



**ASPECTOS DA COMPRESSÃO DE IMAGENS DE
AZULEJOS**

Aluno: **Daniel Marques Oliveira** (dmo@cin.ufpe.br)
Orientador: **Rafael Dueire Lins** (rdl@ufpe.br)

Recife, agosto de 2005

Resumo

O objetivo desse trabalho é tirar proveito dos padrões de simetria existentes nos azulejos de cerâmica para obter uma maneira mais eficiente de armazenar esse tipo de imagem e transmitir via redes de computadores. A simetria pode ser considerada como uma forma de redundância, podendo-se gerar uma imagem original a partir de apenas uma parte dela. Portanto, se torna necessário apenas gravação ou transmissão dessa parte e mais alguns dados que informam os padrões de simetria presentes. Ao se gravar parte da imagem há uma redução significativa no seu tamanho em relação original. A inspeção visual nos azulejos de cerâmicas demonstra a baixa quantidade de cores, sendo a possibilidade de explorar esse aspecto para compressão de imagem um outro tema abordado nesse estudo.

Agradecimentos

Dedico este trabalho a minha namorada, Carla, a meus pais, Sérgio e Glória, minha irmã, Martha, pelo apoio e compreensão que me deram durante esse estudo. Agradeço também ao meu professor e orientador Rafael Dueire Lins que sempre confiou no meu trabalho. Sou grato também ao CNPq que patrocinou essa pesquisa durante 1 ano e meio. Agradeço também a Deus por ter dado a oportunidade de poder fazer um estudo sobre esse assunto.

Índice

1	INTRODUÇÃO	1
2	DETECÇÃO AUTOMÁTICA DE SIMETRIA.....	5
2.1	OUTRAS ABORDAGENS.....	5
2.1.1	<i>Geometria Analítica</i>	<i>5</i>
2.1.2	<i>Geometria Computacional</i>	<i>5</i>
2.2	ABORDAGENS EXISTENTES PARA IMAGENS EM 2 DIMENSÕES.....	6
2.2.1	<i>Direta</i>	<i>6</i>
2.2.2	<i>Esquema de Votação</i>	<i>7</i>
2.2.3	<i>Medida contínua.....</i>	<i>7</i>
2.2.4	<i>Manmatha and Sawhney</i>	<i>7</i>
2.2.5	<i>Novo algoritmo.....</i>	<i>7</i>
3	ESTUDO SOBRE A ARMAZENAGEM DO ARQUIVO	7
3.1	SOBRE A BIBLIOTECA UTILIZADA	7
3.2	ESTUDO COM O JPEG E JPEG 2000	7
3.2.1	<i>Resultados</i>	<i>7</i>
3.3	QUANTIZAÇÃO DE IMAGENS	7
3.3.1	<i>Median-cut</i>	<i>7</i>
3.3.2	<i>Optimized-octree</i>	<i>7</i>
3.3.3	<i>Diversity algorithm</i>	<i>7</i>
3.3.4	<i>Geração automática de textura.....</i>	<i>7</i>
3.3.5	<i>Resultados</i>	<i>7</i>
3.4	ESTUDO SOBRE OS FORMATOS DE IMAGENS	7
3.4.2	<i>JPEG vs JPEG-2000</i>	<i>7</i>
3.4.3	<i>PNG vs JPEG e JPEG-2000</i>	<i>7</i>
4	CONCLUSÃO E TRABALHOS FUTUROS	7
5	REFERÊNCIAS	7
6	DATAS E ASSINATURAS.....	7

Glossário

- **Chiral** – uma imagem que não é idêntica ao seu espelho.
- **RGB** – Padrão de cor onde as cores primitivas são vermelhas verdes e azuis (*Red, Green, Blue*).
- **Cubo RGB** – espaço tridimensional de cores onde os eixos são formados pelas cores primitivas do RGB, representando, assim, todas as cores possíveis no sistema RGB.
- **Coloração** – nome dado a um conjunto de cores próximas que, ao olhar para um azulejo, dão a idéia de pertencer a uma única região.
- **Paleta de cores** – Conjunto de cores escolhidas do espaço RGB, que permite que uma imagem, com quantidade de cores distintas menores que 256, possa ser representada através de endereços para esse conjunto, economizando espaço em disco.
- **BMP** – Formato de arquivo de imagem que contém um mapeamento direto, sem compressão, da imagem na memória para o arquivo. Pode ser salvo com 16 milhões de cores ou com paletas de 256 cores ou menos.
- **JPEG** – Formato de arquivo de imagem aberto com perdas, amplamente utilizado em diversas áreas.
- **JPEG-2000** – Formato de arquivo de imagem sucessor ao JPEG que garante melhores taxas de compressão e introduz novas características.
- **GIF** – Formato de arquivo de imagem desenvolvido pela *CompuServe* para imagens com 256 cores que utiliza o algoritmo de compressão LZW.
- **PNG** – Formato de arquivo de imagem desenvolvido para compressão de imagens com 256 cores, feito para ser uma solução alternativa ao GIF.
- **IJG** – Grupo Independente JPEG (*Independent JPEG Group*) responsável pelo desenvolvimento da implementação de referência aberta *libjpeg*.
- **Driver** – Camada intermediária entre a representação da imagem na memória e o formato de arquivo padrão.

1 Introdução

Ao navegar pela Internet, podem-se achar vários *sites* ao redor do mundo dedicados a azulejos cerâmicos. Estes *sites* variam de Museus Virtuais a Catálogos de Venda. A palavra “Azulejo” provém do árabe *az Zulayj*, que significa “pedra polida”, sendo considerada um artefato de luxo na decoração pelas civilizações antigas. A Pérsia foi o centro de desenvolvimento de praticamente todas as técnicas de fabricação de azulejos utilizadas na Europa, sendo provavelmente o seu local de origem. “Os árabes trouxeram do Oriente para a Itália e Espanha.” [4]. A partir da Europa, essa peça se difundiu pelo mundo, sendo um dos objetos de decoração e acabamento mais importantes na arquitetura atual, predominantemente em países tropicais.

A função do azulejo é impermeabilizar e isolar termicamente um ambiente, no entanto, na maioria das vezes, eles formam painéis de pisos e paredes. Desde seu começo, o tema exposto são figuras geométricas com contornos bem definidos, pintados em cores sólidas, mas raramente exibe mais de 5 colorações.

Recentemente, Lins propôs em seu artigo [16] que era possível tirar proveito disso, gravando-se uma imagem com as cores predominantes e informações necessárias para geração de uma textura semelhante à imagem original. Outro aspecto que pode ser levado em consideração mencionado por ele é o armazenamento de uma imagem binária com dados de identificações de suas regiões.

Logo abaixo podem ser vistos dois azulejos antigos, na Figura 1, um azulejo com apenas 4 cores visuais predominantes (laranja, amarelo, verde e preto) e poucas nuances dentre as regiões das cores. Já há casos, como o do azulejo impresso na Figura 2 [61], que possui uma complexidade maior, tornando mais difícil a identificação de cores predominantes (branco, azul escuro, preto, verde grama, verde oceano), além de existirem nuances mais perceptíveis que na imagem.



Figura 1. Azulejo com 4 cores predominantes



Figura 2. Azulejo com nuances mais perceptíveis

Porém, há alguns azulejos mais modernos como os das Figura 3 e Figura 4 [63], onde se aproxima mais de uma pintura. Nota-se que há grande variação de cores na Figura 3 com mudanças mais suaves de tons, o que pode trazer resultados indesejáveis na geração de textura. Já na Figura 04 não há essa mudança, porém existe certa orientação na região representada pelo verde da figura, o que pode dificultar a sua geração sintética.



Figura 3. Azulejo moderno



Figura 4. Azulejo moderno

Lins também afirma fazer uso das simetrias dos azulejos para a compressão das imagens dos mesmos, possibilitando o armazenamento e a transmissão via redes desse tipo de imagens. No seu artigo é mencionada uma classificação em termos de simetria dos azulejos, onde 1s significa a ausência de simetria, o que é muito comum em grandes painéis, como o da Figura 5. 2s representa a simetria diagonal espelhada, em um ângulo de 45 graus. Sendo a Figura 6 um exemplo desse tipo de azulejo. Ele considera também as simetrias rotacionais, usando a notação $i-r$, onde i representa o número de rotações a partir de uma determinada parte da imagem para se gerar o azulejo todo. Há, também, as simetrias espelhadas verticais e horizontais, que ocorrem nos azulejos apresentados na Figura 1 [64] e na Figura 2 [4].



Figura 5. Azulejo brasileiro do tipo 1s, 13,5 x 13,5 cm



Figura 6. Azulejo português do tipo 2s, 13,5 x 13,5 cm

Pode-se ver como exemplo a transformação do azulejo apresentado na Figura 7 [4]. A Figura 8 ilustra a metade diagonal do quadrante superior direito da Figura 7. É possível fazer algumas transformações com a Figura 8 obtendo a Figura 9, ambas contendo a mesma informação. O número total de cores na imagem da Figura 7 está em torno de 60.000, vale salientar, que toda informação como textura, cores predominantes, etc., ainda está presente na imagem reduzida. Há outros objetos, como texturas de tecidos e tapetes, com características parecidas com a de azulejos, sendo simetria uma delas. O que torna possível a utilização desse algoritmo em outras situações.

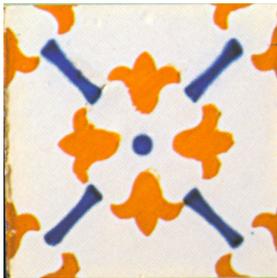


Figura 7. Azulejo Português, 14x14 cm
97kB (JPEG)

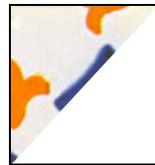


Figura 8. Metade da intra-diagonal de 25% da Figura 7
15kB (JPEG)



Figura 9. Retângulo com Figura 8 transformada
15kB (JPEG)

Imagens a serem estudadas e classificadas podem possuir simetrias bem definidas apresentadas na figura abaixo. O tamanho relativo em relação ao original das imagens *raster* pode ser visto na coluna da direita.

Tabela 1. Padrões de simetria em azulejos a serem estudados

Padrão	Descrição	Tamanho Relativo	Padrão	Descrição	Tamanho Relativo
	Vertical	1/2		Diagonal direita	1/2
	Horizontal	1/2		Horizontal-vertical	1/4
	Diagonal esquerda	1/2		Horizontal-vertical-direita-esquerda-diagonal	1/8

O azulejo apresentado na Figura 7 apresenta padrão de simetria do tipo Horizontal-vertical-direita-esquerda-diagonal, tendo a semente com tamanho igual a 1/8 da imagem original. Alguns azulejos franceses do século XIX, ilustrados na referência [4], demonstram um grau de simetria ainda maior com sua semente de 1/32 em relação a área da imagem original.

