, Oct. 1987.

TIWARI, S. et al. A Blind Blur Detection Scheme Using Statistical Features of Phase Congruency and Gradient Magnitude. **Advances in Electrical Engineering**, [S.l.], v.2014, p.1–10, 2014.

TIWARI, S. et al. Blur Classification using Ridgelet Transform and Feed Forward Neural Network. **International Journal of Image, Graphics and Signal Processing (IJIGSP)**, [S.l.], v.6, n.9, p.47, Aug. 2014.

TIWARI, S.; SHUKLA, V. P.; SINGH, A. K. Review of motion blur estimation techniques. **Journal of Image and Graphics**, [S.l.], v.1, n.4, p.176–184, 2013.

TSAI, R. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. **IEEE Journal on Robotics and Automation**, [S.l.], v.3, n.4, p.323–344, Aug. 1987.

VASS, G.; PERLAKI, T. Applying and removing lens distortion in post production. In: HUNGARIAN CONFERENCE ON COMPUTER GRAPHICS AND GEOMETRY 2003, 2., Budapest, Hungary. **Anais...** [S.l.: s.n.], 2003. p.8.

Vizcarra Melgar, M. E. et al. CQR codes: colored quick-response codes. In: IEEE SECOND INTERNATIONAL CONFERENCE ON CONSUMER ELECTRONICS - BERLIN (ICCE-BERLIN), 2012. **Anais...** IEEE, 2012. p.321–325.

WANG, J.-W. Nested image steganography scheme using QR-barcode technique. **Optical Engineering**, [S.l.], v.48, n.5, p.057004, May 2009.

WEISER, M. The computer for the 21 st century. **ACM SIGMOBILE Mobile Computing and Communications Review**, [S.l.], v.3, n.3, p.3–11, July 1999.

XU, G. et al. Identification and Recovery of the Blurred QR Code Image. In: INTERNATIONAL CONFERENCE ON COMPUTER SCIENCE AND SERVICE SYSTEM, 2012., Nanjing, China. **Anais...** IEEE, 2012. p.2257–2260.

YEH, Y.-L.; YOU, J.-C.; JONG, G.-J. The 2D Bar-Code Technology Applications in Medical Information Management. In: EIGHTH INTERNATIONAL CONFERENCE ON INTELLIGENT SYSTEMS DESIGN AND APPLICATIONS, 2008. **Anais...** IEEE, 2008. v.3, p.484–487.

Yue Liu; Ju Yang; Mingjun Liu. Recognition of QR Code with mobile phones. In: CHINESE CONTROL AND DECISION CONFERENCE, 2008., Yantai, Shandong. **Anais...** IEEE, 2008. p.203–206.

Yunhua Gu; Weixiang Zhang. QR code recognition based on image processing. In: INTERNATIONAL CONFERENCE ON INFORMATION SCIENCE AND TECHNOLOGY, Nanjing. **Anais...** IEEE, 2011. p.733–736.

# APÊNDICES

---

## APÊNDICE A – Matriz de Variação Tonal

A técnica de calibração de cor, baseada na matriz de variação tonal, ancora-se no fato de que qualquer cor RGB pode ser reproduzida a partir de combinações lineares independentes dos valores R, G e B. Por causa dessa linearidade é possível utilizar um filtro  $T$  que permita realizar o balanceamento de cores, convertendo a cor capturada para as cores primárias vermelha, verde e azul, da seguinte forma:

$$\begin{bmatrix} R_{cap} & G_{cap} & B_{cap} \end{bmatrix} \times T = \begin{bmatrix} R & G & B \end{bmatrix}, \quad (\text{A.1})$$

em que  $R_{cap}$ ,  $G_{cap}$  e  $B_{cap}$  são as componentes de cor R, G e B amostradas na detecção, e  $R$ ,  $G$  e  $B$  são as cores primárias saturadas (ciano, magenta, amarela, preta, branca, vermelha, verde e azul). A matriz  $T$  na equação acima realiza a transformação linear que conduz de volta a cor capturada no barcode para as cores puras R, G e B produzidas pela sobreposição de camadas na Figura 5.2.

A adoção da Equação A.1 possibilita a utilização de blocos (de cores) pilotos, que nada mais são do que regiões nas quais as cores embarcadas são conhecidas tanto pelo codificador quanto pelo decodificador. Sendo assim, as cores  $R_{cap}$ ,  $G_{cap}$  e  $B_{cap}$  devem ser equiparadas às suas triplas formadoras  $R$ ,  $G$  e  $B$ , admitindo-se que a variação tonal percebida nos blocos piloto seja a mesma do que a sentida na região de codificação. A deterioração concebida no canal de imagem, em função de diversos fatores já elucidados, está reunida na matriz de transformação  $T$  que, na prática, é desconhecida.

A obtenção da matriz  $T$  é realizada através dos blocos pilotos. Estes foram posicionados nos módulos de localização do QR Code, cada um dos três quadrados concêntricos com uma das cores azul, vermelho e amarelo. A escolha do amarelo em detrimento ao verde no quadrado central se deve a sua luminância muito mais próximo ao branco. A Figura A.1 ilustra essas escolhas.

