

Introdução

1.1 Processamento de Imagens: breve histórico e exemplos de aplicações

A área de processamento de imagens vem sendo objeto de crescente interesse por permitir viabilizar grande número de aplicações em duas categorias bem distintas: (1) o aprimoramento de informações pictóricas para interpretação humana; e (2) a análise automática por computador de informações extraídas de uma cena. Ao longo deste livro, reservaremos a expressão 'processamento de imagens' para designar a primeira categoria, adotando os termos 'análise de imagens', 'visão por computador' (ou 'visão computacional') e 'reconhecimento de padrões' para a segunda.

Uma das primeiras aplicações na primeira categoria remonta ao começo deste século, onde buscavam-se formas de aprimorar a qualidade de impressão de imagens digitalizadas transmitidas através do sistema Bartlane de transmissão de imagens por cabo submarino entre Londres e Nova Iorque. Os primeiros sistemas Bartlane, no início da década de 20, codificavam uma imagem em cinco níveis de intensidade distintos. Esta capacidade seria expandida, já em 1929, para 15 níveis, ao mesmo tempo em que era desenvolvido um método aprimorado de revelação de filmes através de feixes de luz modulados por uma fita que continha informações codificadas sobre a imagem.

Mas o grande impulso para a área de Processamento de Imagens viria cerca de três décadas mais tarde, com o advento dos primeiros computadores digitais de grande porte e o início do programa espacial norte-americano. O uso de técnicas computacionais de aprimoramento de imagens teve início no *Jet Propulsion Laboratory* (Pasadena, Califórnia - EUA)¹ em 1964, quando imagens da lua transmitidas por uma sonda Ranger² eram processadas por computador para corrigir vários tipos de distorção inerentes à câmera de TV acoplada à sonda. Estas técnicas serviram de base para métodos aprimorados de realce e restauração de imagens de outros programas espaciais posteriores, como as expedições tripuladas da série Apollo, por exemplo.

De 1964 aos dias atuais, a área de processamento de imagens vem apresentando crescimento expressivo e suas aplicações permeiam quase todos os ramos da atividade humana. Em Medicina, o uso de imagens no diagnóstico médico tornou-se rotineiro e os avanços em processamento de imagens vêm permitindo tanto o desenvolvimento de novos equipamentos quanto a maior facilidade de interpretação de imagens produzidas por equipamentos mais antigos, como por exemplo o de raio X. Em Biologia, a capacidade de processar automaticamente imagens obtidas de microscópios, por exemplo contando o número de células de um certo tipo presentes em uma imagem, facilita sobremaneira a execução de tarefas laboratoriais com alto grau de precisão e repetibilidade. O processamento e a interpretação automática de imagens captadas por satélites auxiliam os trabalhos nas áreas de Geografia, Sensoriamento Remoto, Geoprocessamento e Meteorologia, dentre outras. Técnicas de restauração de imagens auxiliam arqueologistas a recuperar fotos borradas de artefatos raros, já destruídos. O uso de robôs dotados de visão artificial em tarefas tais como controle de qualidade em linhas de produção aumenta a cada ano, num cenário de crescente automação industrial. Inúmeras outras áreas tão distintas como Astronomia, Segurança, Publicidade e Direito – para citar apenas algumas – vêm sendo beneficiadas com os avanços nas áreas de processamento de imagens e visão por computador.

¹ "<http://www.jpl.nasa.gov>"

² "<http://www.jpl.nasa.gov/missions/ranger/>"

Leitura complementar

Mascarenhas [Mascarenhas 1990] apresenta um abrangente texto introdutório aos assuntos abordados neste livro.

1.2 Um sistema de processamento de imagens e seus componentes

Os elementos de um sistema de processamento de imagens de uso genérico são mostrados na figura 1. Este diagrama permite representar desde sistemas de baixo custo até sofisticadas estações de trabalho utilizadas em aplicações que envolvem intenso uso de imagens. Ele abrange as principais operações que se pode efetuar sobre uma imagem, a saber: aquisição, armazenamento, processamento e exibição. Além disso, uma imagem pode ser transmitida à distância utilizando meios de comunicação disponíveis. Todas estas operações são descritas a seguir.

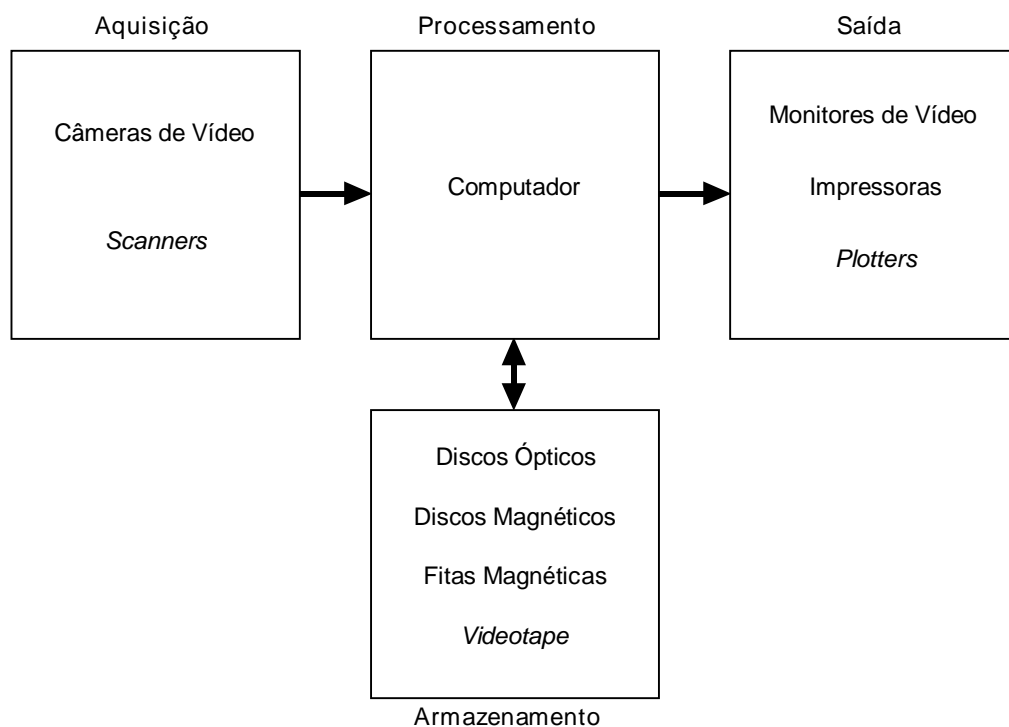


Figura 1 - Elementos de um sistema de processamento de imagens.

1.2.1 Aquisição

A etapa de aquisição tem como função converter uma imagem em uma representação numérica adequada para o processamento digital subsequente. Este bloco compreende dois elementos principais. O primeiro é um dispositivo físico sensível a uma faixa de energia no espectro eletromagnético (como raio X, ultravioleta, espectro visível ou raios infravermelhos), que produz na saída um sinal elétrico proporcional ao nível de energia detetado. O segundo – o digitalizador propriamente dito – converte o sinal elétrico analógico em informação digital, isto é, que pode ser representada através de bits 0s e 1s. Um módulo de aquisição de imagens é normalmente conhecido pelo nome de *frame grabber*. Os capítulos 2 e 7 deste livro trazem mais detalhes sobre os aspectos envolvidos na aquisição de imagens digitais.

1.2.2 Armazenamento

O armazenamento de imagens digitais é um dos maiores desafios no projeto de sistemas de processamento de imagens, em razão da grande quantidade de bytes necessários para tanto. Este armazenamento pode ser dividido em três categorias: (1) armazenamento de curta duração de uma imagem, enquanto ela é utilizada nas várias etapas do processamento, (2) armazenamento

de massa para operações de recuperação de imagens relativamente rápidas, e (3) arquivamento de imagens, para recuperação futura quando isto se fizer necessário. O espaço de armazenamento requerido é normalmente especificado em bytes (8 bits) e seus múltiplos: KB (kilobyte \approx 1000 bytes), MB (megabyte \approx 1 milhão de bytes), GB (gigabyte \approx 1 bilhão de bytes) e TB (terabyte \approx 1 trilhão de bytes). O capítulo 7 discute aspectos de armazenamento de imagens em maior profundidade.

Para o armazenamento de curta duração, a alternativa mais simples é utilizar parte da memória RAM do computador principal. Outra opção consiste no uso de placas especializadas, chamadas *frame buffers*, que armazenam uma ou mais imagens completas e podem ser acessadas a uma alta velocidade, tipicamente 30 imagens completas por segundo. O uso de *frame buffers* permite também que operações de *zoom* (ampliação ou redução para fins de visualização), *scroll* (rolagem na vertical) e *pan* (rolagem na horizontal) sejam executadas de forma praticamente instantânea. Placas *frame buffers* disponíveis no mercado atualmente apresentam capacidade de armazenamento na faixa de alguns MB de memória.

A segunda categoria de armazenamento normalmente requer o uso de discos magnéticos de no mínimo algumas centenas de MB e recentemente passou a utilizar também discos magneto-ópticos, por vezes agrupados em *jukeboxes* contendo de 30 a 100 discos. Nesta categoria o fator 'tempo de acesso' é tão ou mais importante que a capacidade (em bytes) do meio de armazenamento. Através de cálculos simples (n° de pixels na horizontal \times n° de pixels na vertical \times n° de bits necessários para a escala de cinza / 8), pode-se estimar a quantidade de bytes necessários para armazenar uma imagem monocromática em disco. Este cálculo entretanto considera uma imagem representada como uma matriz, cujos elementos são os valores de tons de cinza dos respectivos pixels.³ Na prática, informações adicionais (tamanho da imagem e número de cores ou tons de cinza, no mínimo) são necessárias. Estas informações costumam ser colocadas em um cabeçalho (*header*) no início do arquivo. Infelizmente, não existe um único cabeçalho ou formato de armazenamento de imagens padronizados. Alguns dos formatos mais comuns são o BMP, PCX, TIFF, JPEG e GIF. Estes formatos de arquivos de imagem, além de muitos outros, são apresentados no Apêndice A.

Finalmente, o arquivamento de imagens é caracterizado por quantidades gigantescas de bytes contendo imagens cuja recuperação é esporádica. Nesta categoria, as fitas magnéticas estão dando lugar aos discos ópticos WORM (*Write-Once-Read-Many*), com capacidade que pode chegar a mais de 10 GB por disco, e que também podem ser agrupados em *jukeboxes*, com capacidade total de armazenamento superior a 1 TB.

1.2.3 Processamento

O processamento de imagens digitais envolve procedimentos normalmente expressos sob forma algorítmica. Em função disto, com exceção das etapas de aquisição e exibição, a maioria das funções de processamento de imagens pode ser implementada via software. O uso de hardware especializado para processamento de imagens somente será necessário em situações nas quais certas limitações do computador principal (por exemplo, velocidade de transferência dos dados através do barramento) forem intoleráveis.

A tendência atual do mercado de hardware para processamento de imagens é a comercialização de placas genéricas compatíveis com os padrões de barramento consagrados pelas arquiteturas mais populares de microcomputadores e estações de trabalho. O software de controle destas placas é que determinará sua aplicação específica a cada situação. As vantagens mais imediatas são: redução de custo, modularidade, reutilização de componentes de software em outra aplicação rodando sobre o mesmo hardware e independência de fornecedor. Convém notar, entretanto, que sistemas dedicados continuam sendo produzidos e comercializados para atender a tarefas específicas, tais como processamento de imagens transmitidas por satélites.

³ Para imagens coloridas, a situação é um pouco mais complexa. Normalmente estes números serão índices (endereços) de uma tabela de cores, denominada palheta ou *palette*. Para maiores detalhes, veja a seção 4.6 e o Resumo da Teoria da Prática de Laboratório nº 1.

1.2.4 Transmissão

Imagens digitalizadas podem ser transmitidas à distância utilizando redes de computadores e protocolos de comunicação já existentes. O grande desafio da transmissão de imagens à distância é a grande quantidade de bytes que se necessita transferir de uma localidade a outra, muitas vezes através de canais de comunicação de baixa velocidade e banda passante estreita. Este problema é ainda mais sério quando se deseja transmitir seqüências de vídeo (imagens em movimento com áudio associado) em tempo real, onde outros fatores, como por exemplo sincronização, devem ser considerados. Nestes casos, o uso de técnicas de compressão e descompressão de imagens, como as descritas no capítulo 6, é mandatório.

1.2.5 Exibição

O monitor de vídeo é um elemento fundamental de um sistema de processamento de imagens. Os monitores em uso atualmente são capazes de exibir imagens com resolução de pelo menos 640 x 480 pixels com 256 cores distintas. A tecnologia mais usual ainda é o TRC (Tubo de Raios Catódicos).

Um TRC para um sistema de processamento de imagens normalmente segue um padrão de vídeo. O padrão de vídeo mais comum para sistemas monocromáticos é o RS-170. Ele prevê 480 linhas horizontais entrelaçadas, isto é, a varredura de um quadro é feita em duas etapas, abrangendo primeiramente as linhas ímpares e posteriormente as linhas pares. Cada uma destas etapas é denominada campo. O tempo necessário para percorrer um campo é 1/60 s; conseqüentemente, o tempo total de um quadro é 1/30 s. As características de persistência visual do olho humano fazem com que, nesta velocidade, a varredura individual de cada campo não seja perceptível, bem como dão a impressão de que a seqüência de quadros explorados é perfeitamente contínua.

O padrão RS-170 especifica resolução vertical de 480 pixels, sendo a resolução horizontal determinada pelos circuitos eletrônicos do monitor e pelo tamanho dos pontos de fósforo na tela. Este número costuma ser 512, proporcionando imagens de 512 x 480 pixels.

A resolução espacial dos monitores é normalmente especificada em pontos por polegada (*dots per inch - dpi*). Um valor típico de resolução é 72 dpi, suficiente para exibir uma imagem de 1024 x 1024 pixels em um monitor de 19 polegadas ou uma imagem de 640 x 400 pontos em uma tela cuja diagonal meça 12 polegadas. A título de comparação, uma tela de TV tem resolução na faixa de 40 dpi.

Um TRC colorido difere radicalmente de seu antecessor monocromático, por apresentar três feixes eletrônicos, cada um correspondente a uma das três cores primárias (vermelho, verde e azul). A superfície interna da tela é constituída por três tipos de fósforo, dispostos de forma triangular, cada qual sensível a uma das cores primárias e excitado pelo respectivo canhão eletrônico. Isto significa dizer que, do ponto de vista construtivo, cada pixel é na verdade uma combinação de três pequenos pixels, um para cada cor primária.

A indústria de dispositivos de exibição vem apresentando sistematicamente novas tecnologias de fabricação de monitores de vídeo, dentre eles os monitores de cristal líquido (LCD), cada vez mais populares graças à disseminação dos computadores portáteis (*notebooks*).