As imagens foram extraídas das fontes abaixo, no total foram 307 imagens:

- Das referências [62] [63] [64] [65] [66] [67] [68] [69], nos seus respectivos sites.
- A partir de escaneamentos de imagens fotografadas coloridas a partir de [4] com um *scanner* de mesa feito pela *Hewlett Packard*, modelo *ScanJet 5300* no modo *true-colour*, com 300 dpi.
- A partir de fotografias tiradas com uma câmera digital Olympus D-510 no formato JPEG, modo *best-quality*, no bairro do Poço da Panela localizado em Recife, Pernambuco, Brasil e na loja “Sucata do Azulejo” situado na mesma cidade com o endereço “Rua Paz 186 Lj C Afogados”.

No capítulo 2 serão apresentados os principais algoritmos de detecção automática de simetria, onde na primeira seção serão descritas as principais abordagens existentes na literatura e na seguinte uma proposição de um novo algoritmo de detecção automática será apresentada.

No capítulo 3 será apresentado um estudo sobre os formatos de arquivos de imagens e definição de qual seria o melhor deles para armazenar as imagens, além de introduzir técnicas para gravação de imagens de azulejos com baixa resolução.

2 Detecção Automática de Simetria

A simetria pode ser vista como uma medida binária ou contínua. A primeira considera a existência ou ausência de simetria, sem considerar que há níveis intermediários de simetria. Em contrapartida, a segunda exalta a existência desses níveis intermediários, ensejando o conceito de medida de simetria. Existem várias medidas de simetria, ou seja, medida que informa quão um objeto é simétrico.

O estudo da simetria não é exclusivo da área da computação. Na química, vários processos são dependentes dessa característica [33]. Alguns descrevem que a simetria perfeita de um sistema é necessária para que alguns processos ocorram com eficiência, mas na prática a perfeição não é necessária, e sim uma configuração próxima [31]. Na Biologia, a quantificação da simetria é útil na determinação de desordem e distúrbios em sistemas biológicos [36] [37]. Na Medicina, a medida de simetria é frequentemente um fator importante no diagnóstico de câncer de alguns órgãos. No caso de câncer de pele, uma função pode ser definida para determinar o grau de simetria de um determinado ponto [35].

2.1 Outras abordagens

2.1.1 Geometria Analítica

Um dos primeiros estudos sobre detecção de simetria na geometria analítica demonstrou como seria possível determinar a simetria através de conjuntos convexos. Os estudos avaliam a simetria em conjuntos no espaço euclidiano. É feita uma comparação desses métodos em [9]. As medidas mais conhecidas são as Minkowski, Winternitz e Kovner-Besicovitch, todas quantificam a simetria do ponto de vista teórico e não do método sugerido, para determinar de maneira eficiente tal medida. Porém, Wen apresenta um novo método com complexidade linear em termos de pontos de contorno em [55].

2.1.2 Geometria Computacional

A transformação da detecção de simetria em um problema de reconhecimento de padrões de uma dimensão introduz algoritmos eficientes na detecção de simetrias espelhadas, rotacionais e na localização do eixo de simetria [1] [2] [7] [11] [13] [27]. A complexidade desses algoritmos são de $O(n \log n)$, onde n é o número de objetos geométricos.

Esses algoritmos são extremamente eficientes, mas assumem a ausência de ruído na imagem de entrada. Qualquer erro, seja originado de operações computacionais, seja de desvios de localização, pode afetar o resultado. Todavia, limites de tolerância podem ser encontrados na referência [1].

2.2 Abordagens existentes para imagens em 2 dimensões

A simetria em 2D pode ser discutida como uma característica global ou local [31]. No primeiro caso, todos os pontos da imagem contribuem para a determinação da simetria. Já no segundo caso, todo elemento da simetria pertence ao um subconjunto local do objeto, no caso de imagens, pequenas partes dela. Métodos globais são bem mais eficientes em tempo de execução. Geralmente eles possuem complexidade linear, mas são sensíveis a ruídos e oclusões. As abordagens locais possuem métodos mais robustos, mais fáceis de paralelizar, porém apresentam uma complexidade maior. Em [39] é feito um estudo comparativo entre algoritmos de esquemas de votação (que vai ser visto logo a seguir) que utilizam as duas abordagens. Existem, também, inúmeros exemplos de ambas abordagens em [31]. Em [54] mostra-se um algoritmo detecção de faces humanas através de simetrias locais, primeiramente ele separa a imagem em regiões de interesse (faces humanas), gerando algumas estruturas baseadas nessas regiões para auxiliar a detecção.

A detecção de simetria 2D em imagens digitais apresentam, atualmente, 4 diferentes abordagens:

- Direta
- Esquema de votação
- Medida contínua
- Manmatha and Sawhney

2.2.1 Direta

A maneira mais simples de se verificar a simetria de uma imagem é através de uma transformação simétrica de parte dela e comparar essa transformação com a imagem original [25]. Em [13], Kuehnle adota essa estratégia para detectar automóveis.

Pode-se combinar esse método com detecção em vários níveis de resolução. Começa-se pelo nível mais baixo, o mais eficiente, aumentando-se paulatinamente [6]. Krahe [12] sugere outro método, o da sobreposição. Ele baseia-se na autocorrelação de

escalas polares da imagem, sendo intolerantes a ruídos e oclusões. Em [25], sugere-se que a sobreposição pode ser usada como uma medida de simetria.

2.2.2 Esquema de Votação

O princípio básico desse algoritmo é a escolha de pares de pontos para determinar o eixo de simetria. Cada par “vota” sobre a escolha de um determinado eixo de simetria. Essa abordagem se baseia no fato de que pontos correspondentes das imagens simétricas determinam seu eixo de simetria.

Na referência [22] é feita a representação de um polígono de n pontos por meio de códigos em cadeia. Portanto, a metade de n representa o número de pares de pontos de uma imagem a serem comparados. Nesse método, apenas a direção do eixo é definida, mas não sua posição, pois é incorreta a suposição do eixo dividir igualmente os pontos do polígono.

A transformada de Hough é usada em [14] [21]. Define-se que o eixo votado pelo par passa pelo seu ponto médio do segmentado formado por eles, sendo também perpendicular a esse segmento. O eixo com maior número de votos é selecionado como o eixo de simetria. Uma modificação desse método para detecção rotacional é sugerida em [38]. Uma variante da transformada de Hough que utiliza essa abordagem pode ser achada em [23][24][20]. A utilização de redes neurais *feed-forward* é usada para se detectar simetria através de bordas sendo descrita em [32].

Observa-se que essa abordagem é complexa, vários métodos são sugeridos para reduzir a complexidade de agrupamento de pontos em regiões ou segmentos de curvas. Essa estratégia reduz o número de possíveis pares de votos. Esquemas de votação são robustos em relação a ruídos e oclusão das imagens a serem processadas. Porém, pode-se pagar um preço alto ao detectar equivocadamente a existência de simetrias. Em muitos algoritmos, a imagem dada é simétrica, sabendo-se, a priori, a quantidade de eixos existentes das simetrias locais. Existem vários outros métodos citados em [31].

2.2.3 Medida contínua

As abordagens anteriores consideram a simetria como uma característica binária. A noção de simetria como uma medida contínua é pouco discutida na literatura. Em [40], define-se uma medida de simetria em padrões, chamada de “simetropia”. Nela é utilizada teoria da informação para quantificar a distribuição de simetria do padrão. Marola [41] apresenta um coeficiente de simetria espelhada de determinado eixo.

A noção de medida de *Chirality* é apresentada em [3][8][10][53]. Essa medida baseia-se na diminuição na diferença do volume entre o objeto e sua reflexão, variando-se o plano de reflexão. Em [10], apresenta-se essa medida considerando-se os efeitos rotacionais dos objetos *chiral*. Em [52] é feita uma crítica à metodologia usada em [53], sugerindo uma modificação no algoritmo. Em [29] [30] [31] é apresentado o conceito de distância de simetria bem como medidas genéricas contínuas de graus de simetria.

2.2.4 Manmatha and Sawhney

Essa abordagem é recente, sendo apresentada na referência [17]. Gaussianas e suas derivadas em escalas múltiplas são usadas para achar o eixo de simetria local, sem uma segmentação a priori. O algoritmo pode ser tanto aplicado na simetria espelhada, como na rotacional. Nele assume-se que a direção do eixo de simetria é conhecida.

2.2.5 Novo algoritmo

O algoritmo proposto junta alguns elementos da abordagem Direta e de Medida Contínua de uma maneira completamente diferente e nova. Da Direta, foi trazida a idéia de comparação *pixel a pixel* das partes da imagem, onde essa comparação depende da simetria a ser testada. Da medida contínua, a noção de simetria como uma característica não-binária. A idéia central desse algoritmo é produzir uma medida do quão diferentes as cores dos *pixels* são.

O padrão de simetria pode ser obtido ao aplicar recursivamente o algoritmo de detecção de simetria. É pertinente observar: quanto maior o grau de simetria, maior o ganho na compressão da imagem [16]. Azulejos de cerâmicas tornam a detecção de simetria mais fácil que em outros tipos de imagens porque:

- Suas imagens possuem temas geométricos simples;
- Eles possuem poucas cores;
- Possuem tamanhos padrões;
- O eixo de simetria dos azulejos corresponde ao eixo central da imagem, podendo apresentar um desvio de alguns *pixels*.

Agora, serão apresentados os passos do novo algoritmo.

2.2.5.1 Removendo Bordas

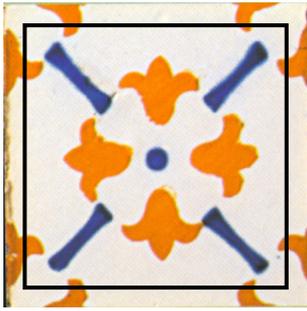


Figura 10. Seleção da área de análise

Ao observar detalhes da imagem na Figura 7 verifica-se que a imagem ao lado apresenta “manchas” em sua periferia. Tal problema é comum ao lidar com imagens de azulejos históricos como os impressos na referência [4]. Essas manchas foram introduzidas pelo tempo, conservação inapropriada, excesso de material quando o azulejo foi fabricado etc. Em contrapartida, ele não faz parte da “informação” e pode ser visto como um ruído adicionado na imagem. Logo, deve ser feita uma filtragem antes que o processo de detecção comece. A análise é só feita em 80% da imagem, o que além de desconsiderar o “ruído” no processamento, aumenta a eficiência do algoritmo, pois uma menor quantidade de pontos é analisada.