A distribuição das cores de referência deve ser realizada de tal forma que permite amostar diferentes sensações tonais ao longo de todo o barcode. Portanto, o posicionamento nos cantos do barcode é uma boa escolha. Estudos futuros podem indicar, também, a utilização dos módulos de alinhamento para inserção de cores de referência, com o intuito de obter uma melhor diversidade espacial.

Expandindo-se a Equação A.1, em função das considerações atribuídas à Figura A.1,

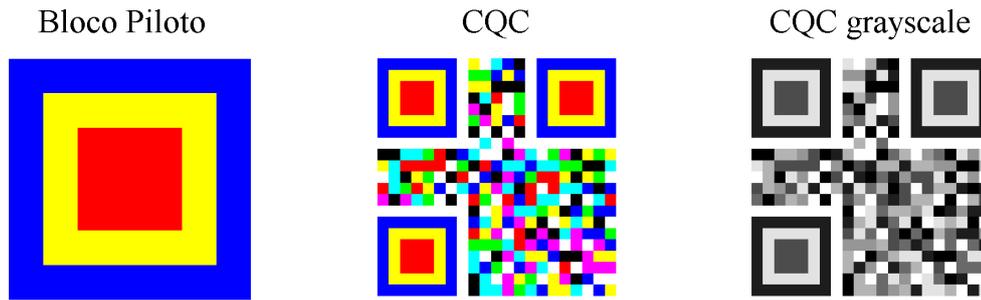


Figura A.1 – Da esquerda para a direita: bloco piloto, CQC com blocos pilotos nos cantos e CQC em escala de cinza com blocos piloto.

obtem-se:

$$\begin{bmatrix} R_{vermelho} & G_{vermelho} & B_{vermelho} \\ R_{amarelo} & G_{amarelo} & B_{amarelo} \\ R_{azul} & G_{azul} & B_{azul} \end{bmatrix} \times T = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (\text{A.2a})$$

$$T = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} R_{vermelho} & G_{vermelho} & B_{vermelho} \\ R_{amarelo} & G_{amarelo} & B_{amarelo} \\ R_{azul} & G_{azul} & B_{azul} \end{bmatrix}^{-1}, \quad (\text{A.2b})$$

em que os subíndices junto aos canais  $R$ ,  $G$  e  $B$  indicam a cor capturada<sup>73</sup> do bloco piloto (vermelha, amarela, ou azul).

Na equação acima, os valores RGB das três cores do bloco piloto são obtidos pela média entre os três módulos de localização. Observe, ainda, que a utilização da cor verde em detrimento ao amarelo poderia produzir a matriz identidade, reduzindo a determinação de  $T$  apenas ao inverso da matriz dos blocos pilotos. Entretanto, a proximidade do valor de luminância do verde ao limiar do padrão QR Code, torna essa escolha menos robusta à sincronização.

A matriz  $T$  permite corrigir cada um dos módulos do CQC, pois ela sintetiza as modificações cromáticas ao longo dos blocos pilotos capturados. Desta forma, qualquer módulo do CQC submetido à Equação A.1 recupera, pois, os valores RGB ortogonalizados. Esta tentativa de ortogonalização dos módulos de cor através das cores de referência aumenta a robustez do código, facilitando a decodificação.

<sup>73</sup> Aqui as cores são consideradas normalizadas, no intervalo  $[0, 1]$ .

## APÊNDICE B – Exemplo de Gráfico de Dispersão de Cores

### B.1 Variações Tonais no Espectro RGB

Nas figuras a seguir serão representadas as variações tonais das cores, tanto nos gráficos de dispersão quanto no próprio barcode.