Existem diversas formas de reprodução de imagens em papel. A melhor, e mais cara, é a reprodução fotográfica, onde o número de gradações de cinza é função da densidade dos grânulos de prata no papel. Outra possibilidade é o uso de papel sensível a temperatura, cuja composição química faz com que ele apresente coloração mais escura à medida que a temperatura aumenta. Este tipo de impressão ainda é o mais difundido em equipamentos de fax. Uma de suas desvantagens é o desvanecimento das imagens com o tempo. Nos últimos anos aumentou consideravelmente a oferta de impressoras térmicas coloridas no mercado. Estas impressoras baseiam-se na deposição de cera colorida sobre um papel especial para produzir a impressão. O capítulo 7 apresenta maiores informações sobre dispositivos de exibição e impressão de imagens.

Dispositivos periféricos de saída especializados na produção de cópias da imagem em forma de fotografias, slides ou transparências também estão se tornando cada vez mais usuais.

Uma alternativa às técnicas fotográficas consiste no uso de técnicas de *halftoning*. É o método usado por jornais e por impressoras convencionais (laser, matriciais ou a jato de tinta) para a impressão de imagens. Esta técnica consiste basicamente em imprimir pontos escuros de diferentes tamanhos, espaçados de tal maneira a reproduzir a ilusão de tons de cinza. À medida que a distância entre o observador e a imagem impressa aumentam, os detalhes finos vão desaparecendo e a imagem parece cada vez mais uma imagem contínua monocromática.

No jargão computacional, dá-se o nome de *dithering* ao processo de produção do efeito de *halftoning*, bem como a todas as técnicas de conversão de uma imagem para adaptá-la a resoluções menores, tanto para efeito de exibição como para impressão. Existem vários algoritmos de *dithering*, sendo o mais comum o de Floyd-Steinberg, que consiste de um processo adaptativo no qual o padrão de *dither* a ser atribuído a um pixel depende de seu tom de cinza e de seus vizinhos. Uma discussão mais detalhada destes algoritmos foge ao escopo deste livro.

Leitura complementar

O capítulo 7 de [Lindley 1991] e o capítulo 11 de [Rimmer 1993] trazem explicações teóricas e código-fonte em C para impressão de imagens monocromáticas e/ou coloridas utilizando *dithering*.

O capítulo 10 de [Dougherty 1994] é uma excelente referência para um estudo mais aprofundado dos conceitos e técnicas de *halftoning*.

1.3 O sistema visual humano

A figura 2 mostra um corte horizontal do olho humano. O globo ocular tem formato aproximadamente esférico e um diâmetro de cerca de 20 mm. Ele é envolvido por três membranas: a camada externa formada pela córnea e pela esclerótica, a coróide e a retina, que é a camada interna. A córnea é uma película transparente que cobre a parte anterior do olho. Dando continuidade à córnea, a esclerótica é uma membrana opaca que reveste o globo ocular.

A coróide está situada abaixo da esclerótica. Essa membrana contém uma rede de vasos sanguíneos que servem como a principal fonte de nutrição do olho. O revestimento da coróide é fortemente pigmentado, o que ajuda a reduzir a quantidade de luz que entra no olho. Ela é dividida em corpo ciliar e diafragma da íris, sendo este último responsável por controlar a quantidade de luz que deve penetrar no olho. O diâmetro da abertura central da íris (pupila) varia entre 2 mm e 8 mm. A parte frontal da íris contém o pigmento visível do olho, enquanto sua porção posterior possui um pigmento negro.

A membrana mais interna do olho é a retina, situada na sua parede posterior. Quando o olho focaliza uma cena, a imagem correspondente é projetada sobre a retina, na qual estão distribuídos dois tipos de receptores de luz discretos: os cones e os bastonetes. Os cones são em número de 6 a 7 milhões em cada olho e estão localizados na porção central da retina, chamada de fóvea. Eles são altamente sensíveis a cor e cada qual está conectado a uma terminação nervosa dedicada. O número de bastonetes é bastante maior, cerca de 75 a 150 milhões, distribuídos em toda a superfície da retina. Os bastonetes servem para dar uma visão geral da imagem captada no campo de visão. Eles não distinguem cores, mas são sensíveis a baixos níveis de iluminação.

A fóvea é uma reentrância circular na retina com aproximadamente 1,5 mm de diâmetro. De forma aproximada, podemos considerá-la um sensor de área quadrada de 1,5 mm por 1,5 mm. A densidade de cones nesta área da retina é de aproximadamente 150.000 elementos por mm^2 . Baseando-se nessas aproximações, o número de cones na região de maior acuidade do olho é de aproximadamente 337.000 elementos. Para efeito comparativo, esta resolução é facilmente alcançada pela tecnologia atual, usando CCD de área não superior a 7 mm x 7 mm.

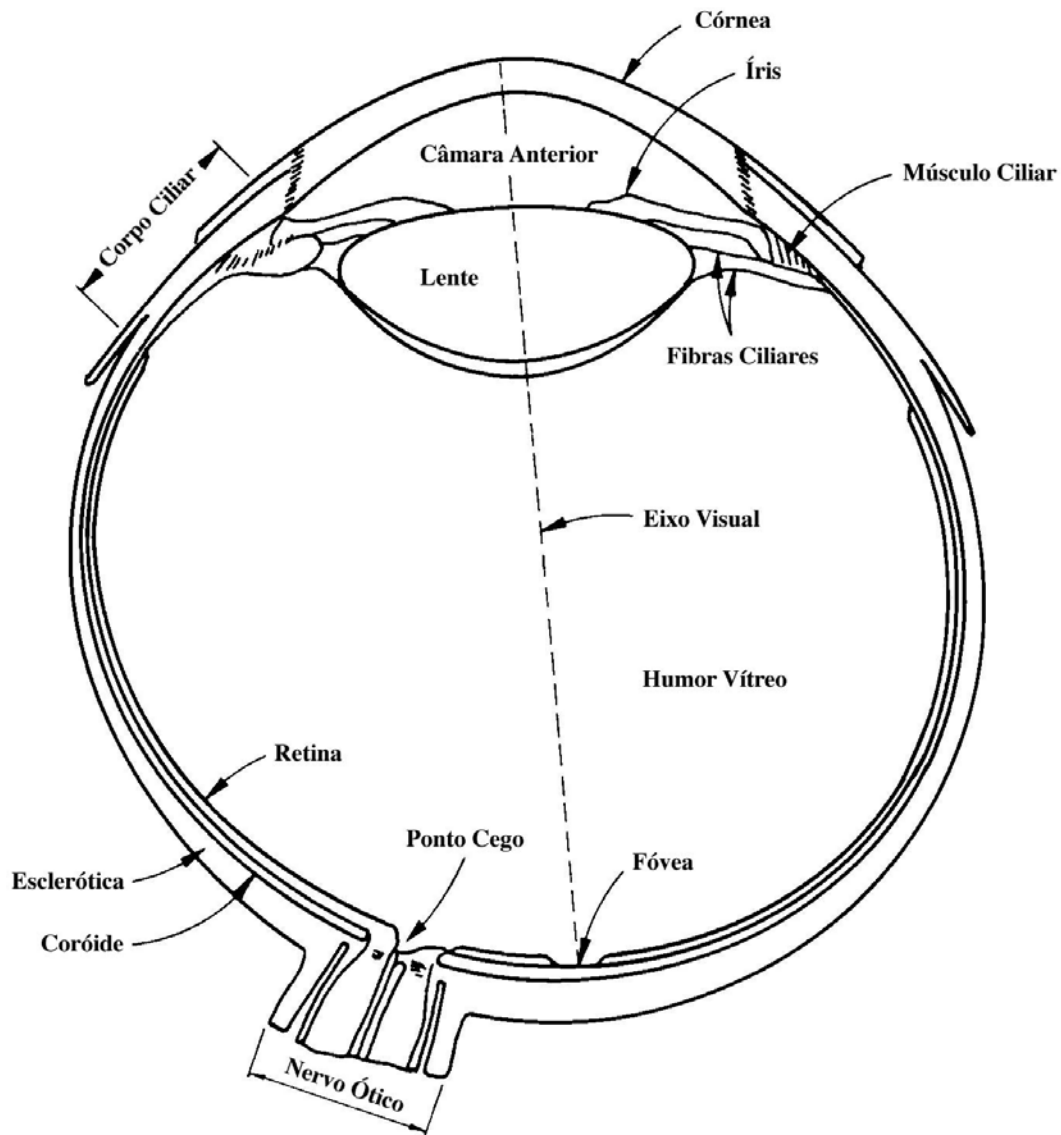


Figura 2 - Vista em corte do olho humano (adaptado de Gonzalez, R.C., Woods, R.E., *Digital Image Processing*, Addison-Wesley, 1992).

Dentre as características do sistema visual humano relevantes para o projeto de sistemas de visão artificial, podemos destacar a enorme faixa de valores de intensidade luminosa (da ordem de 10^{10}) aos quais o olho pode se adaptar e a relação aproximadamente logarítmica entre os níveis de intensidade luminosa presentes na cena e a sensação subjetiva de brilho, ilustrada na figura 3. Convém notar, entretanto, que o olho humano não pode operar sobre toda esta faixa simultaneamente. Ao invés disso, ele excursiona ao longo de toda esta faixa através de adaptações em sua sensibilidade global, um fenômeno conhecido como adaptação de brilho. A faixa total de valores de intensidade que o olho pode discriminar simultaneamente é comparativamente pequena em relação à faixa total de adaptação. Para um certo conjunto de condições, o nível de sensibilidade atual do sistema visual é chamado nível de adaptação de brilho, como por exemplo o valor B_a na figura 3. A pequena porção de curva que intercepta a curva principal representa a faixa de brilho subjetivo que o olho pode perceber quando adaptado a este nível. Esta faixa é bastante restrita, existindo um nível de brilho B_b abaixo do qual todos os estímulos são indistinguíveis. O trecho tracejado da curva não é, na verdade, restrito, mas se estendido além de um limite perde seu sentido, porque neste caso o olho humano buscaria outro nível de adaptação maior que B_a .

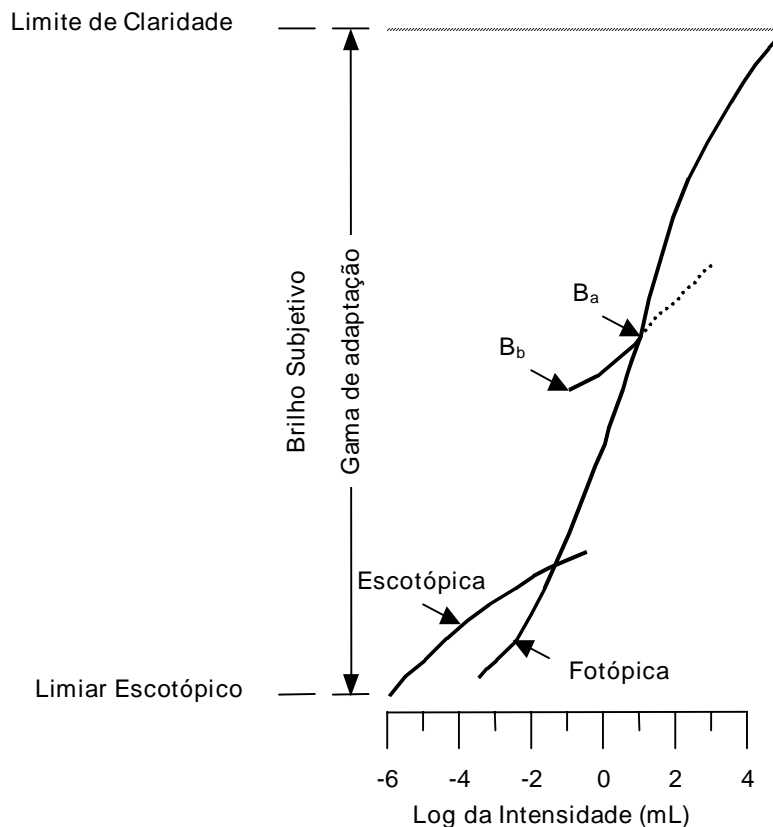


Figura 3 - Curva que relaciona a sensação subjetiva de brilho com o logaritmo da intensidade luminosa incidente sobre o olho humano.

A tabela 1 apresenta uma visão comparativa entre o sistema visual humano e um sistema de visão artificial.

Leitura complementar

Para maiores informações sobre aspectos do processo de percepção visual humana e suas implicações no projeto de sistemas de visão artificial, uma referência obrigatória é [Marr 1982].

A seção 2.1 de [Gonzalez e Woods 1992] apresenta interessantes fenômenos de percepção visual tais como contraste simultâneo, discriminação de brilho usando a razão de Weber e o padrão de faixas de Mach.

1.4 Sistemas de Visão Artificial: fundamentos e desafios

Nas primeiras páginas de seu livro *Digital Image Processing and Computer Vision*, Robert J. Schalkoff [Schalkoff 1989] formula uma sugestiva pergunta quando questiona: "O que estamos tentando fazer e por que isto é tão difícil?"

Uma resposta simplista à primeira parte da pergunta acima é: "Estamos tentando ensinar robôs a enxergar." Ao relacionarmos as dificuldades inerentes ao processo de dotar o computador de uma capacidade visual semelhante à dos seres humanos, deparamo-nos com três admiráveis características do processo de percepção visual humano, que são:

- uma base de dados muito rica;
- altíssima velocidade de processamento; e
- a capacidade de trabalhar sob condições muito variadas.

Os avanços na tecnologia de dispositivos de armazenamento de massa e o surgimento de novas CPUs e arquiteturas computacionais cada vez mais rápidas, com alto grau de paralelismo, nos induzem a crer que dispomos de condições cada vez melhores de modelar as duas primeiras características relacionadas acima. O grande desafio permanece sendo o de fazer com que os sistemas de visão artificial trabalhem em diferentes condições de luminosidade, contraste,

posicionamento relativo dos objetos em uma cena sem perder a capacidade de interpretar a cena, de forma análoga a nossa capacidade de reconhecer um amigo ou parente com relativa facilidade, independentemente de ele estar usando óculos ou não, ter deixado crescer a barba ou estar no carro ao lado do nosso em uma esquina num final de tarde, onde não dispomos de outra imagem senão a vista de perfil e onde as condições de luminosidade são bastante inferiores às que obteríamos ao meio-dia.

Tabela 1 - Comparação entre o sistema visual humano e um sistema de visão artificial.

	Sistema visual humano	Sistema de visão artificial
Espectro	Limitado à faixa de luz visível (300 nm a 700 nm) do espectro de ondas eletromagnéticas.	Pode operar em praticamente todo o espectro de radiações eletromagnéticas, dos raios X ao infravermelho.
Flexibilidade	Extremamente flexível, capaz de se adaptar a diferentes tarefas e condições de trabalho.	Normalmente inflexível, apresenta bom desempenho somente na tarefa para a qual foi projetado.
Habilidade	Pode estabelecer estimativas relativamente precisas em assuntos subjetivos.	Pode efetuar medições exatas, baseadas em contagem de pixels e, portanto, dependentes da resolução da imagem digitalizada.
Cor	Possui capacidade de interpretação subjetiva de cores.	Mede objetivamente os valores das componentes R, G e B para determinação de cor.
Sensibilidade	Capaz de se adaptar a diferentes condições de luminosidade, características físicas da superfície do objeto e distância ao objeto. Limitado na distinção de muitos níveis diferentes de cinza, simultaneamente.	Sensível ao nível e padrão de iluminação, bem como à distância em relação ao objeto e suas características físicas. Pode trabalhar com centenas de tons de cinza, conforme projeto do digitalizador.
Tempo de resposta	Elevado, da ordem de 0,1 s.	Dependente de aspectos de hardware, podendo ser tão baixo quanto 0,001 s.
2-D e 3-D	Pode executar tarefas 3-D e com múltiplos comprimentos de onda (dentro do espectro de luz visível) facilmente.	Executa tarefas 2-D com relativa facilidade, mas é lento e limitado em tarefas 3-D.
Percepção	Percebe variações de brilho em escala logarítmica. A interpretação subjetiva de brilho depende da área ao redor do objeto considerado.	Pode perceber brilho em escala linear ou logarítmica.