A Figura 10 demonstra a área selecionada para análise da Figura 7. Pode-se verificar que não há “ruído” de borda na área selecionada. Para simplificar, será assumido que o eixo de simetria é o centro da imagem original, pois pequenas variações do centro não causam problemas para as análises estatísticas feitas que servem de base para as detecções de simetria.

2.2.5.2 Segmentação

A análise de simetria proposta checa o número de *pixel* que coincidem com transformações e faz estatísticas com eles. Porém, nem todos os *pixels* carregam o mesmo “grau de informação”. O *pixel* do “cluster” de fundo precisa ser reconhecido e tratado da maneira correta. Figura 11 apresenta uma imagem não-simétrica onde o número de *pixel* de “fundo” excede o de “informação”. Qualquer algoritmo descrito que conta o número de *pixels* “coincidentes” poderia concluir equivocadamente que há uma simetria vertical na imagem.

Portanto, uma estratégia de segmentação é introduzida: o primeiro passo do algoritmo é achar a distribuição de cores dos *pixels*. Geralmente, essa distribuição forma uma distribuição multi-modal gaussiana centrada no fundo e cada cor importante da imagem. A imagem apresentada na Figura 11 [4] apresenta 22 mil cores e seu histograma pode ser visto ao seu lado, na Figura 12, onde é possível observar uma distribuição bi-modal com predominância das gaussianas da direita, que formam o fundo da imagem. É notável o ofuscamento dos *pixels* de fundo em relação aos *pixels* de

“informação”, impedindo a visualização tri-modal do histograma nas cores branco, laranja e azul escuro.



Figura 11. Azulejo português não simétrico

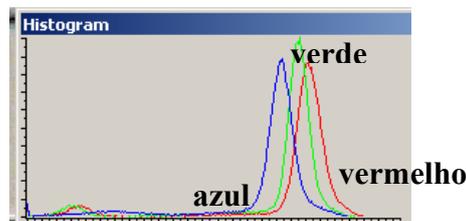


Figura 12. Histograma de cor da Figura 11

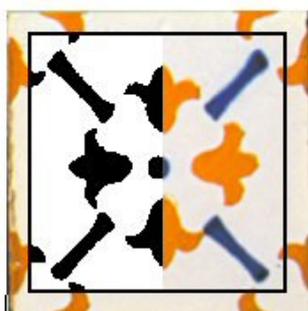


Figura 13. Imagem segmentada da Figura 7

A segmentação que usa a cor mais freqüente como fator pode causar erros. Há a possibilidade de possuírem grandes nuances nas cores de fundo, onde a freqüência de uma cor do *pixel* de informação sobrepõe às cores de fundo. A cor de fundo é calculada através aplicação freqüência de distribuição da cores com uma variação do *cluster* para cada uma delas, obtendo assim o número de membros pertencentes a cada *cluster* de cores. O *cluster* com maior número de membros é considerado a cor de fundo mais freqüente.

Essa parte do algoritmo pode ser implementada em dois passos. No primeiro, uma tabela de freqüência de *cluster* é formada ao varrer a imagem, onde, para cada *pixel* lido, a cor correspondente é incrementada, bem como as cores ao redor de acordo com uma variação. O *cluster* de fundo corresponde a uma entrada na tabela de maior valor. A variação de ± 2 foi considerada a mais adequada para a proposta apresentada. Todos os *pixel*, cujas cores pertencem a uma esfera centrada no cluster de fundo com um raio de tolerância t , foram mapeados em branco, sendo os outros mapeados em preto. Os experimentos demonstraram que um valor de t igual a 40 trouxe melhores resultados.

A Figura 13 demonstra a imagem do azulejo da Figura 7 com o lado esquerdo da área de análise com *pixel* de fundo em branco e os de informação em preto. Na prática, toda imagem na área de análise é segmentada.

2.2.5.3 Estatísticas da Imagem

Tanto a imagem segmentada, quanto a original são usadas para coletar a estatística necessária para verificar a existência de padrões de simetria. Quatro padrões de simetria descritos na Figura 14 são analisados. Cada padrão separa a imagem em

duas regiões: origem e destino. Para cada região é feita uma divisão em 4 sub-regiões de tamanhos iguais, correspondendo a cores diferentes na Figura 14.

Sete parâmetros são medidos:

- i** – número de *pixels* de informação da imagem segmentada;
- b** – número de *pixels* de fundo da imagem segmentada;
- ii** – número de *pixels* de informação coincidentes entre a origem e o destino de cada sub-região da imagem segmentada;
- bb** – número de *pixels* de fundo coincidentes entre a origem e o destino de cada sub-região da imagem segmentada;
- bi** – número de *pixels* de fundo mapeados em *pixels* de informação em cada sub-região da imagem segmentada;
- ib** – número de *pixels* de informação mapeados em *pixels* de fundo em cada sub-região da imagem segmentada;
- dif_ii** – a soma dos quadrados da distância RGB entre o *pixels* da origem com o seu mapeado no destino de cada sub-região da imagem original;

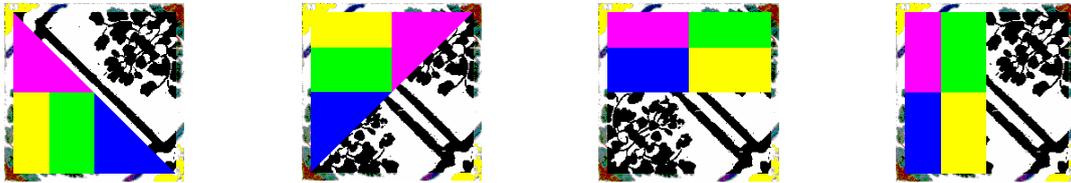


Figura 14. Padrões de simetria com a divisão em sub-regiões

Após o cálculo dos parâmetros descritos anteriormente, é feita uma análise em cima deles, podendo ser vista na tabela abaixo.

O parâmetro *ldii* calculado como na Tabela 2 é uma variante do cálculo do *Peak-Signal-to-Noise-Ratio* (PSNR). O PSNR provê uma medida de qualidade de imagens geradas a partir de uma operação. O quanto maior o PSNR, melhor a qualidade da imagem gerada, ou seja, nesse caso, as cores comparadas coincidem. A fórmula abaixo é a maneira mais comum de calculá-la:

$$PSNR = 20 \log_{10} \frac{255}{\sqrt{\frac{\sum [f(i,j) - F(i,j)]^2}{N \times M}}}$$

Onde:

- $f(i, j)$ é o valor do *pixel* (i, j) na imagem original.
- $F(i, j)$ é o valor do *pixel* (i, j) na imagem gerada
- N e M é o número de linhas e colunas da imagem original, respectivamente

Tabela 2. Estatística de parâmetros da simetria de azulejos

Descrição	Fórmula
Diferença da quantidade de informação de toda imagem	$Id = \frac{ 2 * (ii + ib) - i }{2 * (ii + ib)}$
Diferença da quantidade de fundo de toda imagem	$Bd = \frac{ 2 * (bb + bi) - b }{2 * (bb + bi)}$
Percentual de <i>pixels</i> de informação coincidente para cada sub-região	$pii = \frac{ii}{ii + ib}$
Percentual de <i>pixels</i> de fundo coincidente para cada sub-região	$pbb = \frac{bb}{bb + bi}$
Percentual de <i>pixels</i> de fundo não-coincidente para sub-região	$pmn = \frac{ib + bi}{(ib + bi + bb)}$
Distancia média no espaço de entro os <i>pixels</i> de informação	$ldii = 10 \log \left(\frac{dif - ii}{ii^2} \right)$

Os parâmetros apresentados na Tabela 2 alimentam o algoritmo de teste da existência de padrões de simetria na imagem. O ajuste dos parâmetros foi feito empiricamente – que vai ser visto na próxima seção –, sendo o resultado apresentado na Tabela 3.

CONTADOR = 0;

PARA CADA sub-região **DO**

SE (tol_Pbb < pbb) CONTADOR += kbb;

SE (tol_Pii < pii) CONTADOR += kii;

SE (tol_Pmn > pmn) CONTADOR += kmn;

SE (tol_Pldii > ldii) CONTADOR += kldii;

SE (tol_Id > Id) CONTADOR += kId;

SE (tol_Bd > Bd) CONTADOR += kBd;

SE (CONTADOR > Corretos) ENTÃO É SIMÉTRICO;

Algoritmo proposto para detecção de simetria

Tabela 3. Parâmetros estatísticos para detecção de simetria de azulejos

Parâmetro	Simetrias Horizontais e Verticais	Simetrias Diagonais
Corretos	28	32
tol_pii	80	64
tol_Id	2	0
tol_pbb	88	70
tol_Bd	2	2
tol_pnm	0	0
ldii	8	4
kpii	2	4
kpip	3	4
kpbb	2	3
kBd	2	1
kpnm	1	2
kldii	4	3

O algoritmo descrito acima é capaz de detectar um nível maior de simetria em imagens de azulejos. Caso algum padrão de simetria seja encontrado, a aplicação do algoritmo prossegue recursivamente, até que não haja mais simetria detectada.

2.2.5.4 Metodologia de pesquisa dos melhores parâmetros

Para pesquisar os melhores parâmetros, testes não recursivos foram feitos gravando-se os dados estatísticos necessários para teste descritos na Tabela 2 para cada padrão de simetria descrito na Figura 14. O que tornou possível teste dos melhores parâmetros sem a necessidade de executar os cálculos das estatísticas a cada nova pesquisa. Os testes dos parâmetros foram feitos de maneira exaustiva, o que consumiu um enorme tempo de cálculo. Para minimizar esse tempo, a pesquisa foi feita em duas etapas, garantindo que os valores dos parâmetros estivessem perto do ideal.

Primeiramente foram testados os parâmetros do tipo tol_X num intervalo de 0 a 100, incrementando o valor de 2 em 2, com exceção do parâmetro tol_ldii em que o intervalo foi de -16 e 30, enquanto que os parâmetros kX foram fixamente atribuídos 1. O valor do parâmetro $corretos$ foi testado entre os valores de 6 a 18.

Ao obter os valores dos parâmetros tol_X , partiu-se para obtenção dos parâmetros kX . Estes foram variados entre 0 e 6, incrementando o valor de 1 e 1. O parâmetro de corretos variou-se entre 6 e a soma dos valores de kX atuais. Nas duas etapas, considerou-se o melhores parâmetros aqueles em que o número de falso-positivos não excedesse 7 e contivesse o maior número de acertos.

2.2.5.5 Resultados obtidos

O algoritmo apresentado foi testado sem recursão num grupo de imagens de 187 azulejos. O número de imagens que tiveram as simetrias detectadas corretamente foi de 133, ou seja, 71,13% das imagens. Houve apenas 6 imagens com resultados falso-positivos, ou seja, 3,2% das imagens a simetria foi detectada onde não havia nenhuma. Por outro lado, 48 imagens apresentaram resultados falso-negativos, representando 25,6% das imagens onde não foi detectada a simetria existente.