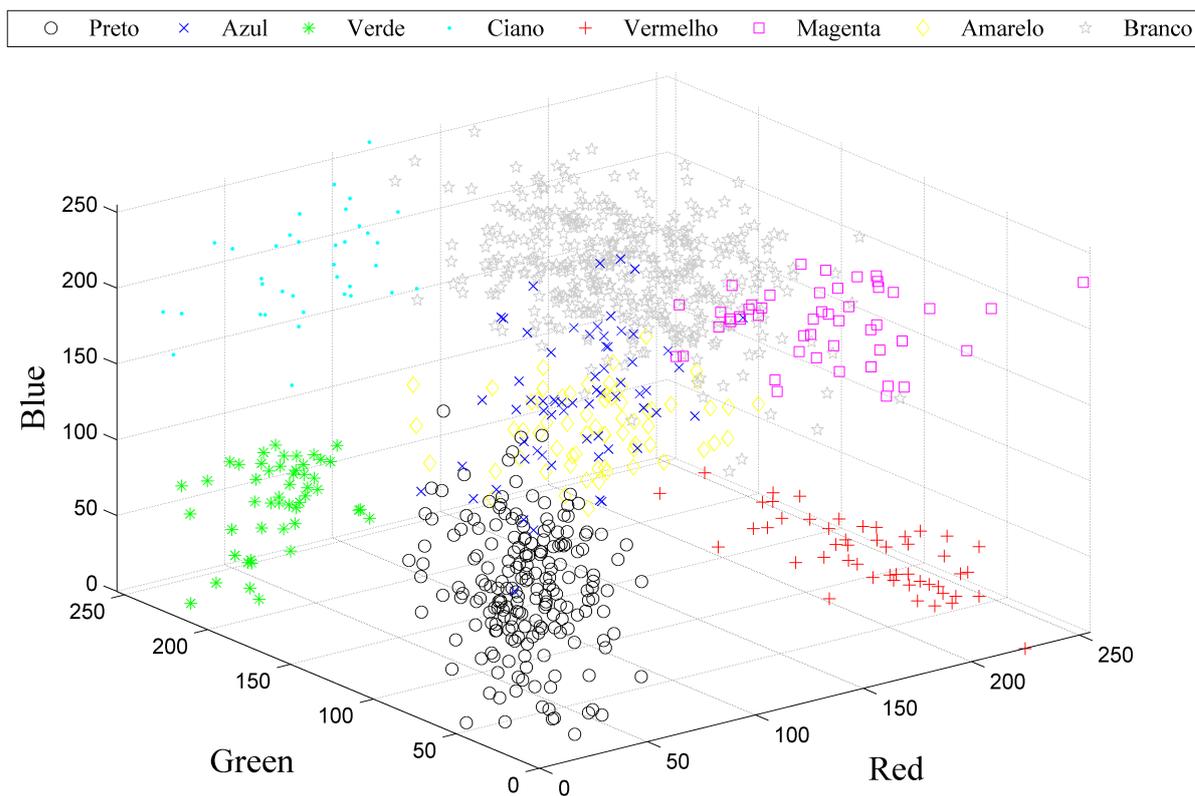


Figura B.1 – Gráfico de dispersão das oito cores utilizadas na síntese do código CQC.

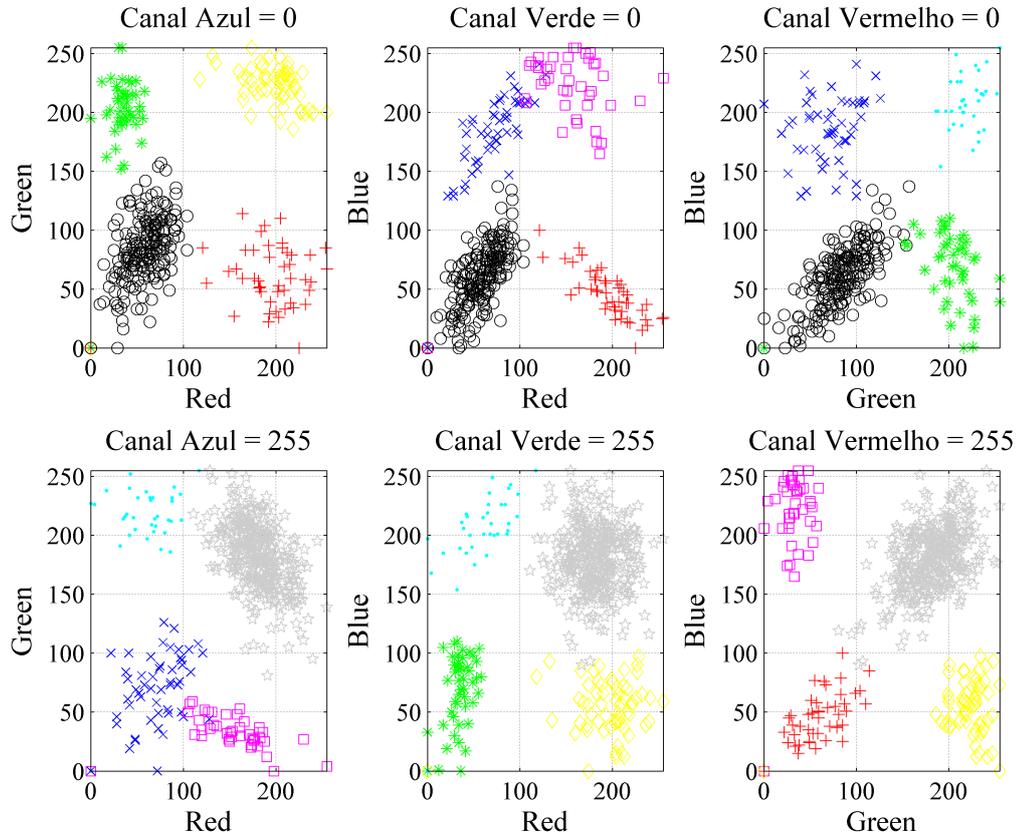


Figura B.2 – Dispersão das cores sob a vista de cada uma das faces do cubo RGB.



Figura B.3 – QR Code submetido aos efeitos do canal, determinando variações tonais.