1.4.1 Estrutura de um Sistema de Visão Artificial

Definiremos um Sistema de Visão Artificial (SVA) como um sistema computadorizado capaz de adquirir, processar e interpretar imagens correspondentes a cenas reais. A figura 4 mostra esquematicamente um diagrama de blocos de um SVA. Suas principais etapas são explicadas a seguir, partindo da premissa de que um problema prático, por exemplo a leitura do Código de Endereçamento Postal (CEP) de um lote de envelopes, deve ser solucionado.

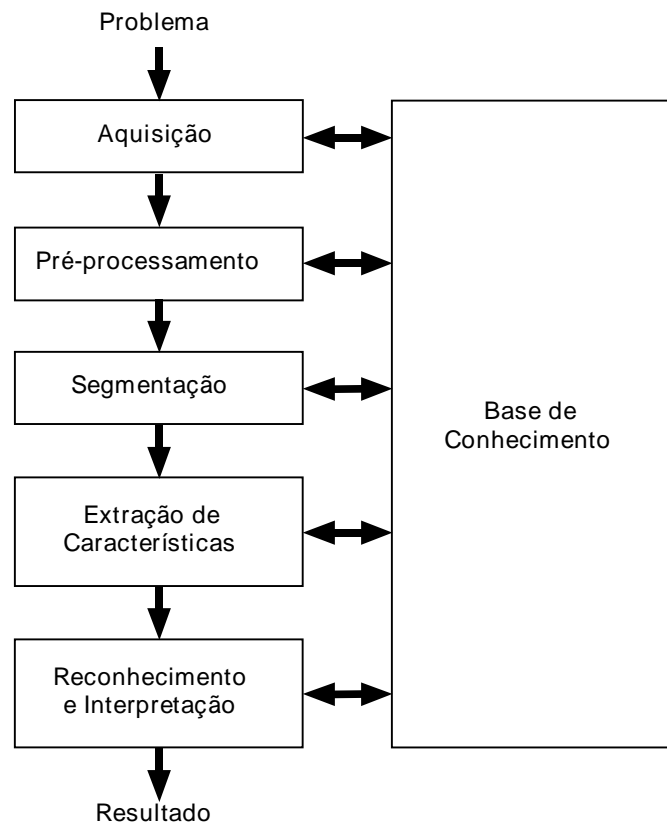


Figura 4 - Um Sistema de Visão Artificial (SVA) e suas principais etapas.

1.4.2 Domínio do problema e resultado

O domínio do problema, neste caso, consiste no lote de envelopes e o objetivo do SVA é ler o CEP presente em cada um deles. Logo, o resultado esperado é uma seqüência de dígitos correspondentes ao CEP lido.

1.4.3 Aquisição da imagem

O primeiro passo no processo é a aquisição de imagens dos envelopes. Para tanto são necessários um sensor e um digitalizador. O sensor converterá a informação óptica em sinal elétrico e o digitalizador transformará a imagem analógica em imagem digital.

Dentre os aspectos de projeto envolvidos nesta etapa, pode-se mencionar: a escolha do tipo de sensor, o conjunto de lentes a utilizar, as condições de iluminação da cena, os requisitos de velocidade de aquisição (assumindo que os envelopes passariam pela frente da câmera a uma certa velocidade), a resolução e o número de níveis de cinza da imagem digitalizada, dentre outros. Esta etapa produz à saída uma imagem digitalizada do envelope.

1.4.4 Pré-processamento

A imagem resultante do passo anterior pode apresentar diversas imperfeições, tais como: presença de pixels ruidosos, contraste e/ou brilho inadequado, caracteres (especialmente os dígitos do CEP) interrompidos ou indevidamente conectados etc. A função da etapa de pré-

processamento é aprimorar a qualidade da imagem para as etapas subseqüentes. As operações efetuadas nesta etapa são ditas de baixo nível porque trabalham diretamente com os valores de intensidade dos pixels, sem nenhum conhecimento sobre quais deles pertencem aos dígitos do CEP, a outras informações impressas no envelope ou ao fundo. A imagem resultante desta etapa é uma imagem digitalizada de melhor qualidade que a original.

1.4.5 Segmentação

A tarefa básica da etapa de segmentação é a de dividir uma imagem em suas unidades significativas, ou seja, nos objetos de interesse que a compõem. Esta tarefa, apesar de simples de descrever, é das mais difíceis de implementar.

No caso específico do problema do CEP, é possível que o problema seja dividido em duas etapas: em um primeiro momento os algoritmos de segmentação tentarão localizar o CEP do restante das informações para posteriormente, trabalhando sobre esta subimagem, segmentar cada dígito individualmente. Segundo esta linha de raciocínio, este bloco produzirá à saída oito subimagens, cada qual correspondendo a um dígito do CEP.

1.4.6 Extração de Características

Esta etapa procura extrair características das imagens resultantes da segmentação através de descritores que permitam caracterizar com precisão cada dígito e que apresentem bom poder de discriminação entre dígitos parecidos, como o '5' e o '6'. Estes descritores devem ser representados por uma estrutura de dados adequada ao algoritmo de reconhecimento. É importante observar que nesta etapa a entrada ainda é uma imagem, mas a saída é um conjunto de dados correspondentes àquela imagem.

Para maior clareza, suponhamos que os descritores utilizados para descrever um caractere sejam as coordenadas normalizadas x e y de seu centro de gravidade e a razão entre sua altura e largura. Neste caso, um vetor de três elementos é uma estrutura de dados adequada para armazenar estas informações sobre cada dígito processado por esta etapa.

1.4.7 Reconhecimento e Interpretação

Nesta última etapa do sistema, denominamos reconhecimento o processo de atribuição de um rótulo a um objeto baseado em suas características, traduzidas por seus descritores. A tarefa de interpretação, por outro lado, consiste em atribuir significado a um conjunto de objetos reconhecidos. Neste exemplo, uma forma simples de interpretação seria a verificação do CEP em uma base de dados de CEPs válidos, para descobrir se o conjunto dos oito caracteres (cada qual reconhecido individualmente) faz sentido ou não.

1.4.8 Base de Conhecimento

Todas as tarefas das etapas descritas acima pressupõem a existência de um conhecimento sobre o problema a ser resolvido, armazenado em uma base de conhecimento, cujo tamanho e complexidade podem variar enormemente. Idealmente, esta base de conhecimento deveria não somente guiar o funcionamento de cada etapa, mas também permitir a realimentação entre elas. Por exemplo, se a etapa de representação e descrição recebesse 7 caracteres ao invés de 8, ela deveria ser capaz de realimentar a etapa de segmentação (provável responsável pela falha) para que esta procurasse segmentar novamente a subimagem 'suspeita' (aquela de maior largura), buscando dividi-la em duas. Esta integração entre as várias etapas através da base de conhecimento ainda é um objetivo difícil de alcançar e não está presente na maioria dos SVAs existentes atualmente.

Finalmente, cumpre observar que nem todos os SVAs possuem todos estes blocos e que a maioria das técnicas descritas neste livro estão delimitadas até o bloco 'pré-processamento'.

Fundamentos de Imagens Digitais

Este capítulo tem por objetivo apresentar as principais características das imagens digitais. A primeira seção caracteriza uma imagem do ponto de vista matemático e descreve o processo de aquisição de imagens e sua conversão para o formato digital. A seção 2.2 descreve as principais propriedades de uma imagem digital, padronizando a terminologia a ser utilizada no restante do livro. As principais operações lógicas e aritméticas sobre imagens são exemplificadas na seção 2.3. A seção 2.4 introduz o importante conceito de convolução com máscaras e fornece exemplos de máscaras úteis para enfatizar e/ou detectar propriedades de uma imagem. Finalmente, a seção 2.5 trata das transformações geométricas que podem ser aplicadas a imagens, fundamentando-as matematicamente e ilustrando-as com exemplos.

2.1 Aquisição e digitalização de imagens

Uma imagem monocromática pode ser descrita matematicamente por uma função $f(x,y)$ da intensidade luminosa, sendo seu valor, em qualquer ponto de coordenadas espaciais (x,y) , proporcional ao brilho (ou nível de cinza) da imagem naquele ponto. A figura 1 mostra uma imagem monocromática e a convenção utilizada neste livro para o par de eixos (x,y) ¹.

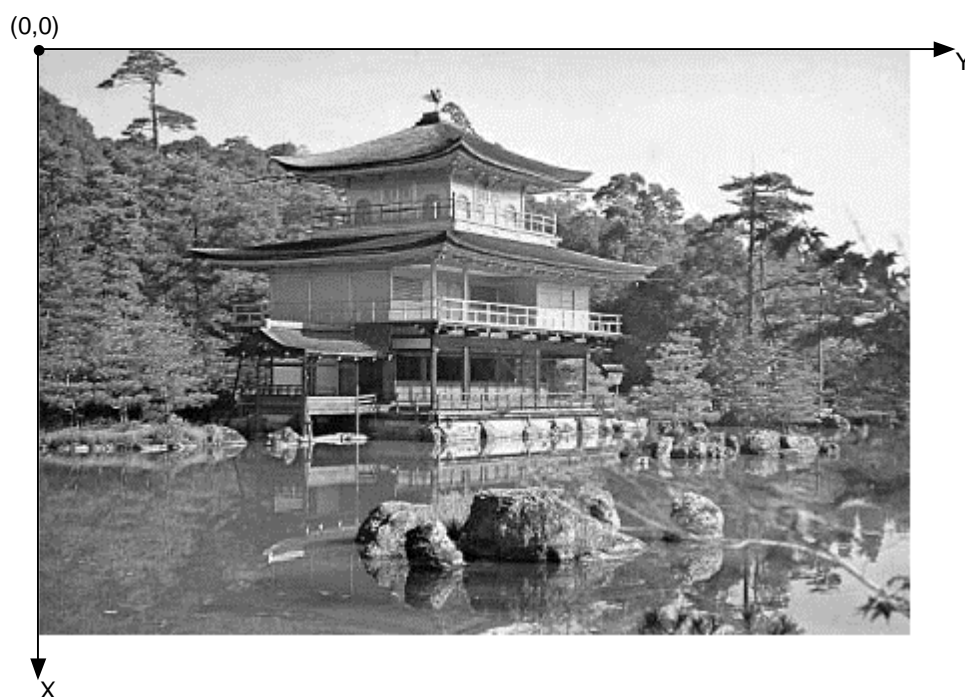


Figura 1 - Uma imagem monocromática e a convenção utilizada para o par de eixos (x,y) .

A função $f(x,y)$ representa o produto da interação entre a iluminância $i(x,y)$ – que exprime a quantidade de luz que incide sobre o objeto – e as propriedades de refletância ou de transmitância próprias do objeto, que podem ser representadas pela função $r(x,y)$, cujo valor

¹ Como o leitor deve ter notado, a posição e a direção dos eixos x e y são diferentes das utilizadas na Geometria Analítica. Em Processamento de Imagens, a notação (x,y) pode ser entendida como (*linha*, *coluna*). Convém observar que esta notação não está padronizada na literatura técnica da área.

exprime a fração de luz incidente que o objeto vai transmitir ou refletir ao ponto (x,y) . Estes conceitos estão ilustrados na figura 2. Matematicamente:

$$f(x, y) = i(x, y) \cdot r(x, y) \quad (2.1)$$

com:

$$0 < i(x, y) < \infty \quad \text{e} \\ 0 < r(x, y) < 1$$

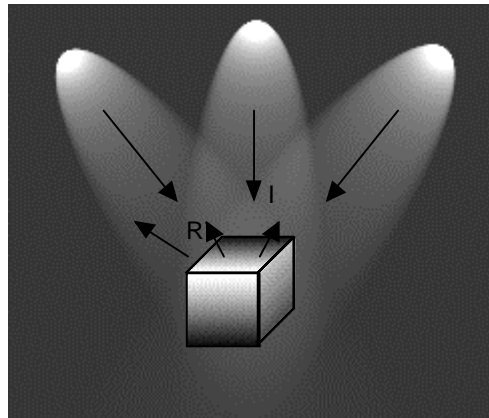


Figura 2 - Os componentes iluminância (I) e refletância (R) de uma imagem.

As tabelas 1 e 2 apresentam valores típicos de iluminância e refletância.

Tabela 1 - Exemplos de valores para $i(x,y)$ [em lux ou lúmen/m²]

$i(x,y)$	
900	dia ensolarado
100	dia nublado
10	iluminação média de escritório
0,001	noite clara de lua cheia

Tabela 2 - Exemplos de valores para $r(x,y)$

$r(x,y)$	
0,93	neve
0,80	parede branco-fosca
0,65	aço inoxidável
0,01	veludo preto

Ao longo deste livro, a intensidade de uma imagem monocromática f nas coordenadas (x,y) será denominada nível de cinza (ou tom de cinza) (L) da imagem naquele ponto. Este valor estará no intervalo:

$$L_{\min} \leq L \leq L_{\max}$$

sendo L_{\min} e L_{\max} valores positivos e finitos.

O intervalo $[L_{\min}, L_{\max}]$ é denominado escala de cinza da imagem. É comum deslocar este intervalo numericamente para o intervalo dos inteiros $[0, W)$, onde $L = 0$ significa pixel preto e $L = W-1$ representa pixel branco. Normalmente, W é uma potência inteira positiva de 2.

No caso de uma imagem que possui informações em intervalos ou bandas distintas de frequência, é necessário uma função $f(x,y)$ para cada banda. É o caso de imagens coloridas padrão RGB, que são formadas pela informação de cores primárias aditivas, como o vermelho (*R - Red*), verde (*G - Green*) e azul (*B - Blue*). A seção 4.6 apresenta informações adicionais sobre imagens coloridas.

As técnicas de processamento de imagens descritas neste livro trabalham fundamentalmente com imagens monocromáticas, bidimensionais e estáticas. Para que uma imagem seja processada por alguma destas técnicas, é fundamental representar sua informação num formato adequado ao tratamento computacional, por exemplo, uma matriz de números inteiros não-negativos, cujos valores referenciam o brilho médio amostrado no ponto correspondente da cena.

Para converter uma cena real em uma imagem digitalizada, duas etapas são imprescindíveis: a aquisição da imagem e sua digitalização.

2.1.1 Aquisição

Chamaremos de aquisição de uma imagem o processo de conversão de uma cena real tridimensional em uma imagem analógica, ou seja, delimitaremos esta etapa ao processo de transdução optoeletrônica.