O resultado apresentado é conservador, porque as imagens apresentam um grau de complexidade razoável para verificar a robustez do algoritmo. Em termos gerais, se houvesse simetria ela era detectada pelo algoritmo. De qualquer modo, como o explicado abaixo, algumas imagens apresentaram um padrão quase simétrico sendo difícil detectar a não simetria, até por seres humanos.

2.2.5.6 Limitações do Algoritmo

Os parâmetros apresentados na Tabela 3 foram atribuídos para prover resultados conservadores e fazer com que o número de falso-positivos fosse minimizado. Os azulejos turcos apresentados na Figura 15 [66] são responsáveis por 3 das 6 imagens com resultado falso-positivas do teste com 187 imagens. É possível admitir que o primeiro e segundo azulejo apresentasse um alto grau de complexidade simétrica que até pessoas possuem dificuldade de identificar traços não simétricos das imagens.

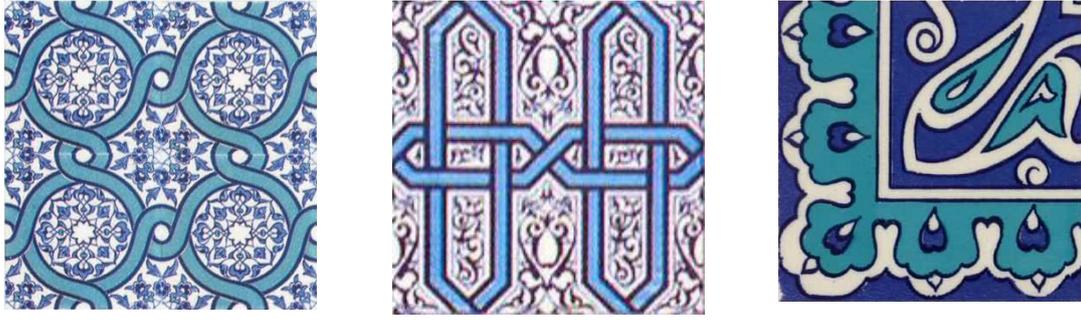


Figura 15. Azulejos turcos não-simétricos.

3 Estudo sobre a armazenagem do arquivo

Neste capítulo iremos descrever uma análise feita para identificar a melhor maneira de armazenar os arquivos de imagens de azulejos. Inicialmente, demonstrar-se-ão algumas notas sobre a biblioteca utilizada no desenvolvimento desse trabalho.

Depois será feita uma comparação entre o JPEG e JPEG-2000, sendo estes amplamente utilizados na digitalização de fotos e imagem. Logo após, será feito um estudo sobre a quantização de cores (diminuição de cores) deste tipo de imagem, onde tentará se tirar proveito do fato dos azulejos apresentarem poucas colorações. O armazenamento do azulejo com um número de cores reduzido poderia ser melhorado através da geração de uma imagem sintética do mesmo a partir de informações extraídas da imagem original, um estudo referente a isso é feito na seção 4.

3.1 Sobre a biblioteca utilizada

Durante esse estudo foi utilizada a biblioteca gratuita GFL SDK, disponível no site da <http://www.xnview.com>. Com essa biblioteca é possível ler e gravar os formatos de imagens mais populares. No nosso estudo, utilizaremos o *driver* para JPEG, PNG, GIF e JPEG 2000.

Internamente, o GFL SDK utiliza a biblioteca *libjpeg* desenvolvida pelo grupo independente do JPEG (IJG) para gravar apenas JPEG, disponível em <http://freshmeat.net/projects/libjpeg>. Essa biblioteca define um fator de qualidade de 1 a 100, onde 1 representa a menor qualidade e 100 a maior. Tal fator não é proveniente da especificação original do JPEG. Outros *driver* definem seu próprio fator, podendo ter escalas e ajustes na imagem diferentes.

No caso do JPEG 2000, a biblioteca utiliza o *driver* livre *jasper* (<http://www.ece.uvic.ca/~mdadams/jasper>). Esse *driver* é bastante conhecido na literatura, sendo citado no site do oficial do padrão JPEG (<http://www.jpeg.org>) e usado em vários programas conhecidos comercialmente e academicamente. Do mesmo modo, o formato define uma escala de qualidade entre 1 e 100: 1 para menor qualidade e 100 para maior. Por ser de outro grupo de desenvolvedores em relação ao formato anterior, não será considerada uma correlação prévia entre seus fatores de qualidade. Essa correlação será estabelecida empiricamente. A escala de qualidade do JPEG-2000 também não é padronizada na definição do padrão. A não utilização de um mesmo fabricante para os dois formatos se deve a ausência de bibliotecas confiáveis que o faça.

3.2 Estudo com o JPEG e JPEG 2000

Nesta seção será mostrado o estudo comparativo entre os formatos JPEG e JPEG-2000 realizado com os azulejos. Nestes foram separados grupos com características similares. Pelo menos um membro de cada grupo foi observado nos dois formatos.

Todas as imagens foram salvas com fatores de qualidade entre 1 e 100, nos dois formatos, variando-se 5 unidades dessa escala. Para cada arquivo salvo foi calculado o fator de qualidade PSNR em relação à imagem original. Nesse fator, quanto menor seu valor, maior a distinção entre a imagem original e a imagem proveniente da salvagem com outros fatores de qualidade.

Todas as imagens estavam originalmente no formato JPEG. Inicialmente, foi determinado o fator de qualidade da imagem original através da comparação visual com os arquivos salvos anteriormente, pois o JPEG não possui método padronizado que identifique tal propriedade.

Depois se determinou visualmente o fator de qualidade do JPEG-2000 em que o arquivo fosse o menor possível sendo as perdas insignificantes. A partir disso é que se pode extrair uma relação de equivalência entre os formatos, onde foi possível comparar, tanto o PSNR como o tamanho do arquivo final de ambos os formatos.

Houve a tentativa de se realizarem os testes de maneira automática através da pesquisa pelo arquivo JPEG-2000 correspondente ao de tamanho próximo do JPEG salvo, determinando um fator de qualidade do JPEG-2000 para cada imagem. Uma inspeção visual foi realizada dentre as imagens e verificou-se que são poucas as distinções dentre as imagens dos dois formatos, justificando o uso de uma comparação direta dos fatores de qualidade das imagens resultantes dos dois padrões. Ao fazer isso é possível serem estabelecidas estatísticas mais precisas sobre os dois formatos. Com o PSNR, isso foi feito de maneira análoga, e mesmo considerando certa tolerância, o método foi ineficiente.

3.2.1 Resultados

Para analisar melhor o resultado, as imagens foram separadas em três grandes grupos: JPEG superior, JPEG-2000 superior e semelhante. Este último foi considerado quando a razão JPEG-2000 - JPEG e sua inversa não fossem menor que 90%, minimizando o fator subjetivo do observador ao analisar as imagens. A média do fator de qualidade do JPEG foi de 78,16, com variância de 2,48. Já o JPEG-2000 apresentou

um fator de 74,14 em média, com variância de 3,08. Em 38,44% dos casos, os formatos e as taxas de compressões foram próximas, não se considerando como uma superioridade de um formato sobre outro.

Observou-se que o JPEG-2000 produziu um efeito, à medida que seu fator de qualidade decresce, semelhante a filtros de suavização de imagens e, em muitos casos, a homogeneização de regiões. Essa característica está presente no JPEG, porém não é tão perceptível. A Figura 16 [65] ilustra tal fenômeno no desaparecimento das linhas onduladas claras e menos perceptível nas regiões dos corpos do peixe. Há também a ocorrência de serrilhado conforme diminuição de qualidade [15].

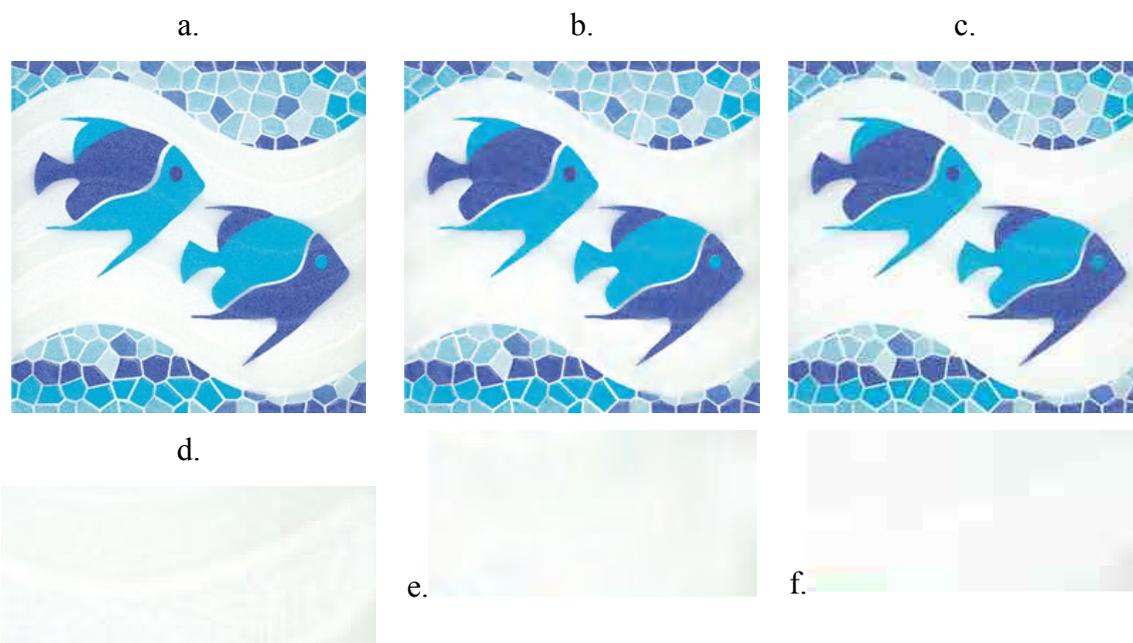


Figura 16. a. Imagem original b. Imagem salva no formato JPEG-2000 com qualidade 70 c. Imagem salva no formato JPEG com qualidade 25 d. Detalhe da imagem original e. Detalhe da imagem salva no JPEG-2000 f. Detalhe da imagem salva no JPEG

O JPEG se apresentou mais econômico em 34 imagens (11,07%). O melhor aproveitamento, na maioria dos casos, deve-se ao fato do JPEG-2000 “suavizar” a imagem de modo que há a perda de informações relevantes.