O primeiro passo na conversão de uma cena real tridimensional em uma imagem eletrônica é a redução de dimensionalidade. Assumiremos que uma câmera fotográfica, câmera de vídeo ou outro dispositivo converterá a cena 3-D em uma representação 2-D adequada, sem nos preocuparmos com as questões envolvidas nesta etapa.

O dispositivo de aquisição de imagens mais utilizado atualmente é a câmera CCD (*Charge Coupled Device*). Ela consiste de uma matriz de células semicondutoras fotossensíveis, que atuam como capacitores, armazenando carga elétrica proporcional à energia luminosa incidente. O sinal elétrico produzido é condicionado por circuitos eletrônicos especializados, produzindo à saída um Sinal Composto de Vídeo (SCV) analógico e monocromático.

Para a aquisição de imagens coloridas utilizando CCDs é necessário um conjunto de prismas e filtros de cor encarregados de decompor a imagem colorida em suas componentes R, G e B, cada qual capturada por um CCD independente. Os sinais elétricos correspondentes a cada componente são combinados posteriormente conforme o padrão de cor utilizado (NTSC (*National Television Standards Committee*) ou PAL (*Phase Alternating Line*), por exemplo). Uma câmera CCD monocromática simples consiste basicamente de um conjunto de lentes que focalizarão a imagem sobre a área fotossensível do CCD, o sensor CCD e seus circuitos complementares. A figura 3 mostra uma visão simplificada da aquisição de imagens com câmera CCD. O capítulo 7 traz maiores detalhes sobre sensores de imagem.

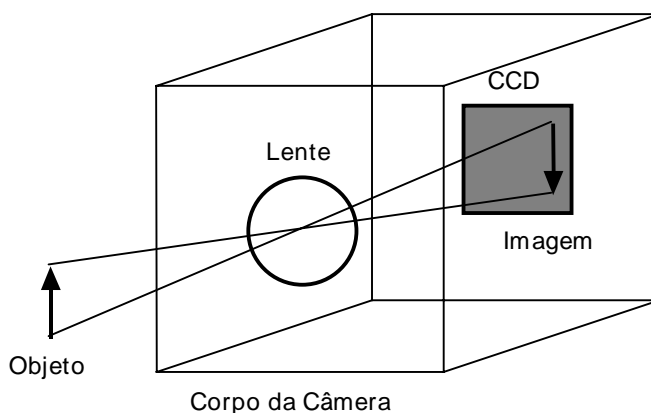


Figura 3 - Visão esquemática de uma câmera CCD.

2.1.2 Digitalização

O sinal analógico de vídeo obtido à saída do dispositivo de aquisição deve ser submetido a uma discretização espacial e em amplitude para tomar o formato desejável ao processamento computacional.

Chamaremos de amostragem o processo de discretização espacial e daremos o nome de quantização ao processo de discretização em amplitude.

Basicamente, a amostragem converte a imagem analógica em uma matriz de M por N pontos, cada qual denominado pixel (ou elemento de imagem):

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0, N-1) \\ f(1,0) & f(1,1) & \dots & f(1, N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1, N-1) \end{bmatrix} \quad (2.2)$$

Maiores valores de M e N implicam em uma imagem de maior resolução.

Por seu lado, a quantização faz com que cada um destes pixels assuma um valor inteiro, na faixa de 0 a 2^n-1 . Quanto maior o valor de n , maior o número de níveis de cinza presentes na imagem digitalizada.

Do ponto de vista eletrônico, a digitalização consiste em uma conversão analógico-digital na qual o número de amostras do sinal contínuo por unidade de tempo indica a taxa de amostragem e o número de bits do conversor A/D utilizado determina o número de tons de cinza resultantes na imagem digitalizada.

Sob uma abordagem matemática formal, o processo de amostragem pode ser visto como uma divisão do plano xy em uma grade, com as coordenadas do centro de cada grade sendo uma dupla de elementos do produto cartesiano $Z \times Z$ (também escrito Z^2), o qual é o conjunto de todos os pares ordenados dos elementos (a, b) com a e b sendo números pertencentes a Z (conjunto dos inteiros). Portanto $f(x,y)$ é uma imagem digital se (x,y) forem números inteiros de $Z \times Z$ e f uma função que atribui um valor de nível de cinza (isto é, um número real do conjunto de números reais R) para cada par distinto de coordenadas, ou seja, f é o processo de quantização descrito anteriormente. Se os níveis de cinza resultantes forem também números inteiros (como geralmente é o caso), Z substitui R e uma imagem digital então se torna uma função bidimensional cujas coordenadas e valores de amplitude são números inteiros.

Na especificação do processo de digitalização deve-se decidir que valores de N , M e n são adequados, do ponto de vista de qualidade da imagem e da quantidade de bytes necessários para armazená-la. A tabela 3 fornece uma idéia estimativa do número de bytes necessários para armazenar uma imagem de $N \times N$ pixels com 2^n tons de cinza, calculados como: $N \times N \times n / 8$. Assume-se que um pixel estará inteiramente contido em um byte, mesmo que isto signifique que alguns bits de cada byte permaneçam vazios. Por exemplo, para $n = 5$, assume-se que cada pixel ocupa um byte, restando 3 bits sem utilização em cada byte.

Do ponto de vista qualitativo, poder-se-ia perguntar: quantos pontos e níveis de cinza serão necessários para que a versão digitalizada de uma imagem apresente qualidade comparável à imagem original? Parece evidente que quanto maiores os valores de M , N e n , melhor a imagem digital resultante. Mas sabendo que elevados valores de M , N e n implicarão em maiores custos de digitalização e armazenamento, deve existir uma forma de definir valores adequados à qualidade desejada. Convém observar ainda que 'qualidade de imagem' é um conceito altamente subjetivo, que também depende fortemente dos requisitos da aplicação dada.

Para que o leitor possa tirar suas próprias conclusões sobre a dependência entre qualidade subjetiva e resolução espacial, a figura 4(a) mostra uma imagem de 256×256 pixels, com 256 níveis de cinza. Mantendo constante o número de tons de cinza, as figuras 4 (b)-(d) mostram os resultados da redução espacial de $N = 256$ para $N = 128$, 64 e 32, respectivamente.

Tabela 3 - Número de bytes necessários para armazenar uma imagem digital $N \times N$ com 2^n níveis de cinza

n	1	2	3	4	5	6	7	8
N								
32	128	256	512	512	1.024	1.024	1.024	1.024
64	512	1.024	2.048	2.048	4.096	4.096	4.096	4.096
128	2.048	4.096	8.192	8.192	16.384	16.384	16.384	16.384
256	8.192	16.384	32.768	32.768	65.536	65.536	65.536	65.536
512	32.768	65.536	131.072	131.072	262.144	262.144	262.144	262.144
1.024	131.072	262.144	393.216	524.288	655.360	786.432	917.504	1.048.576



(a)



(b)



(c)



(d)

Figura 4 - Efeito da resolução espacial na qualidade da imagem.

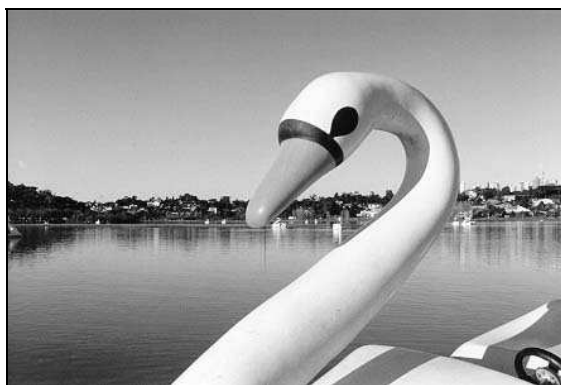
A figura 5 ilustra os efeitos da redução do número de níveis de cinza sobre a qualidade da imagem. Na figura 5(a) tem-se uma imagem de 442 x 299 pixels com 256 tons de cinza ($n = 8$). As figuras 5(b)-(h) foram obtidas reduzindo-se o número de bits de $n = 7$ até $n = 1$, enquanto a resolução espacial foi mantida constante em 442 x 299 pixels. A partir da imagem com 32 tons de cinza é perceptível o surgimento de uma imperfeição na imagem, conhecida como 'falso contorno' (*false contouring*).



(a)



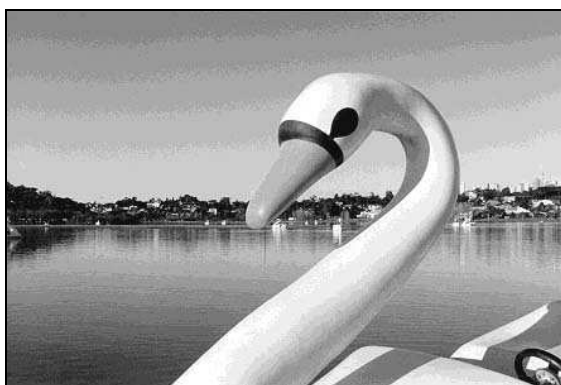
(b)



(c)



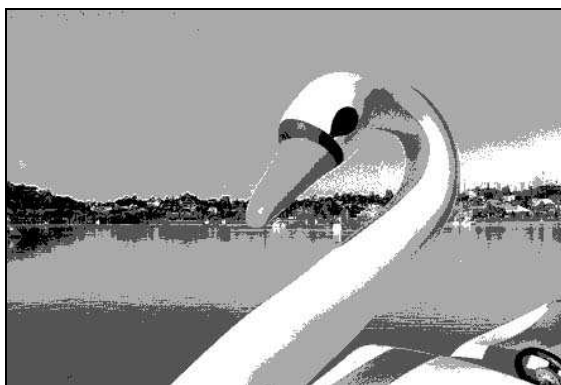
(d)



(e)



(f)



(g)



(h)

Figura 5 - Efeito do número de níveis de cinza na qualidade de uma imagem 442 x 299 com 256, 128, 64, 32, 16, 8, 4 e 2 níveis de cinza, respectivamente.

Para obter uma imagem digital de qualidade semelhante a de uma imagem de televisão P&B, são necessários 512 x 512 pixels e 128 níveis de cinza. Em geral, 64 níveis de cinza são considerados suficientes para o olho humano. Apesar disto, a maioria dos sistemas de visão artificial utiliza imagens com 256 níveis de cinza.

Os processos de amostragem e quantização podem ser aprimorados usando técnicas adaptativas. Sob o aspecto da amostragem, a idéia básica é utilizar maior número de pontos em regiões de grande detalhe, em detrimento das regiões homogêneas de grandes dimensões, que poderiam ser amostradas com menor número de pixels. Sob o ângulo da quantização, uma vez que o olho humano não é capaz de perceber sutis diferenças de tons de cinza nas imediações de variações abruptas de intensidade, o objetivo seria utilizar poucos níveis de cinza nestas regiões. O principal obstáculo para a implementação destas técnicas é a necessidade de identificação prévia (ainda que aproximada) das regiões presentes na imagem e das fronteiras entre elas. No caso da quantização, entretanto, outra técnica adaptativa pode ser utilizada. Efetuando um levantamento da frequência de ocorrência de todos os níveis de cinza permitidos, pode-se diminuir os degraus de quantização nas regiões da escala de cinza com maior concentração de ocorrência de pixels, aumentando-os nas demais regiões.

Leitura complementar

Para uma introdução à questão da conversão de uma cena tridimensional em uma imagem bidimensional, incluindo transformações de perspectiva, aspectos de calibração de câmeras e visão estéreo, sugerimos [Faugeras 1993], o capítulo 2 de [Schalkoff 1989] e a seção 2.5 de [Gonzalez e Woods 1992].

Para maiores detalhes sobre o funcionamento de câmeras CCD e fundamentos de sinais analógicos de vídeo recomendamos [Nince 1991].

Os capítulos 1 a 5 de [Lindley 1991] trazem uma descrição pormenorizada de um projeto de digitalizador de imagens (hardware e software).

Para uma análise dos efeitos produzidos na qualidade da imagem pela variação simultânea da resolução espacial e do número de níveis de cinza, ver o trabalho de Huang [Huang 1965], resumido na seção 2.3 de [Gonzalez e Woods 1992].

O capítulo 2 de [Pavlidis 1982] traz informações adicionais sobre os aspectos de amostragem e quantização.

Aos interessados em um aprofundamento matemático dos aspectos abordados nesta seção, recomendamos os capítulos 1, 4, 5 e 6 de [Pratt 1991].

2.2 Propriedades de uma imagem digital

Nesta seção consideraremos as principais relações entre pixels em uma imagem digital. Uma imagem digital é uma imagem $f(x,y)$ discretizada tanto espacialmente quanto em amplitude. Portanto, uma imagem digital pode ser vista como uma matriz cujas linhas e colunas identificam um ponto na imagem, cujo valor corresponde ao nível de cinza da imagem naquele ponto. Para efeito de notação, uma imagem digital será indicada por $f(x,y)$. Quando nos referirmos a um pixel em particular, utilizaremos letras minúsculas, tais como p e q . Um subconjunto de pixels de $f(x,y)$ será indicado por S .

2.2.1 Vizinhança

Um pixel p , de coordenadas (x,y) , tem 4 vizinhos horizontais e verticais, cujas coordenadas são $(x+1, y)$, $(x-1, y)$, $(x, y+1)$ e $(x, y-1)$. Estes pixels formam a chamada "4-vizinhança" de p , que será designada $N_4(p)$.

Os quatro vizinhos diagonais de p são os pixels de coordenadas $(x-1, y-1)$, $(x-1, y+1)$, $(x+1, y-1)$ e $(x+1, y+1)$, que constituem o conjunto $N_d(p)$.

A "8-vizinhança" de p é definida como:

$$N_8(p) = N_4(p) \cup N_d(p) \quad (2.3)$$

Os vários tipos de vizinhança estão ilustrados na figura 6.

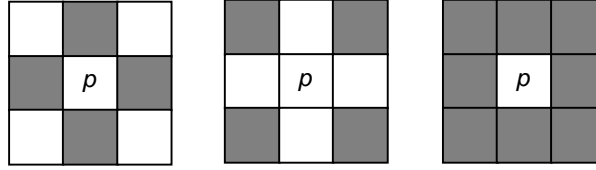


Figura 6 - Conceitos de 4-vizinhança, vizinhança diagonal e 8-vizinhança.

2.2.2 Conectividade

A conectividade entre pixels é um importante conceito usado para estabelecer limites de objetos e componentes de regiões em uma imagem. Para se estabelecer se dois pixels estão conectados, é necessário determinar se eles são adjacentes segundo algum critério e se seus níveis de cinza satisfazem a um determinado critério de similaridade. Por exemplo, em uma imagem binária, onde os pixels podem assumir os valores 0 e 1, dois pixels podem ser 4-vizinhos, mas somente serão considerados 4-conectados se possuírem o mesmo valor.

Seja V o conjunto de valores de tons de cinza utilizados para se definir a conectividade. Por exemplo, numa imagem binária, $V = \{1\}$ para a conexão de pixels com valor 1. Numa imagem de múltiplos tons de cinza, para a conexão de pixels com valores de intensidade na faixa de 32 a 64, $V = \{32, 33, \dots, 63, 64\}$. Conhecendo o conceito de vizinhança e dado o conjunto V , podemos definir os seguintes critérios de conectividade:

1. "4-conectividade": dois pixels p e q com valores de tom de cinza contidos em V , são "4-conectados" se $q \in N_4(p)$.
2. "8-conectividade": dois pixels p e q com valores de tom de cinza contidos em V , são "8-conectados" se $q \in N_8(p)$.
3. "m-conectividade (conectividade mista)": dois pixels p e q com valores de tom de cinza contidos em V , são "m-conectados" se:

(i) $q \in N_4(p)$ ou

(ii) $q \in N_d(p) \text{ e } N_4(p) \cap N_4(q) = \emptyset$.

A conectividade mista é uma modificação da 8-conectividade e é introduzida para eliminar os múltiplos caminhos que geralmente surgem quando a 8-conectividade é usada. Por exemplo, seja o trecho de imagem da figura 7(a). Para $V = \{1\}$ os caminhos entre 8 vizinhos do pixel do centro são indicados por linhas contínuas na figura 7(b), onde se pode observar a existência de caminhos redundantes entre os pixels do centro e do canto superior esquerdo da figura. Esta redundância é resolvida utilizando-se a m-conectividade, que remove a conexão diagonal redundante, como mostra a figura 7(c).

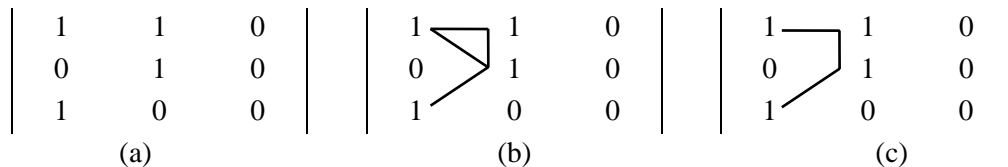


Figura 7 - (a) Segmento de imagem binária, (b) 8-vizinhos do pixel central, (c) m-vizinhos do pixel central.

2.2.3 Adjacência

Um pixel p é adjacente a um pixel q se eles forem conectados. Há tantos critérios de adjacência quantos são os critérios de conectividade. Dois subconjuntos de imagens, S_1 e S_2 , são adjacentes se algum pixel em S_1 é adjacente a algum pixel em S_2 .

2.2.4 Caminho

Um caminho (*path*) de um pixel p de coordenadas (x,y) a um pixel q de coordenadas (s,t) é uma seqüência de pixels distintos de coordenadas: $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$,

onde:

$$(x_0, y_0) = (x, y)$$

$$(x_n, y_n) = (s, t)$$

$$(x_i, y_i) \text{ é adjacente a } (x_{i-1}, y_{i-1})$$

$$1 \leq i \leq n$$

n é denominado o comprimento do caminho.

2.2.5 Medições de distância

Dados os pixels p, q e z , de coordenadas (x,y) , (s,t) e (u,v) , respectivamente, define-se a função distância D , cujas propriedades são:

$$(i) D(p,q) \geq 0 \text{ (} D(p,q) = 0 \text{ se e somente se } p = q \text{)}$$

$$(ii) D(p,q) = D(q,p)$$

$$(iii) D(p,z) \leq D(p,q) + D(q,z)$$

Distância Euclidiana

$$D_e(p,q) = \sqrt{(x-s)^2 + (y-t)^2} \quad (2.4)$$

Para esta medida de distância, os pixels com distância euclidiana em relação a (x,y) menor ou igual a algum valor r , são os pontos contidos em um círculo de raio r centrado em (x,y) .

Distância D_4 (city-block)

$$D_4(p,q) = |x-s| + |y-t| \quad (2.5)$$

onde $| \cdot |$ denota módulo (ou valor absoluto).

Neste caso, os pixels tendo uma distância D_4 em relação a (x,y) menor ou igual a algum valor r formam um losango centrado em (x,y) . Os pixels com $D_4 = 1$ são os 4-vizinhos de (x,y) .

Distância D_8 (tabuleiro de xadrez)

$$D_8(p,q) = \max(|x-s|, |y-t|) \quad (2.6)$$

onde \max é um operador que devolve o maior valor dentre um conjunto de valores entre parênteses.

Neste caso os pixels com distância D_8 em relação a (x,y) menor ou igual a algum valor r formam um quadrado centrado em (x,y) . Os pixels com $D_8 = 1$ são os 8-vizinhos de (x,y) .

O conceito de distância pode estar relacionado ao conceito de conectividade. A distância D_m expressa a distância entre dois pontos m-conectados.

Exercício resolvido

Seja o trecho de imagem binária a seguir:

	p_3	p_4
p_1	p_2	
p		

Supondo que $V = \{1\}$, $p = p_2 = p_4 = 1$ e que p_1 e p_3 podem apresentar valores 0 ou 1, calcular a distância D_m entre p e p_4 para as seguintes situações:

a) Se $p_1 = p_3 = 0$.

Solução: a distância D_m vale 2, pois o caminho m entre p e p_4 é obtido unindo-se os pixels p , p_2 e p_4 .

b) Se p_1 ou p_3 valem 1.

Solução: a distância D_m vale 3, pois o caminho m entre p e p_4 será p , p_1 , p_2 , p_4 ou p , p_2 , p_3 , p_4 .

c) Se p_1 e p_3 valem 1.

Solução: a distância D_m vale 4, pois o caminho m entre p e p_4 será p , p_1 , p_2 , p_3 , p_4 .

Leitura complementar

As subseções 2.4.3 e 2.4.4 de [Gonzalez e Woods 1992] apresentam um método de atribuição de rótulos a aglomerados de pixels conectados de uma imagem e relacionam este procedimento aos conceitos matemáticos de relação binária, relação de equivalência e fecho transitivo.

2.3 Operações lógicas e aritméticas

Sabemos que após uma imagem ter sido adquirida e digitalizada, ela pode ser vista como uma matriz de inteiros e portanto pode ser manipulada numericamente utilizando operações lógicas e/ou aritméticas. Estas operações podem ser efetuadas pixel a pixel ou orientadas a vizinhança. No primeiro caso, elas podem ser descritas pela seguinte notação:

$$X \text{ opn } Y = Z$$

onde X e Y podem ser imagens (matrizes) ou escalares, Z é obrigatoriamente uma matriz e opn é um operador aritmético (+, -, x e /) ou lógico (AND, OR, XOR) binário².

Sejam duas imagens X e Y de igual tamanho. Estas imagens podem ser processadas pixel a pixel utilizando um operador aritmético ou lógico, produzindo uma terceira imagem Z , cujos pixels correspondem ao resultado de $X \text{ opn } Y$ para cada elemento de X e Y , conforme ilustra esquematicamente a figura 8.

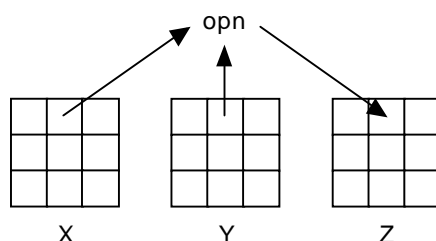


Figura 8 - Operações lógicas / aritméticas pixel a pixel.

² Aqui, o termo binário deve ser entendido como 'que requer dois operandos' e não tem qualquer relação com imagem binária.

2.3.1 Operações aritméticas pixel a pixel

Ao executarmos operações aritméticas sobre imagens, devemos tomar especial cuidado com os problemas de *underflow* ou *overflow* do resultado. A adição de duas imagens de 256 tons de cinza, por exemplo, pode resultar em um número maior que 255 para alguns pixels, ao mesmo tempo que a subtração de duas imagens pode resultar em valores negativos para alguns elementos. Para contornar estes problemas, existem basicamente duas alternativas: (1) manter os resultados intermediários em uma matriz na qual o espaço em memória alocado para cada pixel permita a representação de números negativos e/ou maiores que 255 e em seguida proceder a uma normalização destes valores intermediários; (2) truncar os valores maiores que o máximo valor permitido, bem como os valores negativos, igualando-os a 255 e 0, respectivamente. A decisão depende do objetivo que se tem em mente ao executar determinada operação. Efetivamente, a segunda alternativa é mais simples que a primeira.

Exercício resolvido

Dadas as matrizes X e Y a seguir, correspondentes a trechos 3 x 3 de imagens de 256 tons de cinza, adicioná-las e informar: (a) o resultado intermediário (sem considerações de *underflow* e *overflow*), (b) o resultado final utilizando normalização, (c) o resultado final utilizando truncamento.

$$X = \begin{bmatrix} 200 & 100 & 100 \\ 0 & 10 & 50 \\ 50 & 250 & 120 \end{bmatrix} \quad Y = \begin{bmatrix} 100 & 220 & 230 \\ 45 & 95 & 120 \\ 205 & 100 & 0 \end{bmatrix}$$

Solução:

$$(a) \begin{bmatrix} 300 & 320 & 330 \\ 45 & 105 & 170 \\ 255 & 350 & 120 \end{bmatrix}$$

(b) Fazendo com que a escala [45, 350] seja adequada ao intervalo [0, 255], utilizando-se a relação

$$g = \frac{255}{f_{\max} - f_{\min}} (f - f_{\min}), \quad (2.7)$$

obtem-se:

$$\begin{bmatrix} 213 & 230 & 238 \\ 0 & 50 & 105 \\ 175 & 255 & 63 \end{bmatrix}$$

(c) Truncando os valores maiores que 255, obtém-se:

$$\begin{bmatrix} 255 & 255 & 255 \\ 45 & 105 & 170 \\ 255 & 255 & 120 \end{bmatrix}$$

As principais aplicações das operações aritméticas sobre imagens estão resumidas na tabela 4. Assim como Y foi implicitamente considerado até aqui como sendo uma matriz, ele também pode ser um escalar. A segunda coluna da tabela 4 avalia os efeitos qualitativos das operações aritméticas sobre imagens, abordando ambas as possibilidades. As figuras 9 a 12 mostram exemplos de cada operação aritmética.

Tabela 4 - Efeitos e aplicações das operações aritméticas sobre imagens

Operação	Efeito sobre a imagem	Aplicações
Adição	Z é o resultado da soma dos valores de intensidade de X e Y . Se Y for um escalar positivo, Z será uma versão mais clara de X ; o acréscimo de intensidade será o próprio valor de Y .	<ul style="list-style-type: none"> • Normalização de brilho³ de imagens • Remoção de ruídos (ver técnica da filtragem pela média de múltiplas imagens na subseção 4.2.4)
Subtração	Z é o resultado da diferença dos valores de intensidade de X e Y . Se Y for um escalar positivo, Z será uma versão mais escura de X ; o decréscimo de intensidade será o próprio valor de Y .	<ul style="list-style-type: none"> • Detecção de diferenças entre duas imagens (eventualmente adquiridas de forma consecutiva) da mesma cena
Multiplicação	Z é o produto dos valores de intensidade de X e Y . Se Y for um escalar positivo, os valores de intensidade de Z serão diretamente proporcionais a X por um fator Y .	<ul style="list-style-type: none"> • Calibração de brilho⁴
Divisão	Z é o razão dos valores de intensidade de X pelos valores correspondentes em Y . Se Y for um escalar positivo, os valores de intensidade de Z serão inversamente proporcionais a X por um fator Y .	<ul style="list-style-type: none"> • Normalização de brilho

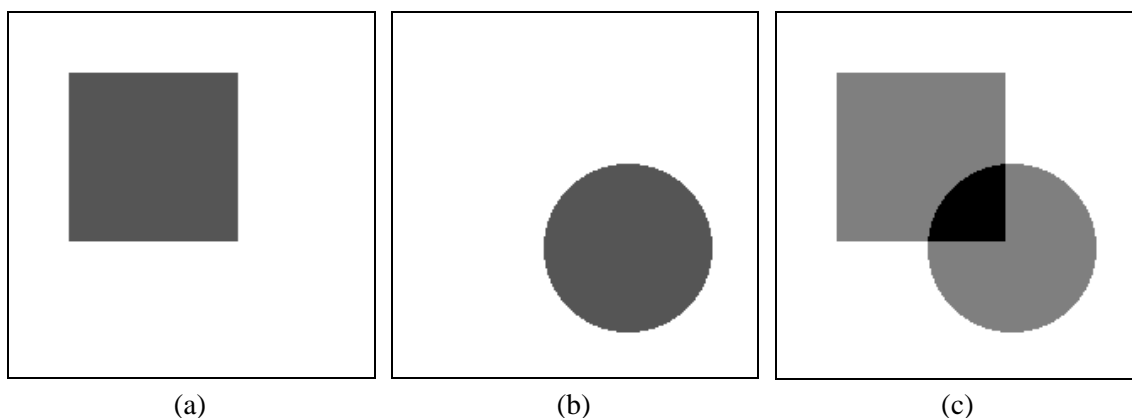


Figura 9 - Exemplo de adição de imagens monocromáticas: (a) X , (b) Y , (c) $X + Y$ (normalizado).

³ O processo de normalização de brilho consiste em adequar a faixa total de níveis de cinza a um intervalo pré-definido, de forma semelhante ao efetuado na parte (b) do Exercício Resolvido desta seção.

⁴ A calibração de brilho é um processo semelhante à normalização de brilho, mas que pode estar relacionado à adequação a diferentes valores de iluminância sobre uma mesma cena, por exemplo.

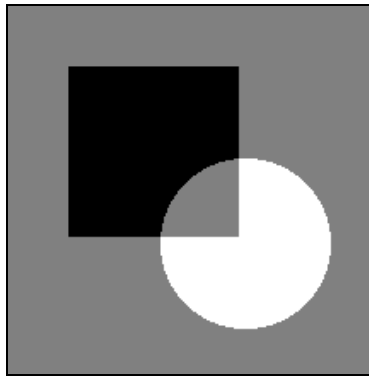


Figura 10 - Exemplo de subtração das imagens monocromáticas das figuras 9(a) e 9(b): $X - Y$ (normalizado).

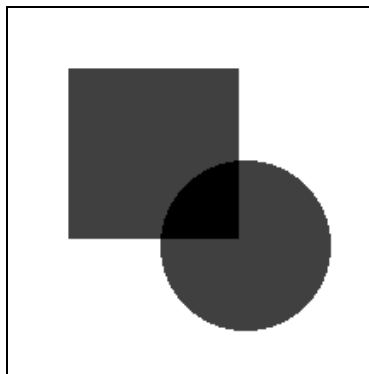


Figura 11 - Exemplo de multiplicação das imagens monocromáticas das figuras 9(a) e 9(b): XY (normalizado).

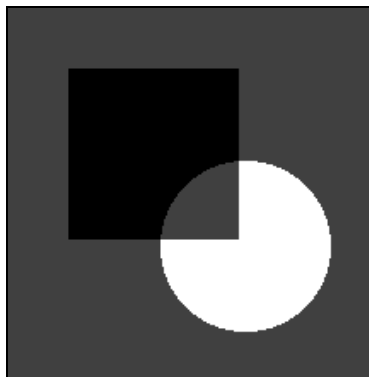
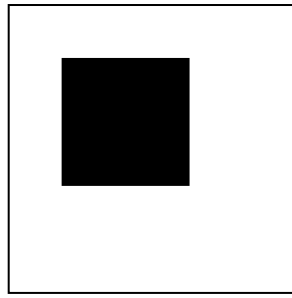


Figura 12 - Exemplo de divisão de imagens monocromáticas das figuras 9(a) e 9(b): X / Y (normalizado).