Na Figura abaixo, a imagem original quando salva em JPEG (fator 80) ocupa 73,45% de espaço em relação a sua cópia no formato JPEG-2000 (fator 85). Primeiramente, é exibido o azulejo como um todo, ao lado há uma ampliação do detalhe do azulejo onde há uma textura verde, logo ao lado, há o mesmo detalhe salvo com o fator 50, onde há praticamente perda total da textura. Nota-se que o JPEG-2000 contribuiu para a “extinção” da textura na imagem, pois a degradação começa a ser

perceptível a partir do fator 75. No JPEG, a suavização só ocorre a partir do fator 40. A Figura 17 também apresentou melhor compressão com o JPEG, ao colocar um fator de qualidade inferior 90 já é notável o aparecimento de suavizações.

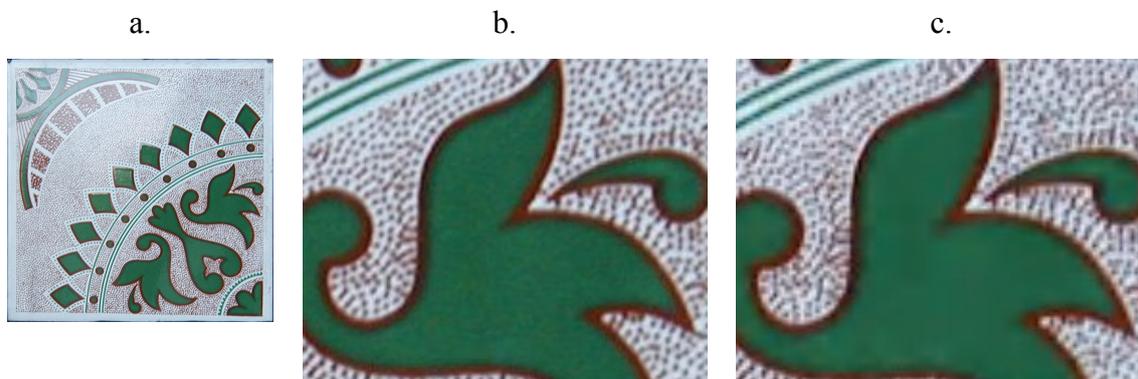


Figura 17. a. Imagem original b. Ampliação de detalhe da imagem original c. Detalhe de imagem salva no formato JPEG-2000 com qualidade 50

Em 72,64% dos casos, o JPEG-2000 foi superior. Na Figura abaixo [66] pode ser observado o caso em que o formato apresentou maior aproveitamento em relação ao seu anterior. Ocupando apenas 34,62% se compararmos com seu antecessor. Tal acontecimento não era esperado, pois há muitos detalhes pequenos na imagem que poderiam desaparecer devido à suavização decorrente da compressão.



Figura 18. Imagem de azulejo turco de 300x300 pixels

3.3 Quantização de imagens

Na literatura de processamento de imagens, quantizar imagens é sinônimo de diminuir o número de cores delas. São propostos vários métodos, o principal objetivo da maioria deles é minimizar a percepção dessa diminuição. Há técnicas auxiliares como transformações de meio-tom e de utilização de pseudo-cores são normalmente vistas em

conjunto com a quantização. A referência [42] traz mais detalhe sobre as duas técnicas. Geralmente a imagem quantizada utiliza tabelas de procura (CLUT - *color lookup table*) que contém a cor num formato de espaço de cores maior, como o RGB, o que pode dar a impressão que a imagem possui grande quantidade de cores distintas.

A quantização de imagens é utilizada nesse contexto para tentar diminuir o tamanho do arquivo final. Pode-se gravar a imagem apenas com 16 cores ou 2 cores. Poderia se transformar a figura quantizada em uma sintética, que simula a textura das regiões de cores através de algumas informações trazidas da original. Com 16 cores, cada cor da imagem resultante representaria uma região, com 2 cores poderiam ser representadas n regiões, onde as fronteiras de cada região poderiam ser representadas por uma cor (preto) e o conteúdo pela outra (branco). A transformação da representação de 16 cores para 2 é trivial, é necessário apenas aplicar um filtro de detecção de bordas, substituindo na presença de bordas o *pixel* correspondente pelo valor preto e na ausência pelo valor branco.

A seguir serão mostrados os principais métodos de quantização.

3.3.1 Median-cut

Este método pode ser visto com mais detalhes em [43] e [44]. Seu propósito consiste na tentativa de dividir o cubo RGB, que contém a frequência das cores existentes na imagem, de maneira igualitária. Cada região de cores deve conter um número próximo de pontos. O resumo do algoritmo pode ser visto abaixo:

1. Coloca-se no cubo RGB as cores presentes nas imagens com sua frequência
2. Pegue as caixas de cores recém-criadas do cubo RGB até atingir o número máximo de cores
 - Selecione a maior dimensão dessa caixa (vermelho, verde ou azul);
 - Divida essa caixa na sua mediana com um plano perpendicular a maior dimensão, ou seja, as metades devem possuir a mesma quantidade de pontos;
3. Calcule a cor final de caixa a partir da média das cores.

3.3.2 Optimized-octree

A idéia central desse algoritmo é montar uma árvore com informações sobre imagem analisada [46] [47]. Nessa árvore há uma distinção na sua estrutura entre os nós

folhas e não-folhas (raiz e seus intermediários com as folhas). As folhas contêm a cor que vai representar um determinado conjunto de cores e os nós não-folhas contêm informações sobre as folhas descendentes. Os nós próximos da raiz possuem informações menos precisas, pois armazenam dados dos bits mais significativos das componentes.

O segredo desse algoritmo está na diminuição das folhas na adição de uma cor nova na árvore, de modo que ultrapasse a quantidade máxima de cores. Quando isso acontece, duas folhas são escolhidas, sendo então agrupadas com seu pai comum para formar uma única folha, estabelecendo assim um teto na quantidade de folhas. A consequência disso, é que durante todo processo, a imagem não é representada por mais que um número máximo previamente definido. A memória utilizada pode ser menor ao comparar-se com o Median-Cut. A escolha do número de níveis intermediários entre a raiz e a folha fica a cargo do desenvolvedor. A referência [49] informa que 4 ou 5 níveis são suficientes para maioria dos casos.

3.3.3 Diversity algorithm

O objetivo principal desse algoritmo é garantir a diversidade de cores na imagem resultante [48]. Nele não há a preocupação do quanto a cor é freqüente, a imagem final apresenta cores contrastantes. O algoritmo é descrito logo a seguir:

1. Forma-se uma lista ordenada por ordem de freqüência da cor na imagem, da maior freqüência para a menor. Denomina-se um conjunto de cores escolhidas que conterão as cores da imagem resultante.
2. Retira-se a cor mais freqüente da lista e adiciona-a ao conjunto de cores escolhidas.
3. Procura-se a cor mais distante da escolhida anteriormente e coloca-se no conjunto de cores escolhidas. Repete-se esse passo até que a quantidade de cores escolhidas atinja o limite de dado.

3.3.4 Geração automática de textura

Existem inúmeros algoritmos que tem como base a geração sintética de textura. Mello [50] propõe um método para geração da textura de papéis. Em seu outro artigo [51], Mello descreve a geração para os papéis dos documentos históricos. Ambos se baseiam em gaussianas presentes no histograma para gerar a textura. Há também outros

mecanismos mais genéricos sugeridos em [56] e [57], que utilizam redes neurais e lógica *fuzzy*, respectivamente, para fazer esse tipo de processamento.

Apesar de existirem vários estudos para geração sintética de texturas, o que foi utilizado nesse estudo foi a base de muitos algoritmos. Assim, é possível estabelecer um limitante inferior em termos de qualidade da textura gerada, podendo ser utilizada outras técnicas mais avançadas para obter melhores resultados.

A textura deverá ser gerada para cada região da imagem, definida pela quantização da imagem, analisadas aqui pelos algoritmos descritos anteriormente. Nesse caso, considera-se uma região os pontos de toda a imagem que pertencem a cor quantizada. Ou seja, vão fazer parte de uma mesma região representada pela cor X , os pontos da imagem original que mapeados na imagem quantizada possuem essa cor X . Note que uma região pode estar presente em várias áreas da imagem, não precisam estar conectados pela vizinhança entre *pixels*.

O algoritmo para geração automática pode ser visto logo abaixo:

1. Cada cor presente na imagem quantizada é definida como uma região;
2. Determinar o histograma de cada região em relação a original;
3. Gera-se uma imagem, onde a cor do ponto pertencente a certa região é determinada de acordo com a probabilidade definida pelo histograma calculada anteriormente

3.3.5 Resultados

Dentre os algoritmos de quantização aqui apresentados, o que se saiu melhor foi o *median-cut*, pois garantiu que o espectro de cores para cada região variasse menos, sendo mais visível nos outros algoritmos. O resultado com os outros algoritmos é, de certa forma, imprevisível, pois não garante a largura do espectro das imagens, como será visto a seguir. Para ilustrar, iremos demonstrar um teste feito com a Figura 20 [64]. Ela possui uma resolução de 275x277 e uma quantidade de 31530 cores distintas no espaço RGB. Essa Figura foi escolhida, pois a mediana das quantidades de cores pertencia a essa imagem dentre as 307 utilizadas nesse estudo. O histograma dessa imagem pode ser observado na figura ao lado.

Além de estudar as imagens quantizadas, o objetivo é dividir em regiões de cores para a futura reconstrução da imagem original. Portanto, serão exibidos

histogramas de algumas regiões formadas. Será analisada a quantidade de *pixel* pertencentes à região de maior e de menor população, assim como a largura do espectro (o maior espalhamento dentre as componentes). Será exibido também o valor da cor no formato $rgb(R,G,B)$. Onde R representa o valor da componente vermelha, G verde e B azul. A região mais populosa será aquela que contiver o maior número de *pixel*. Nos histogramas a serem exibidos, cada componente é representado por sua respectiva cor.

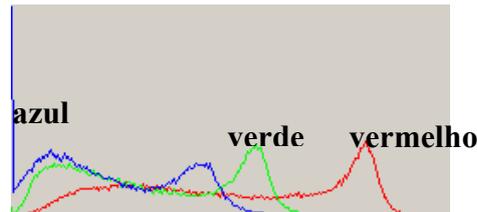


Figura 19. Histograma da Figura 20.

Figura 20. Imagem original, com resolução de 275x277 e 31530.

Logo abaixo pode ser vista a imagem quantizada com o *diversity*. A seu lado consta o histograma da região mais populosa e de maior espectro (cor marrom $rgb(163,113,90)$), e ao lado o histograma da região com menor (cor esbranquiçada $rgb(255,252,255)$), referente ao reflexo da câmera). A primeira representa 17494 *pixels* e uma largura de espectro de 168 da imagem original, a segunda 209 *pixels* com largura de espectro 98.