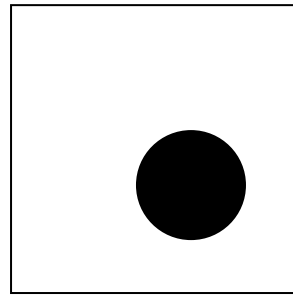
2.3.2 Operações lógicas pixel a pixel

Todas as operações lógicas (ou booleanas) conhecidas podem ser aplicadas entre imagens, inclusive a operação de complemento (NOT), que é uma operação unária (requer apenas um operando). Operações lógicas podem ser efetuadas em imagens com qualquer número de níveis de cinza mas são melhor compreendidas quando vistas em imagens binárias, como ilustra a figura 13. As figuras 14 a 17 ilustram as operações AND, OR, XOR e NOT aplicadas a imagens com múltiplos tons de cinza.



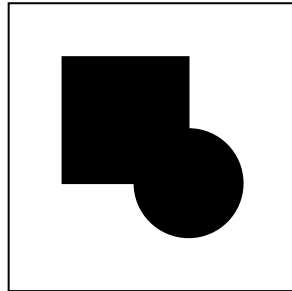
X

(a)



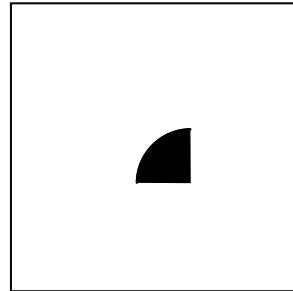
Y

(b)



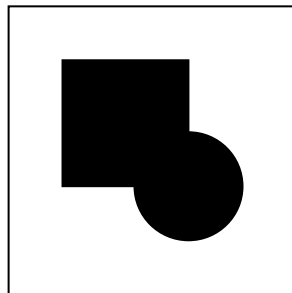
X and Y

(c)



X or Y

(d)



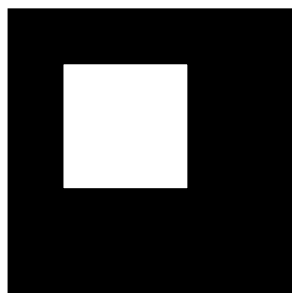
X xor Y

(e)



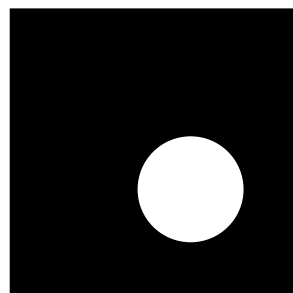
(not X) and Y

(f)



not X

(g)



not Y

(h)

Figura 13 - Exemplos de operações lógicas em imagens binárias.

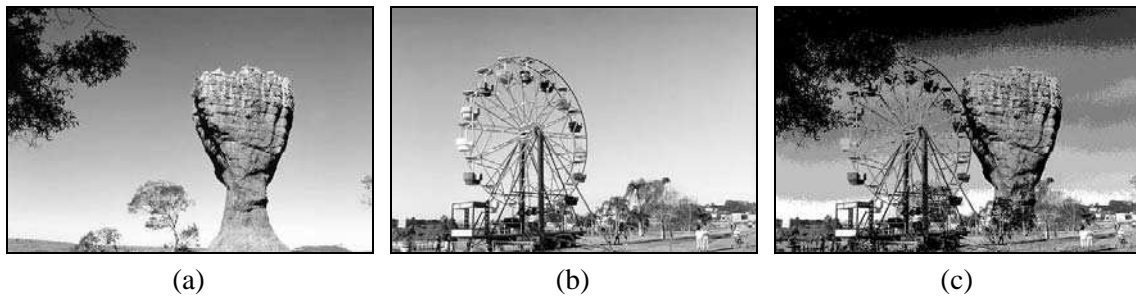


Figura 14 - Exemplo de operação AND entre imagens monocromáticas: (a) X , (b) Y , (c) $X \wedge Y$.

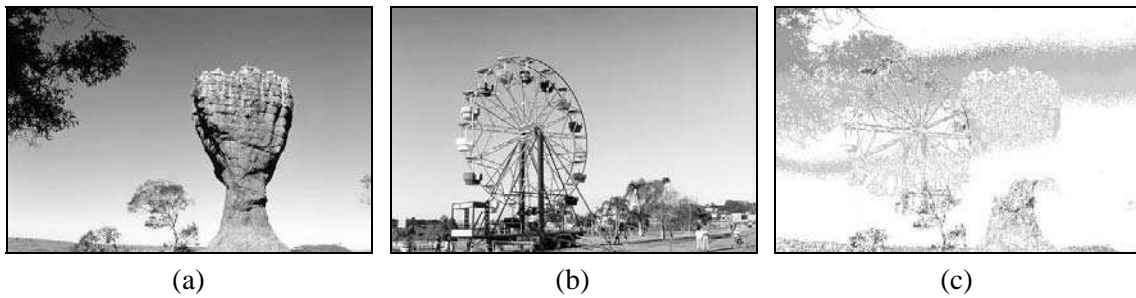


Figura 15 - Exemplo de operação OR entre imagens monocromáticas: (a) X , (b) Y , (c) $X \vee Y$.

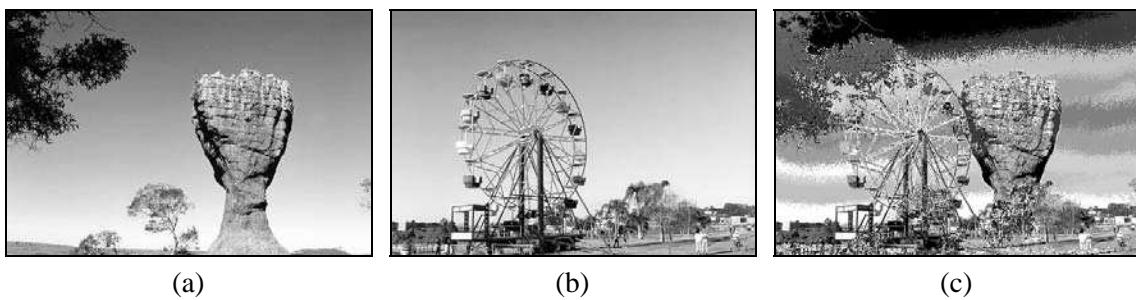


Figura 16 - Exemplo de operação XOR entre imagens monocromáticas: (a) X , (b) Y , (c) $X \oplus Y$.



Figura 17 - Exemplo de operação NOT sobre imagem monocromática: (a) X , (b) $\text{NOT } X$.

2.3.3 Operações orientadas a vizinhança

As operações lógicas e aritméticas orientadas a vizinhança utilizam o conceito de convolução com máscaras (ou janelas ou *templates*), que será introduzido a seguir e detalhado na próxima seção.

Seja uma sub-área de uma imagem:

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

onde: Z_1, \dots, Z_9 são os valores de tons de cinza de cada pixel.

Seja uma máscara 3 x 3 de coeficientes genéricos W_1, \dots, W_9 :

W_1	W_2	W_3
W_4	W_5	W_6
W_7	W_8	W_9

A máscara acima percorrerá a imagem, desde o seu canto superior esquerdo até seu canto inferior direito. A cada posição relativa da máscara sobre a imagem, o pixel central da subimagem em questão será substituído, em uma matriz denominada 'imagem-destino', por um valor:

$$Z = \sum_{i=1}^9 W_i \cdot Z_i, \quad (2.8)$$

As operações de convolução com máscaras são amplamente utilizadas no processamento de imagens. Uma seleção apropriada dos coeficientes W_1, \dots, W_9 torna possível uma grande variedade de operações úteis, tais como redução de ruído, afinamento e detecção de características da imagem. Deve-se observar, entretanto, que a operação de convolução com máscaras exige grande esforço computacional. Por exemplo, a aplicação de uma máscara 3 x 3 sobre uma imagem 512 x 512 requer nove multiplicações e oito adições para cada localização de pixel, num total de 2.359.296 multiplicações e 2.097.152 adições. Por esta razão, aliada à relativa simplicidade de implementação de multiplicadores, somadores e registradores de deslocamento (*shift registers*), a literatura registra diversas implementações de convolução com máscaras em hardware.

Leitura complementar

O capítulo 11 de [Lindley 1991] apresenta, além das operações abordadas nesta seção, outras funções interessantes que operam pixel a pixel e podem ser aplicadas a imagens monocromáticas.

O capítulo 1 de [Dougherty e Giardina 1987] apresenta as operações básicas sobre imagens monocromáticas sob um enfoque exclusivamente matricial.

2.4 Operações de convolução com máscaras

Conforme antecipamos na seção anterior, inúmeras operações úteis em processamento de imagens são efetuadas a partir de um mesmo conceito básico, o de convolução com máscaras. Nesta seção abordaremos em mais detalhes o funcionamento das operações de convolução e apresentaremos alguns exemplos típicos de máscaras e os resultados que elas produzem quando aplicadas a imagens monocromáticas.

A operação de convolução unidimensional entre dois vetores A e B, denotada $A*B$, pode ser entendida como um conjunto de somas de produtos entre os valores de A e B, sendo que inicialmente o vetor B é espelhado e após cada soma de produtos é deslocado espacialmente de uma posição. Para ilustrar este conceito, mostraremos a seguir, passo a passo, a convolução do vetor $A = \{0, 1, 2, 3, 2, 1, 0\}$ com o vetor $B = \{1, 3, -1\}$.

1. Inicialmente, o vetor B é espelhado e alinhado com o primeiro valor de A. O resultado da convolução é $(0 \times (-1)) + (0 \times 3) + (1 \times 1) = 1$ (valores em branco assumidos como zero) e é colocado em A*B na posição correspondente ao centro do conjunto B.

A		0	1	2	3	2	1	0	
B	-1	3	1						
A*B		1							

2. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(0 \times (-1)) + (1 \times 3) + (2 \times 1) = 5$.

A		0	1	2	3	2	1	0	
B		-1	3	1					
A*B		1	5						

3. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(1 \times (-1)) + (2 \times 3) + (3 \times 1) = 8$.

A		0	1	2	3	2	1	0	
B			-1	3	1				
A*B		1	5	8					

4. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(2 \times (-1)) + (3 \times 3) + (2 \times 1) = 9$.

A		0	1	2	3	2	1	0	
B				-1	3	1			
A*B		1	5	8	9				

5. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(3 \times (-1)) + (2 \times 3) + (1 \times 1) = 4$.

A		0	1	2	3	2	1	0	
B					-1	3	1		
A*B		1	5	8	9	4			

6. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(2 \times (-1)) + (1 \times 3) + (0 \times 1) = 1$.

A		0	1	2	3	2	1	0	
B						-1	3	1	
A*B		1	5	8	9	4	1		

7. O conjunto B é deslocado de uma posição. O resultado da convolução A*B é $(1 \times (-1)) + (0 \times 3) + (0 \times 1) = -1$. (valores em branco assumidos como zero)

A		0	1	2	3	2	1	0	
B							-1	3	1
A*B		1	5	8	9	4	1	-1	

O conjunto {1, 5, 8, 9, 4, 1, -1} é o resultado final da operação de convolução.

Este raciocínio pode ser expandido para o caso bidimensional, onde a imagem a ser processada é uma matriz bidimensional relativamente grande e corresponde ao conjunto A de

nosso exemplo anterior, enquanto uma matriz de pequenas dimensões (também chamada máscara ou janela) corresponde ao conjunto B. A máscara, após ter sido espelhada tanto na horizontal quanto na vertical, percorrerá todos os pontos da imagem deslocando-se ao longo de cada linha e entre as várias linhas, da direita para a esquerda, de cima para baixo, até ter processado o último elemento da matriz imagem. O resultado será armazenado em uma matriz de mesmas dimensões que a imagem original.

Seja a matriz A (imagem) dada por:

$$\begin{bmatrix} 5 & 8 & 3 & 4 & 6 & 2 & 3 & 7 \\ 3 & 2 & 1 & 1 & 9 & 5 & 1 & 0 \\ 0 & 9 & 5 & 3 & 0 & 4 & 8 & 3 \\ 4 & 2 & 7 & 2 & 1 & 9 & 0 & 6 \\ 9 & 7 & 9 & 8 & 0 & 4 & 2 & 4 \\ 5 & 2 & 1 & 8 & 4 & 1 & 0 & 9 \\ 1 & 8 & 5 & 4 & 9 & 2 & 3 & 8 \\ 3 & 7 & 1 & 2 & 3 & 4 & 4 & 6 \end{bmatrix}$$

e seja a matriz B (máscara) a seguir:

$$\begin{bmatrix} 2 & 1 & 0 \\ 1 & 1 & -1 \\ 0 & -1 & -2 \end{bmatrix}.$$

A operação de convolução bidimensional produzirá como resultado a matriz:

$$\begin{bmatrix} 20 & 10 & 2 & 26 & 23 & 6 & 9 & 4 \\ 18 & 1 & -8 & 2 & 7 & 3 & 3 & -11 \\ 14 & 22 & 5 & -1 & 9 & -2 & 8 & -1 \\ 29 & 21 & 9 & -9 & 10 & 12 & -9 & -9 \\ 21 & 1 & 16 & -1 & -3 & -4 & 2 & 5 \\ 15 & -9 & -3 & 7 & -6 & 1 & 17 & 9 \\ 21 & 9 & 1 & 6 & -2 & -1 & 23 & 2 \\ 9 & -5 & -25 & -10 & -12 & -15 & -1 & -12 \end{bmatrix}$$

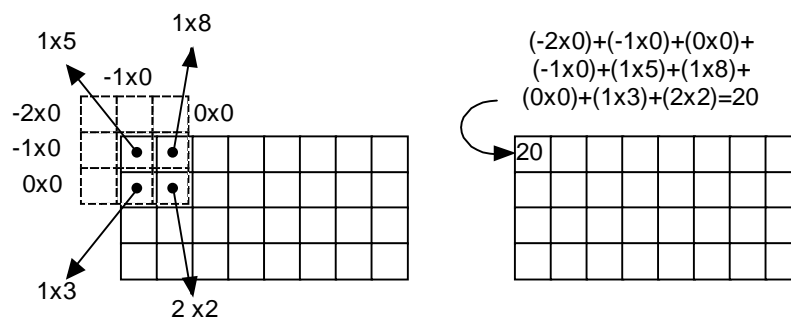


Figura 18 - Cálculo do primeiro valor da convolução de A por B.

A figura 18 ilustra em detalhes o cálculo do resultado correspondente ao pixel no canto superior esquerdo da imagem. Observar que a máscara B foi espelhada em relação a x e a y antes do cálculo das somas de produtos.