O interessante dessa figura é que a proposta do *diversity* – que seria de diversificar – não foi refletida nesse caso, pois a cor mais distante da última escolhida é próxima da anterior desta. Nota-se a aparência de gaussianas no histograma da região mais populosa, o que muitas vezes é desejado no desenvolvimento de um algoritmo de geração de texturas.



Figura 21. Resultado da quantização com o *diversity* da Figura 20.

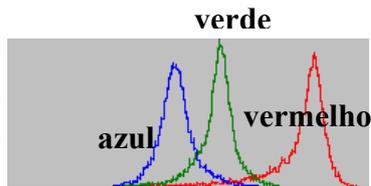


Figura 22. Histograma da região mais populosa (marrom escuro $rgb(163,113,90)$) da Figura 20.

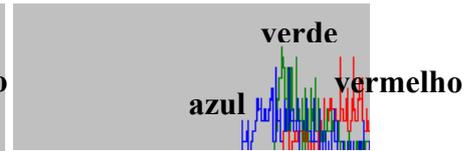


Figura 23. Histograma da cor menos populosa (cor esbranquiçada $rgb(255,252,255)$) da Figura 20.

Pode-se observar abaixo a imagem quantizada com o algoritmo *optimized-octree*. É notável sua superioridade em relação ao anterior, possuindo apenas 14 cores. Verifica-se, também, maior concentração de pontos nas regiões em relação a anterior. A cor mais freqüente da imagem abaixo, marrom escuro (valor $rgb(86,45,29)$), representa 17715 *pixels* da imagem original, com largura de espectro 63.

Já a menos freqüente, surpreendentemente, representa apenas 1 *pixel* e, obviamente, possui largura de espectro igual a 0. Vale salientar que existem outras 4 regiões que possuem até 10 *pixels*, em uma delas (cor próxima de preto, valor $rgb(88,61,64)$), o espectro é de 44. O histograma desta região é ilustrado na Figura 27. A região com maior espectro, representada pela cor marrom clara (valor $rgb(209,162,143)$), possui o valor de 127, contendo 2449 *pixels*.



Figura 24. Resultado da quantização com o *optimized-octree* da Figura 20.

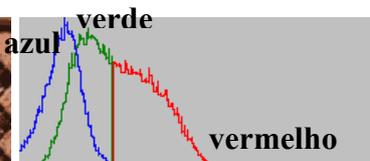


Figura 25. Histograma da região mais populosa (marrom escuro $rgb(86,45,29)$) da Figura 20.

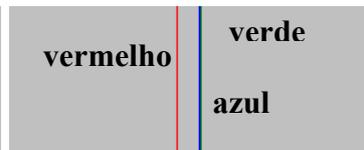


Figura 26. Histograma da cor menos populosa (cinza) da Figura 20.



Figura 27. Histograma da região pouco populosa (valor $rgb(88,61,64)$) com largura de espectro igual a 44.

Pode-se observar abaixo a Figura quantizada com o algoritmo *median-cut*, apresentando superioridade em relação aos anteriores. Constata-se uma uniformidade na distribuição de pontos dentre as regiões devido à própria formulação do algoritmo, que é dividir as regiões do cubo pela mediana.

A cor mais freqüente da imagem abaixo, marrom claro (valor $rgb(94,162,143)$), representa 5489 *pixels* da imagem original, com largura de espectro 70. Já a menos freqüente representa 4143 *pixels* (valor $rgb(205,143,108)$) e possui largura de espectro igual a 59. Observam-se nos histogramas cortes bruscos, sendo o caso dos dois histogramas apresentados abaixo na componente verde. Provavelmente, as duas regiões descritas são vizinhas quando representadas em blocos do cubo RGB.

Nota-se que se a componente verde dos dois histogramas for unida, uma gaussiana surgirá, o que é cobijado nos algoritmos de geração. Infelizmente, o *median-cut* elimina em muitos casos a gaussiana, pois a mediana da gaussiana é o seu eixo de simetria. Porém, não há como ter um algoritmo que separe a imagem em regiões, pois, o que foi observado nos azulejos através dos 3 algoritmos é a ausência de gaussianas em pelo menos 3 regiões. Um fator que contribui é que o mesmo valor das componentes pode ser usado na definição de diferentes regiões de textura da imagem.



Figura 30. Resultado da quantização com o *median-cut* da Figura 20.

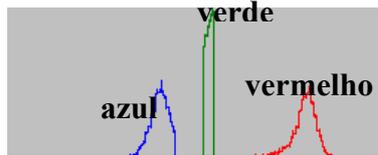


Figura 28. Histograma da região mais populosa (marrom claro #C88766) da Figura 20.

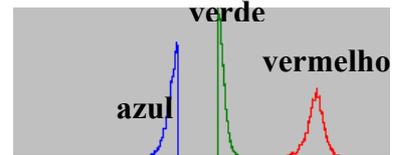


Figura 29. Histograma da região mais populosa (marrom claro #CD8F6C) da Figura 20.

Nas figuras abaixo, verifica-se o resultado da geração de textura a partir das suas imagens quantizadas. Na Figura 31 (quantizada com o *diversity*), há uma sensação de aleatoriedade, pois a região mais freqüente possui um espectro largo o suficiente para dar essa impressão. Isso não é tão perceptível na Figura 32, que foi gerada a partir do *optimized-octree*. Já a Figura 33 (imagem gerada a partir do *median-cut*) não apresenta esse fato, visto que há uma distribuição mais uniforme dentre as populações das regiões.



Figura 31. Imagem de azulejo gerada sinteticamente a partir da Figura 21.



Figura 32. Imagem de azulejo gerada sinteticamente a partir da Figura 24.



Figura 33. Imagem de azulejo gerada sinteticamente a partir da Figura 30.

3.4 Estudo sobre os formatos de imagens

Nesta seção será relatada a pesquisa para comparar as imagens em estudo nos diferentes formatos. Estas foram salvas nos formatos mais populares, o JPEG, JPEG 2000, GIF, PNG e BMP sem compressão. Sabe-se que este último é inferior aos outros em termos de tamanho de imagem, porém é útil para comparações [15].

Na literatura, o primeiro estudo entre formatos foi realizado por Lins na referência [15]. Ele separa o estudo em dois grandes grupos de formatos: com perdas e sem perdas. No primeiro, a imagem salva sofre transformações imperceptíveis ao olho humano, assegurando melhor compressão, no nosso estudo iremos focar no JPEG, JPEG-2000 e BMP de 16 milhões. Os dois primeiros apresentam melhor aproveitamento segundo esse estudo, o terceiro será utilizado para fins comparativos.

O segundo grupo de formatos parte do princípio que não há diferença alguma entre imagem salva e a original, sendo utilizado na área médica, na engenharia de documentos e na distribuição de imagens sintéticas. Nessa análise, serão objetos de estudo apenas o GIF e o PNG, que se baseiam na abordagem sem perdas em imagens com 256 cores.

Lins, no seu artigo, informa que o formato JPEG é inferior ao GIF e PNG em imagens sintéticas, em termos de qualidade, pois a imagem final sofre algumas transformações e muitas vezes perceptíveis, já que esse tipo de imagem, geralmente, apresenta muitas regiões chapadas. Sabe-se que um azulejo raramente exibe mais de 16 colorações. Portanto, uma análise é feita dentre os formatos em questão. Mais informações sobre os formatos podem ser encontradas nas referências [26] [28] [18] [19] [58] [59].

Para quantizar as imagens foi utilizado o *median-cut* com 256 e 16 cores. O resultados foram armazenado nos 4 formatos descritos anteriormente. O BMP foi salvo utilizando uma paleta de cores de tamanho equivalente a sua quantidade de cores para poder se comparar.

No início do estudo, comparar-se-á o GIF e o PNG, estabelecendo qual comprime mais a imagem. Depois, será feita uma breve análise entre o JPEG e JPEG-2000, em diversas quantidades de cores. Em seguida, uma comparação entre o que sair melhor na primeira comparação com os dois formatos da segunda.

3.4.1.1 GIF vs PNG

Primeiramente foi comparado o GIF com o BMP de 256 cores. Surpreendentemente, o primeiro só supera o segundo em 66,23% dos casos – o esperado seria o contrário em todos os casos –, registrando uma média de 92,16% de ganho do primeiro com segundo com uma variância de 40,35%. A Figura 34 ilustra uma imagem que obteve a maior taxa de compressão de 1:5 em relação a sua BMP.

O PNG superou o BMP de 256 cores em todos os casos, registrou-se uma média de 78,88% (com variância de 36,75%) do espaço ao seu concorrente. Na Figura abaixo, o PNG apresentou a maior taxa de compressão de 1:6,64 em relação ao BMP.

É notável que o fundo do azulejo é praticamente liso – pois ao se analisar a figura, observa-se uma alternância entre cores próximas de branco –, possibilitando maiores taxas de compressão, visto que ambos os formatos comprimem melhor quando carreiras de *pixels* forem mais freqüentes.

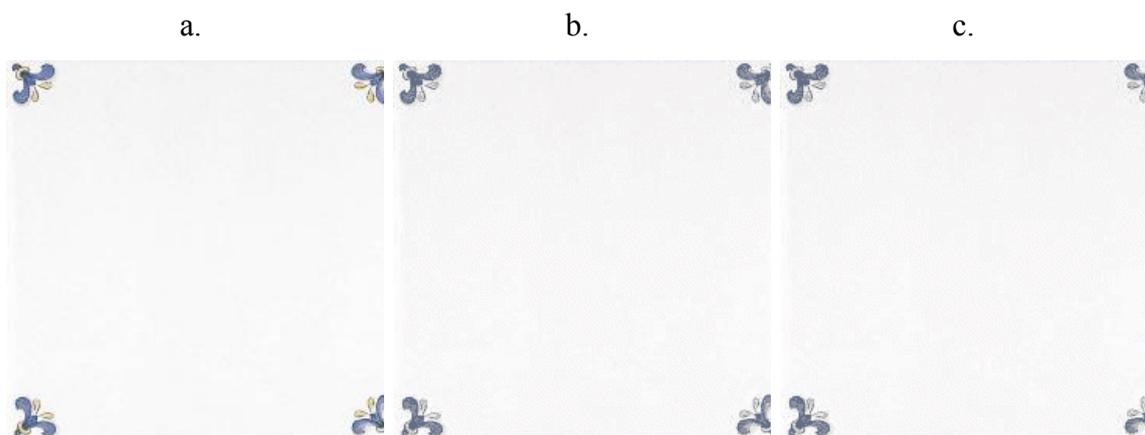


Figura 34. a. Imagem original b. Imagem quantizada em 256 cores. c. Imagem quantizada em 16 cores

A tabela abaixo demonstra os demais resultados descritos anteriormente, bem como os demais, em que o número de cores é diminuído para 16. Note que o BMP comparado com os formatos com 16 cores, é salvo com uma paleta de 16 cores, ocupando metade em relação à versão com 256.

Formatos com quantidade de cores	Superou o BMP	Espaço relativo em média	Variância
GIF com 256 cores	66,23%	92,16%	40,35%
PNG com 256 cores	100,00%	78,88%	36,75%
GIF com 16 cores	100,00%	61,71%	37,50%
PNG com 16 cores	100,00%	58,97%	35,65%

Tabela 4. Tabela contendo a comparação feita entre o PNG e GIF com o BMP

Para confirmar a superioridade do PNG, foi necessário fazer uma análise dele com o GIF. Como era de se esperar, o PNG demonstrou melhores resultados. Com 256 cores, ele ocupou em média 85,63% do espaço ocupado pelo GIF (com uma variância de 14,39%), superando-o em todas as imagens. Já com 16 cores, o PNG foi melhor em 95,81% dos casos, apresentado seu tamanho em média de 95,81%, se comparado ao seu oponente (com variância de 14,61%). A imagem em que o PNG demonstrou mais sua

superioridade diante o GIF foi a da mesma figura (Figura 34) em que estes formatos apresentaram melhor este parâmetro diante o BMP.

Com 16 cores, tanto o PNG e o GIF apresentaram a imagem com maior economia descrita na Figura 35. Porém, as duas imagens obtiveram melhor rendimento relativo dentre os dois formatos podem ser vistas na Figura 36 e na Figura 37.

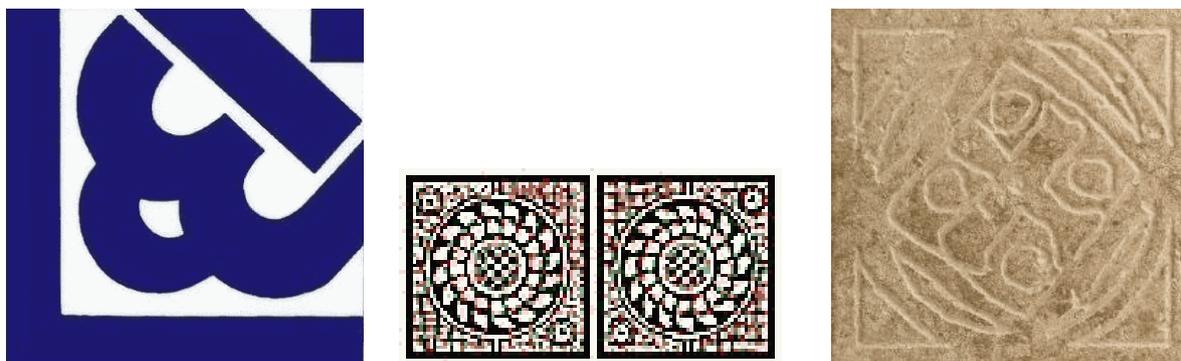


Figura 35. Figura em que o PNG e GIF com 16 cores apresentam melhor taxa de compressão. **Figura 36.** Imagem com 256 cores em que o PNG ocupa . **Figura 37.** Imagem de azulejo gerada sinteticamente a partir da Figura 24.

Já que o PNG supera o GIF em ambos os casos, é necessário fazer uma comparação entre os casos daquele com 16 e 256 cores. Dentre todos os casos, apenas duas imagens apresentaram-se, ligeiramente, mais econômicas com 256 cores que com 16. Uma delas é a ilustrada na Figura 34. A outra pode ser vista logo abaixo. Não há muitas diferenças dentre as diferentes versões da imagem abaixo. Porém, é um fato que merece destaque e que pode ser estudado no futuro.

A imagem original da Figura 34 apresenta 3052 cores distintas, ao ser quantizada para 256 cores somente 54 cores são efetivamente usadas, e ao se fazer o mesmo processo com 16 cores, apenas 15 são utilizadas. Ao ser salva em PNG com 16, ocupa-se 161,46% em relação ao arquivo com 256.

A diferença não é tão gritante no segundo caso. Essa imagem originalmente possui 5362 cores, ao se baixar para 256, 98 das suas cores são usadas. Ao diminuir para 16, todas 16 são usadas.



Figura 38. a. Imagem original b. Imagem com 256 cores c. Imagem com 16 cores

Já a imagem que apresentou melhor aproveitamento, originalmente ela possui 26374 cores. A quantização com 256 cores, resultou em 244 cores distintas, e a com 16, em 16 cores. Não há diferenças perceptíveis entre as três versões. Porém, a última apresentou um ganho de 25,43% em relação a salva em PNG com 256 cores. Observam-se diferenças relevantes dentre a imagem da Figura 39 e as outras duas descritas anteriormente, podendo servir de base para um estudo mais aprofundado.



Figura 39. a. Imagem original b. Imagem com 256 cores c. Imagem com 16 cores

3.4.2 JPEG vs JPEG-2000

Primeiramente, as imagens originais foram salvas nos formatos JPEG e JPEG-2000 com qualidades 80 e 75, respectivamente (valores próximos à média do estudo anterior).

Ao comparar as imagens geradas a partir do JPEG, verifica-se que 79,80% de todas as imagens obtiveram um menor tamanho ao se aplicar a quantização em 256 cores em contrapartida com a original. Ao se diminuir a quantidade de cores para 16, apenas 23,13% das imagens apresentaram melhora que a com 256 cores. Porém, o ganho das imagens no primeiro caso é em média de 2,07%. No segundo esse fator é negativo, sendo igual à -5,12%.

Portanto, não há justificativa para armazenar uma imagem com uma menor quantidade de cores. Esse fenômeno se deve ao fato do JPEG assumir que a imagem de entrada é uma foto, raramente possuem áreas chapadas e mudanças bruscas de cor (borda).

O JPEG-2000 com 256 cores superou o original em 50,49%, e o com 16 cores superou-o em 51,79% dos casos. No entanto, como no JPEG, o ganho nos dois casos mal ultrapassou 2%. Conclui-se então que não há razão para utilizar a imagem com pequena quantidade de cores nesse tipo de arquivo.

3.4.3 PNG vs JPEG e JPEG-2000

Diante dos fatos, é necessário apenas comparar as imagens de JPEG-2000 e JPEG com PNG de 16 cores. O PNG só superou o JPEG-2000 em 5,63% dos casos, ocupando o espaço em média de 152,0% em relação a este. Se comparado ao JPEG, o PNG supera em 38,74% dos casos. A média do espaço ocupado pelo PNG em relação ao JPEG é de 125,28%.

4 Conclusão e Trabalhos Futuros

No segundo capítulo, apresentaram-se alguns algoritmos existentes na literatura para detectar simetria e foi proposto um especialmente para imagens de azulejos. Esse algoritmo é robusto, onde apenas nos casos mais difíceis a simetria foi detectada de maneira equivocada – sendo difícil até para humanos a percepção das partes que invalidam a simetria –, tendo como taxa de acerto acima de 70%. Nenhuma das referências dos algoritmos de simetria citou algum teste feito com mais de 150 imagens, tampouco, utilizaram uma metodologia na determinação dos melhores valores de parâmetros provenientes dos algoritmos. O estudo estabeleceu os conceitos de fundo e informação – podendo ser útil em outras áreas – e na formulação de parâmetros a partir deles (fundo-fundo, fundo-informação etc). O segundo capítulo resultou num artigo científico descrito na referência [60].

Porém esse algoritmo apresenta baixo desempenho devido a dois escaneamentos da imagem e a determinação do fundo da imagem. Esta limitação pode ser superada mais facilmente através de outras estratégias.

Na primeira seção do terceiro capítulo foi descrito um estudo entre o JPEG e JPEG-2000 com as imagens de azulejos. Descobriu-se que este último não é mais econômico que o primeiro em todos os casos. Em casos específicos o JPEG apresenta melhor aproveitamento. Observou-se também os diferentes fenômenos decorrentes ao se diminuir o fator de qualidade dos dois formatos. Esse estudo comparativo é um dos poucos dentre os dois formatos realizados na literatura, e o único que faz mais comparações visuais e estatísticas em cima dela com mais de 300 imagens.

Na segunda seção foram apresentados os mais diversos algoritmos de quantização. A proposta inicial seria ao diminuir o número de cores, obter uma imagem sintética próxima a original. Na diminuição de cores, o melhor método foi o *median-cut*. Ele garante uma largura de espectro pequena, diminuindo a sensação de aleatoriedade. Porém, esse método evita a aparição de gaussianas, o que é extremamente útil em muitos algoritmos de geração de texturas.

Recomenda-se um estudo a ser realizado com algoritmos de *clustering* de imagens. Nele a pertinência de um ponto a uma determinada região da imagem não depende somente de sua cor, e sim de sua localização. As gaussianas no histograma nesse caso poderiam estar presentes, pois o valor de uma componente que define texturas distintas não estaria presente na mesma região. Vale ressaltar a outra utilidade

da detecção de regiões para profissionais (como designers, publicitários etc) que desejam extrair as texturas existentes de cada região.

Na terceira seção do terceiro capítulo, compararam-se entre si os quatro formatos mais populares: JPEG, JPEG-2000, GIF e PNG. Os dois últimos são bastante utilizados na compressão de imagens de 256 cores. Aplicou-se o *median-cut* (melhor algoritmo de quantização para a geração de texturas) sobre as imagens, salvando-se nos 4 formatos. A diminuição para 16 cores também foi feita, gravando-se nos mesmos formatos. Surpreendentemente, os dois do grupo JPEG não tiveram ganhos significativos de espaço ao diminuir sua quantidade de cores. O que não aconteceu com o GIF e PNG. Dentre os formatos, o PNG superou o GIF, mas não o JPEG e muito menos o JPEG-2000. Tal acontecimento se deve ao fato da imagem original ser uma foto, ao ser quantizada não apresenta o mesmo aspecto de uma imagem sintética, favorecendo os do grupo JPEG.