Para calcular os valores resultantes dos pixels próximos às bordas da imagem, podem ser adotadas diversas estratégias, dentre elas:

1. preencher com zeros o contorno da imagem, de maneira condizente com o tamanho de máscara utilizado, como ilustra a figura 18.
2. preencher o contorno da imagem com os mesmos valores da(s) primeira(s) e última(s) linha(s) e coluna(s).
3. prevenir a eventual introdução de erros nas regiões de bordas da imagem causados por qualquer um dos métodos acima, considerando na imagem resultante apenas os valores para os quais a máscara de convolução ficou inteiramente contida na imagem original.

A seguir, ilustraremos o uso do conceito de convolução com máscaras aplicado à detecção de características de imagens, particularmente pontos isolados, linhas e bordas.

2.4.1 Detecção de pontos isolados

A máscara a seguir é um exemplo de operador de convolução que, quando aplicado a uma imagem, destacará pixels brilhantes circundados por pixels mais escuros. Como será visto na seção 4.3, este operador corresponde a um filtro passa-altas.

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

2.4.2 Detecção de linhas

As máscaras a seguir podem ser utilizadas para a detecção de linhas horizontais e verticais (acima) e diagonais (abaixo).

$$\begin{bmatrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{bmatrix}$$

$$\begin{bmatrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{bmatrix}$$

$$\begin{bmatrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{bmatrix}$$

2.4.3 Detecção de bordas

O tema 'detecção de bordas' (*edge detection*) vem desafiando os pesquisadores da área de Processamento de Imagens há muitos anos e sobre ele continuam sendo experimentadas novas técnicas, cujos resultados são publicados ainda hoje nos mais conceituados periódicos científicos mundiais. Trata-se, portanto, de um tema em aberto, a detecção de bordas em cenas consideradas 'difíceis'.

Apenas a título de ilustração da operação de convolução com máscaras apresentamos a seguir alguns exemplos de máscaras que podem ser utilizadas para a tarefa de detecção de bordas.

Define-se borda (*edge*) como a fronteira entre duas regiões cujos níveis de cinza predominantes são razoavelmente diferentes. Pratt [Pratt 1991] define uma borda de luminosidade como uma descontinuidade na luminosidade de uma imagem. Analogamente, pode-se definir borda de textura ou borda de cor, em imagens onde as informações de textura ou

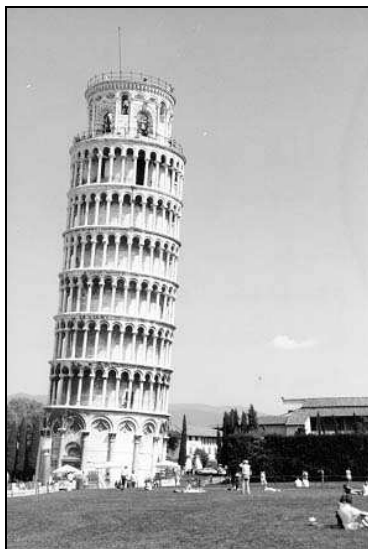
cor, respectivamente, são as mais importantes. Neste livro trataremos somente de bordas de luminosidade, às quais denominaremos simplesmente bordas.

Para a detecção e realce de bordas, aplicam-se habitualmente filtros espaciais lineares de dois tipos: (a) baseados no gradiente da função de luminosidade, $I(x,y)$, da imagem, e (b) baseados no laplaciano de $I(x,y)$.

Tanto o gradiente quanto o laplaciano costumam ser aproximados por máscaras de convolução ou operadores 3 x 3. Exemplos destas máscaras são os operadores de Roberts, Sobel, Prewitt e Frei-Chen, mostrados na tabela 5.

Tabela 5 - Operadores 3 x 3 utilizados para estimar a amplitude do gradiente através de uma borda.

Operador	Vertical	Horizontal
Roberts	$\begin{bmatrix} 0 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
Sobel	$\frac{1}{4} \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{4} \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$
Prewitt	$\frac{1}{3} \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{3} \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$
Frei-Chen	$\frac{1}{2 + \sqrt{2}} \begin{bmatrix} 1 & 0 & -1 \\ \sqrt{2} & 0 & -\sqrt{2} \\ 1 & 0 & -1 \end{bmatrix}$	$\frac{1}{2 + \sqrt{2}} \begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix}$



(a)



(b)



(c)

Figura 19 - Exemplo de realce e detecção de bordas. (a) imagem original, (b) realce de bordas utilizando os operadores de Prewitt horizontal e vertical, (c) realce de bordas utilizando os operadores de Sobel horizontal e vertical.

A figura 19 mostra os resultados da aplicação dos operadores de Prewitt e Sobel a uma imagem monocromática. Os resultados obtidos com a aplicação dos operadores verticais e

horizontais foram combinados por meio de uma operação lógica OR. Notar que as diferenças são pouco perceptíveis.

O laplaciano é um operador definido como:

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \quad (2.9)$$

e que pode ser aproximado pelas máscaras da figura 20.

$$\begin{array}{ccc} \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix} & \begin{bmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & 24 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{bmatrix} & \begin{bmatrix} -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & 8 & 8 & 8 & -1 & -1 \\ -1 & -1 & -1 & 8 & 8 & 8 & -1 & -1 \\ -1 & -1 & -1 & 8 & 8 & 8 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{bmatrix} \\ (a) & (b) & (c) \end{array}$$

Figura 20 - Máscaras para o cálculo do laplaciano: (a) 3 x 3, (b) 5 x 5, (c) 9 x 9.

A figura 21 mostra os resultados obtidos com cada uma das máscaras da figura 20 aplicadas a uma imagem monocromática.

Embora o laplaciano seja insensível à rotação, e portanto capaz de realçar ou detectar bordas em qualquer direção, seu uso é restrito devido a sua grande suscetibilidade a ruído.

A figura 22 mostra um exemplo de aplicação do laplaciano 3 x 3 acima a uma imagem monocromática com e sem ruído.

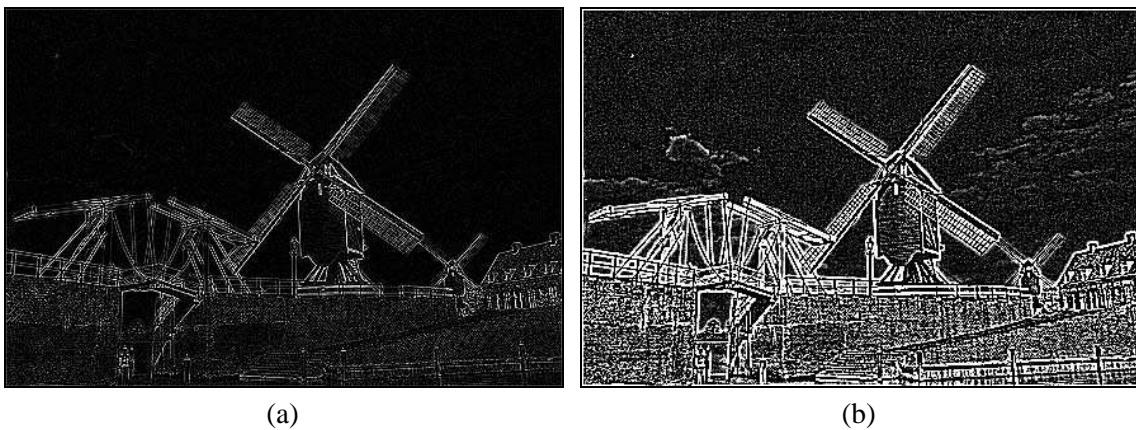
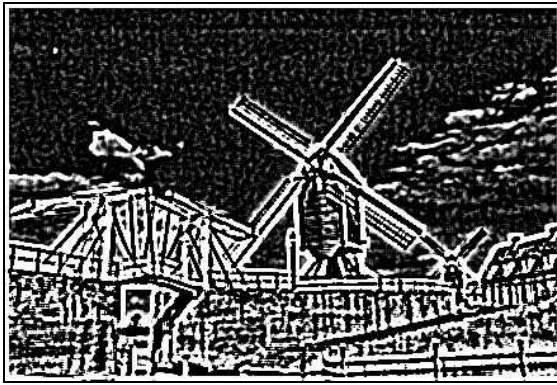


Figura 21 - Resultados da aplicação da máscara do laplaciano: (a) 3 x 3, (b) 5 x 5, (c) 9 x 9, (d) imagem original.



(c)

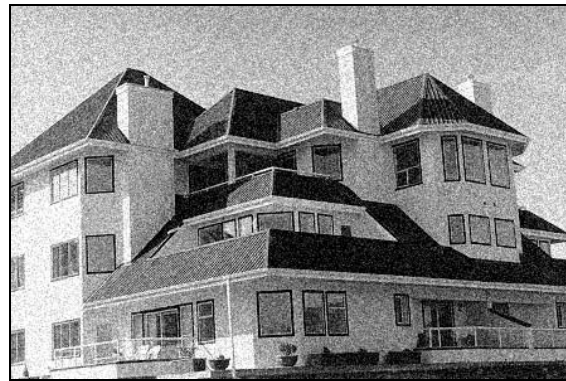


(d)

Figura 21 - Continuação.



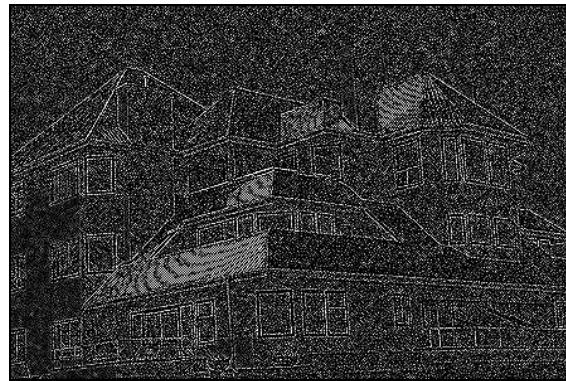
(a)



(b)



(c)



(d)

Figura 22 - Exemplo de utilização do laplaciano: (a) imagem original, (b) imagem ruidosa, (c) laplaciano sobre (a), (d) laplaciano sobre (b).

Existem outros operadores direcionais, que nada mais são que conjuntos de máscaras que representam aproximações discretas de bordas ideais em várias direções. Estes operadores incluem as máscaras direcionais introduzidas por Prewitt [Prewitt 1970], Kirsch [Kirsch 1971], e as máscaras simples de 3 e 5 níveis de Robinson [Robinson 1977]. A tabela 6 mostra estas máscaras com suas respectivas direções cardeais.

Tabela 6 - Máscaras de Prewitt, Kirsch e Robinson.

Direção da borda	Direção grad.	Prewitt			Kirsch			Robinson 3 níveis			Robinson 5 níveis		
0	N	1	1	1	5	5	5	1	1	1	1	2	1
		1	-2	1	-3	0	-3	0	0	0	0	0	0
		-1	-1	-1	-3	-3	-3	-1	-1	-1	-1	-2	-1
1	NO	1	1	1	5	5	-3	1	1	0	2	1	0
		1	-2	-1	5	0	-3	1	0	-1	1	0	-1
		1	-1	-1	-3	-3	-3	0	-1	-1	0	-1	-2
2	O	1	1	-1	5	-3	-3	1	0	-1	1	0	-1
		1	-2	-1	5	0	-3	1	0	-1	2	0	-2
		1	1	-1	5	-3	-3	1	0	-1	1	0	-1
3	SO	1	-1	-1	-3	-3	-3	0	-1	-1	0	-1	-2
		1	-2	-1	5	0	-3	1	0	-1	1	0	-1
		1	1	1	5	5	-3	1	1	0	2	1	0
4	S	-1	-1	-1	-3	-3	-3	-1	-1	-1	-1	-2	-1
		1	-2	1	-3	0	-3	0	0	0	0	0	0
		1	1	1	5	5	5	1	1	1	1	2	1
5	SE	-1	-1	1	-3	-3	-3	-1	-1	0	-2	-1	0
		-1	-2	1	-3	0	5	-1	0	1	-1	0	1
		1	1	1	-3	5	5	0	1	1	0	1	2
6	E	-1	1	1	-3	-3	5	-1	0	1	-1	0	1
		-1	-2	1	-3	0	5	-1	0	1	-2	0	2
		-1	1	1	-3	-3	5	-1	0	1	-1	0	1
7	NE	1	1	1	-3	5	5	0	1	1	0	1	2
		-1	-2	1	-3	0	5	-1	0	1	-1	0	1
		-1	-1	1	-3	-3	-3	-1	-1	0	-2	-1	0
Fator de escala		1/5			1/15			1/3			1/4		

Leitura complementar

Aos interessados em um aprofundamento matemático dos aspectos abordados nesta seção, recomendamos os capítulos 7 e 16 de [Pratt 1991].

A seção 6.4 de [Haralick e Shapiro 1992] trata dos temas convolução e correlação. A questão de detecção de bordas e linhas é vista no capítulo 7 do mesmo livro.

O capítulo 3 de [Dougherty e Giardina 1987] é inteiramente dedicado à detecção de bordas.

O artigo de Dawson [Dawson 1987] traz fragmentos de código em C para a convolução de imagens com máscaras 3 x 3 e explica sua possível utilização em processos de filtragem e detecção de bordas.

Prosise [Prosise 1994a] apresenta exemplos de máscaras de convolução úteis para produção de efeitos em imagens, tais como realce, borramento (*blurring*) e o efeito de baixo relevo (*emboss*).

Jain [Jain 1989] apresenta o conceito de gradientes estocásticos para resolver o problema da detecção de bordas em imagens ruidosas.

2.5 Transformações geométricas

Transformações geométricas são operações de processamento de imagens cujo principal efeito é a alteração da posição espacial dos pixels que a compõem. Elas costumam ser úteis em situações que vão desde a correção de distorções até a produção de efeitos artísticos sobre imagens.

2.5.1 Ampliação e redução (*zoom*)

As operações de ampliação e redução de imagens (em inglês, *zoom in* e *zoom out*, respectivamente) são processos pelos quais as dimensões de uma imagem são aumentadas ou diminuídas para efeito de visualização. A maneira mais simples de ampliar uma imagem é duplicar os valores dos pixels na direção X ou Y ou em ambas. Se o fator de ampliação não for o mesmo para as duas direções, a razão de aspecto (relação entre a dimensão horizontal e a vertical de uma imagem) da imagem será alterada.

Para expandir uma imagem por um fator 2, cada pixel é copiado 4 vezes na imagem resultante, conforme ilustra a figura 23. Convém notar que a resolução da imagem não é alterada, apenas seu tamanho para efeito de visualização.

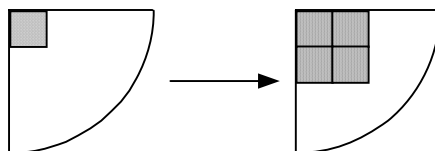


Figura 23 - Expansão de um pixel em 4 (*zoom 2x*)

Para reduzir as dimensões de uma imagem de um fator 2, basta utilizar o processo inverso, isto é converter cada agrupamento de quatro pixels novamente em 1 pixel. O problema neste caso é que normalmente estes pixels apresentarão valores diferentes de cinza, o que equivale a dizer que poderá haver perda de informação no processo de *zoom out*. Para minimizar este aspecto, uma técnica comum é substituir na imagem resultante o valor do pixel pela média dos quatro pixels equivalentes na imagem original.