5 Referências

- [1] H. Alt, K. Mehlhorn, H. Wagener and E. Welzl. Congruence, similarity and symmetries of geometric objects. *ACM Journal of Computing*, 4:308-315, 1987.
- [2] M. Atallah. On symmetry detection. *IEEE Transactions on Computers*, 34(7):663-666, 1985.
- [3] D. Avnir and A.Y. Meyer. Quantifying the degree of molecular shape deformation: a chirality measure. *Journal of Molecular Structure (Theochem)*, 94:211-222, 1991.
- [4] S. Cavalcanti and A.Menezes e Cruz. Tiles in the Secular Architecture of Pernambuco, 19th Century, *Metalivros*, 2002.
- [5] M. Berger, *Computer Graphics with Pascal*. Addison-Wesley, 1986.
- [6] F. W. Burton, J.G. Kollins, and N.A. Alexandridis. An implementation of the exponential pyramid data structure with applications to determination of symmetries in pictures. *Computer Vision, Graphics, and Image Processing*, 25:218-225, 1984.
- [7] P. Eades. Symmetry finding algorithms, *Computational Morphology*, pp 41-51, Elsevier, 1988.
- [8] G. Gilat. Chiral coefficient – a measure of the amount of structural chirality. *Journal of Physics A: Math. Gen.*, 22:545-545, 1989.
- [9] B. Grunbaum. Measures of symmetry for convex sets. *Proc. Symp. Pure Math: American Mathematical Society*, 7:233-270, 1963.
- [10] Y. Hel_Or, S. Peleg, and D.Avnir. Characterization of right handed and left handed shapes. *Computer Vision, Graphics, and Image Processing*, 53(2), 1991.
- [11] P.T. Highnam. Optimal algorithms for finding the symmetries of planar point sets. *Information Processing Letters*, 22:219-222, 1986.
- [12] J.L. Krahe. Detection of symmetric and radial structures in images. In *International Conference on Pattern Recognition*, pp. 947-950, 1986.
- [13] J.L. Kuehnle. Symmetry-based recognition of vehicle rears. *Pattern Recognition Letters*, 12: 249—258, 1991.
- [14] T.S. Levitt. Domain independent object description and decomposition, in

- Proceedings American Association of Artificial Intelligence, pp. 207-211, 1984.
- [15] R. D. Lins and D.S.A. Machado, A Comparative Study of File Formats for Image Storage and Transmission, vol. 13(1), pp 175-183, 2004, Journal of Electronic Imaging, Jan/2004.
- [16] R. D. Lins, A New File Format for Decorative Tiles, Proceedings of ICIAR 2004, LNCS 3211, pp: 175—182, Springer Verlag, September 2004.
- [17] R. Manmatha and H.S. Sawhney. Finding Symmetry in Intensity Images, IBM Tech. Report, Almaden, 1995.
- [18] J. Miano. Compressed Image File Formats: JPEG, PNG, GIF, XBM, BMP. Addison Wesley Longman. Inc, 1999.
- [19] J.D. Murray, D. James and W. vanRyper. Encyclopedia of Graphics File Formats. O'Reilly & Associates, Inc, 1996.
- [20] R. Nevatia and T.O. Binford. Description and recognition of complex curved objects. Artificial Intelligence, 8:77-98, 1977.
- [21] H. Ogawa. Symmetry analysis of line drawings using the Hough transform. Pattern Recognition Letters, 12: 9-12, 1991.
- [22] S. K. Parui and D.D. Majumder. Symmetry analysis by computer. Pattern Recognition, 16:63-67, 1983.
- [23] J. Ponce. On characterising ribbons and finding skewed symmetries. Computer Vision, Graphics and Image Processing, 52:328-340, 1990.
- [24] S. Posch. Detecting skewed symmetries. In Inter. Conf. on Pattern Recognition, pp. 602-606, August 2002.
- [25] A. A. Vasilier. Recognition of symmetrical patterns in images. In International Conference on Pattern Recognition, pages 1138-1140, Montreal, July 1984.
- [26] G. K. Wallace. The JPEG Still Picture Compression Standard. Communication of the ACM, Volume 34, Number 4, April 1991, pp 31-44.
- [27] J. Wolter, T. Woo, and R. Volz. Optimal algorithms for symmetry detection in two and three dimensions. The Visual Computer, 1:37-48, 1985.
- [28] E.Hamilton. JPEG File Interchange Format. V 1.02, C-Cube Microsystems,

September 1992.

- [29] H. Zabrodsky, S. Peleg and D. Avnir. A measure of symmetry based on shape similarity. In IEEE Conference on Computer Vision and Pattern Recognition, pp. 703-706, June 1992.
- [30] H. Zabrodsky, S. Peleg and D. Avnir. Completion of occluded shapes using symmetry. In IEEE Conference on Computer Vision and Pattern Recognition, pages 678-679, June 1993.
- [31] H. Zabrodsky, Computational Aspects of Pattern Characterization – Continuous Symmetry, PhD Thesis, Hebrew University of Jerusalem, June 1993.
- [32] T. Zielke, M. Brauckmann, and W. von Seelen. Intensity and edge-based symmetry detection applied to car-following. In European Conference on Computer Vision, pages 865-873, May 1992.
- [33] P. W. Atkins. Physical Chemistry. Oxford University Press, Oxford, 1986.
- [34] D. W. Thompson. On Growth and Form. Cambridge Univ. Press, London, 1966.
- [35] H. T. Skin Heredity, and Malignant Neoplasms. Medical Examination Pub., Flushing, NY, 1972.
- [36] M. Jacobson. Developmental Neurobiology. Plenum Press, NY, 1978.
- [37] G. C. Evans. The Quantitative Analysis of Plant Growth. Blackwell Scientific Pub., Oxford, 1972.
- [38] S. Y. K. Yuen. Shape from contour using symmetries. In European Conference on Computer Vision, pages 437-453, Antibes, April 1990.
- [39] A. D. Gross and T. E. Boult. Syman: A symmetry analyzer. In IEEE Conference on Computer Vision and Pattern Recognition, pages 744-746, 1991.
- [40] E. Yodogawa. Symmetry, an entropy-like measure of visual symmetry. Perception and Psychophysics, 32(3): 230-240, 1982.
- [41] G. Marola. On the detection of the axes of symmetry of symmetric and almost symmetric planar images. IEEE Trans. On Pattern Analysis and Machine Intelligence, 11(1): pp. 104-108, 1989.
- [42] R. C. Gonzalez e R. E. Woods. Digital Image Processing. Addison-Wesley

Publishing Company, Estados Unidos, 1992.

- [43] A. Kruger. “Median-Cut Color Quantization – Fitting true-color images onto VGA displays,” *Dr. Dobbs’s Journal*, pp. 46–54, Sept. 1994. Disponível em <http://www.ddj.com>.
- [44] P. Heckbert. “Color Image Quantization for Frame Buffer Display,” *Computer Graphics*, vol. 16, no. 3, pp. 297–303, 1982.
- [45] J. Bradley. “XV: Interactive Image Display for the X Window System,” 1994. Disponível em: <ftp://ftp.cis.upenn.edu/pub/xv/docs/xvdocs.pdf>.
- [46] M. Gervautz e W. Purgathofer. “A simple method for color quantization: Octree quantization”. Academic Press, 1990, pp. 287–293.
- [47] D. Clark. “Color quantization using octrees”. *Dr. Dobbs’s Journal*, pp. 54-57 and 102-104, Jan. 1996.
- [48] J. Bradley. XV - Interactive Image Display for the X Windows System, apêndice F: The Diversity Algorithm, pp. 105-107. 1989.
- [49] J. Proisse. Wicked Code. *Microsoft Systems Journal*. Aug 1996. Disponível em: <http://www.microsoft.com/msj/archive/S3F1.aspx>.
- [50] C. A. B. Mello e R. D. Lins. Generating Paper Texture Using Statistical Moments. Silver Jubilee IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2000. Istanbul, Turquia. Jun 2000.
- [51] C. A. B. Mello e R. D. Lins. Colourising Paper Texture of Greyscale Historical Documents. I Congresso de Informática da Amazônia. Manaus, Brasil, Abr 2001.
- [52] K. Kamatani. Comments on “Symmetry as a continuous feature”. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 19, pp. 246–247, Fev 1995.
- [53] R. Zabrodsky, S. Peleg, and D. Avnir. Symmetry as a Continuous Feature. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 17, no. 12, pp. 1154-1166, Dez 1995.
- [54] Q. B. Sun, W. M. Huang, and J. K. Wu. Face Detection Based on Color and Local Symmetry Information. *Proceedings of Third IEEE International Conference on*

- Automatic Face and Gesture Recognition. pp. 130–135, Abr 1998.
- [55] W. Wen e B. YUAN. Skewed Symmetry Detection of Closed Contours. Based on Their Geometric Properties. Computer Architectures for Machine Perception Proceedings. pp. 161-168, Set 1995.
- [56] R. Chang e P. Hsiao. Artificial Texture Generation using Force Directed Self-Organizing Maps. IEEE International Conference on Neural Networks, IEEE World Congress on Computational Intelligence. Vol. 6, pp. 4123 - 4128, Jul 1994.
- [57] P. Parada, J. Ruiz-del-Solar, W. Plagges e M. Koppen. Interactive Texture Synthesis. 11th International Conference on Image Analysis and Processing Proceedings. pp. 434-439, Set 2001.
- [58] A. N. Skodras, C. A. Christopoulos e T. Ebrahimi. JPEG 2000: Upcoming Still Image Compression Standard. 11th Portuguese Conference on Pattern Recognition. Porto, Portugal, pp 359-366, 2000.
- [59] Shuo-yen Choo e Gregory Chew. JPEG2000 and Wavelet Compression. Disponível em: http://www.designer.com/focus/articles/web_future/web_future_1.htm.
- [60] R. D. Lins e D. M. Oliveira. Automatically Detecting Symmetries in Decorative Tiles. A ser publicado em Proceedings of ICIAR 2005, LNCS 3211, pp: 175—182, Springer Verlag, Set 2005.
- [61] Anthony Slayter-Ralph Fine Art. <http://www.anthonyslayter-ralph.com>, (visitado em 21.04.2004).
- [62] Armenian Cerâmics, <http://www.armenianceramics.com>, (visitado em 21.04.2004).
- [63] Cerâmica Portinari, <http://www.ceramicaportinari.com.br>, (visitado em 31.08.2004).
- [64] Cerâmica Fênix, <http://www.ceramicafenix.com.br>, (visitado em 04.09.2004).
- [65] Cerâmica Gytoku, <http://www.gytoku.com.br>, (visitado em 31.08.2004).
- [66] Istanbul Caravan, <http://www.istanbulcaravan.com>, (visitado em 09.11.2004).
- [67] Museu do Azulejo, <http://www.museudoazulejo.ufpe.br>, (visitado em 21.04.2004).
- [68] Nardini Cerâmica, <http://www.nardiniceramica.com.br>, (visitado em 31.08.2004).

[69] Design in Tile. <http://designsintile.com>, (visitado em 21.04.2004).

6 Datas e assinaturas

2 de Setembro de 2005

Daniel Marques Oliveira
(Aluno)

Rafael Dueire Lins
(Orientador)