Para um *zoom* de quatro vezes, utiliza-se uma vizinhança de dezesseis pixels e assim por diante. Para ampliar ou reduzir uma imagem de um fator fracionário, são necessários algoritmos de interpolação cujo detalhamento foge ao escopo deste livro.

A figura 24 mostra exemplos de *zoom in* e *zoom out* para imagens monocromáticas.

2.5.2 Alterações de dimensões (*scaling* e *sizing*)

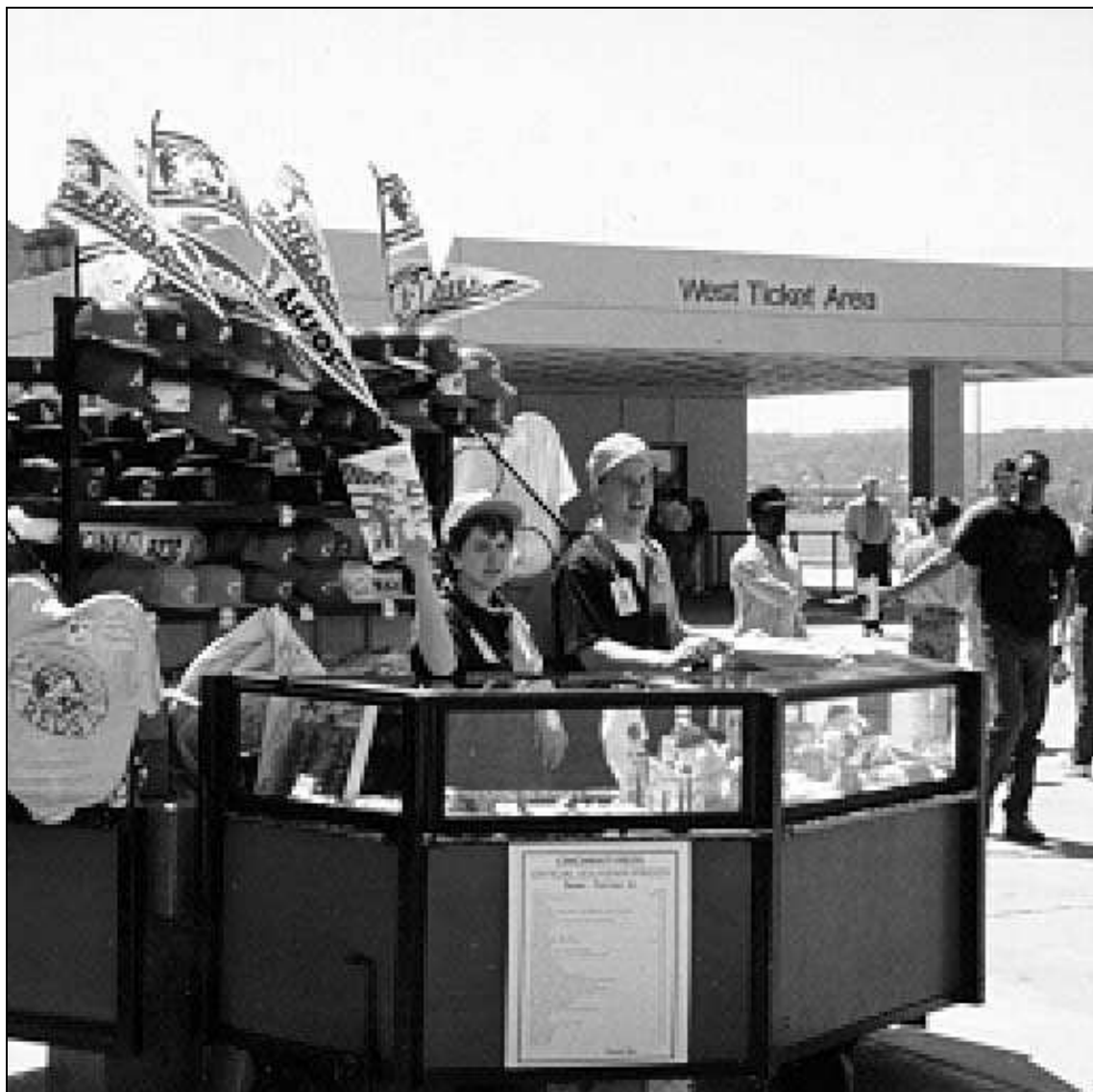
As modificações de uma imagem descritas anteriormente visam predominantemente facilitar a visualização e, via de regra, não representam alterações nas dimensões reais da imagem. Ou seja, quando dissermos que um pixel é multiplicado por 4, o leitor deve entender que um pixel da imagem que originalmente ocuparia um ponto do dispositivo de exibição passará a ocupar quatro pontos, sem afetar em nada as dimensões reais da imagem.

Caso nosso objetivo seja a alteração de dimensões da imagem, as técnicas descritas acima podem ser utilizadas, com a diferença de que a saída será o arquivo contendo a imagem alterada (ampliada/reduzida) e não o resultado visual de sua exibição em maior ou menor tamanho na tela. A literatura técnica de processamento de imagens por vezes distingue dois tipos de alterações de dimensões de uma imagem, embora tecnicamente idênticos:

1. o processo denominado *scaling* refere-se ao caso em que a imagem é ampliada ou reduzida por um fator (que pode ser igual para as dimensões horizontal e vertical – preservando a relação de aspecto original – ou não);
2. o nome *sizing* (algumas vezes *resizing*) é utilizado nos casos em que, ao invés de especificar o fator de ampliação / redução, o usuário especifica o novo tamanho que a imagem deve possuir.



(a)



(b)

Figura 24 - (a) imagem original, (b) imagem ampliada (*zoom in*) de 2 vezes; (c) imagem reduzida (*zoom out*) de 2 vezes.



(c)

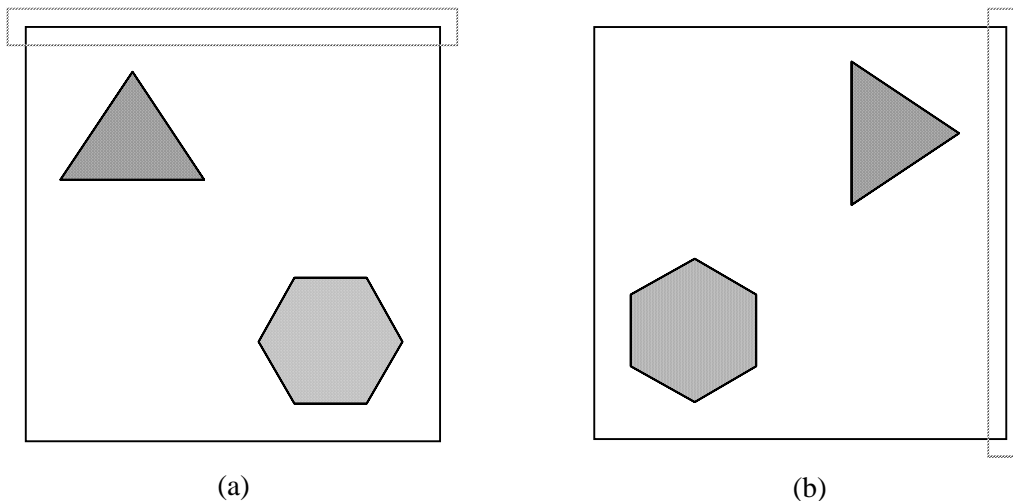
Figura 24 - Continuação.

2.5.3 Translação

A translação de uma imagem consiste basicamente no deslocamento linear de cada pixel de coordenadas (X, Y) na horizontal e/ou na vertical, mapeando para o ponto de coordenadas (X', Y') , calculadas como: $(X', Y') = (X + \Delta X, Y + \Delta Y)$, onde ΔX é o deslocamento vertical e ΔY é o deslocamento horizontal (em pixels).

2.5.4 Rotação

Uma imagem pode ser rotacionada de um ângulo arbitrário, tanto no sentido horário quanto no anti-horário. Rotações com ângulos múltiplos de 90° são mais simples de implementar, pois consistem na cópia de pixels que estão organizados em linhas, reordenando-os em colunas na direção em que se deseja rotacionar a imagem. A figura 25 ilustra o processo de rotação de 90° no sentido horário. A área tracejada destaca as primeiras linhas da imagem original, que são reposicionadas em formas de colunas, da direita para a esquerda, na imagem rotacionada.



(a)

(b)

Figura 25 - Exemplo de rotação de 90° no sentido horário.

A rotação por ângulos quaisquer é uma tarefa mais complexa, que pode ser implementada usando as técnicas de *warping* descritas adiante. Matematicamente, a rotação de cada ponto (X, Y) de uma imagem por um ângulo arbitrário Ang , mapeará este ponto na localidade de coordenadas (X', Y') , onde X' e Y' são calculados pelas equações:

$$X' = X \cos(Ang) + Y \sin(Ang) \quad (2.10)$$

$$Y' = Y \cos(Ang) - X \sin(Ang) \quad (2.11)$$

O processo de rotação normalmente exige a correção de razão de aspecto da imagem resultante, tendo em vista que diversos modos de exibição de vídeo utilizam pixels não-quadrados.

2.5.5 Espelhamento (*Flip*)

O espelhamento (*flip*) é uma operação que combina a rotação por ângulos múltiplos de 90° com o cálculo de matriz transposta. Um *flip* horizontal nada mais é que uma rotação de 90° no sentido anti-horário (ou 270° no sentido horário) da versão transposta da imagem, enquanto um *flip* vertical é uma rotação de 90° no sentido horário (ou 270° no sentido anti-horário) da versão transposta da imagem. A figura 26 mostra exemplos de *flip* horizontal e vertical.

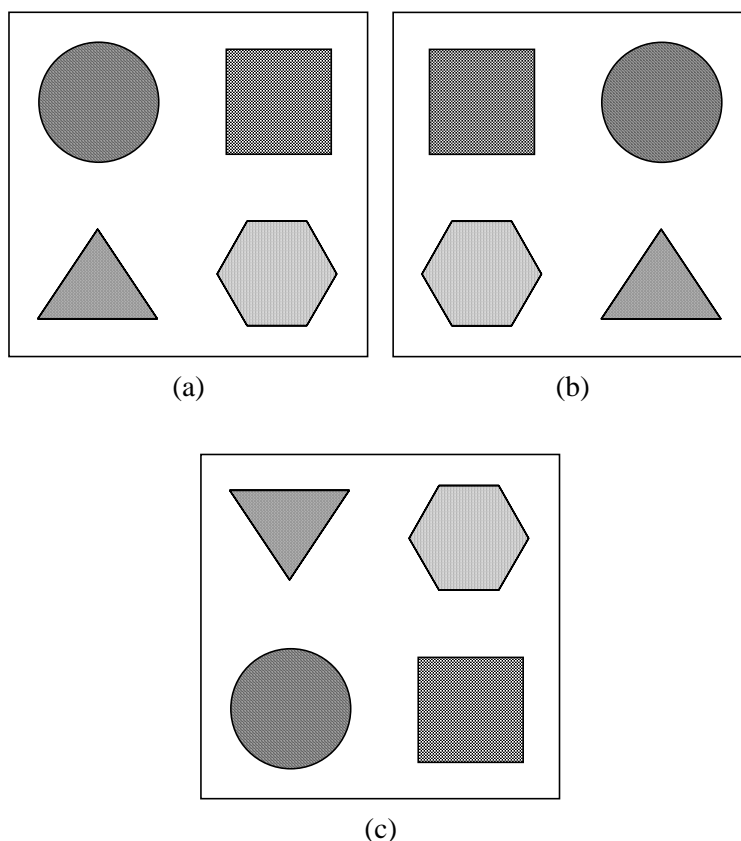


Figura 26 - (a) Imagem original, (b) *flip* horizontal, (c) *flip* vertical.

2.5.6 *Warping*

Warping é o nome dado ao processo de alteração de uma imagem de tal modo que a relação espacial entre seus objetos e características é alterada conforme outra imagem ou gabarito (*template*).

A transformação matemática mais comum é baseada na projeção afim (*affine projection*), dada pelas equações:

$$X' = \frac{aX + bY + c}{iX + jY + 1} \quad (2.12)$$

$$Y' = \frac{dX + eY + f}{iX + jY + 1} \quad (2.13)$$

onde X e Y são as coordenadas antigas e X' e Y' as novas. Os coeficientes a, b, c, d, e, f, i e j são determinados a partir de um conjunto de pontos de controle que correspondem à congruência

desejada entre as duas imagens ou entre a imagem original e o template selecionado. A figura 27 mostra um exemplo do processo de *warping* aplicado a uma imagem binária simples.

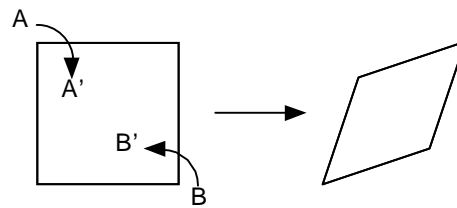
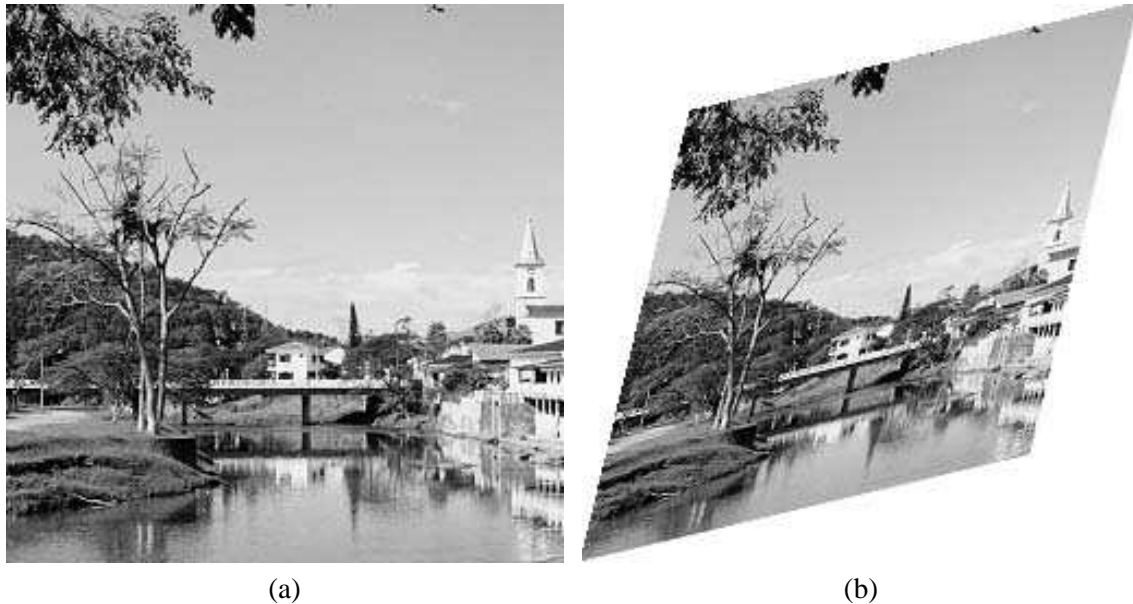


Figura 27 - Exemplo de *warping*.

Neste exemplo, o ponto A é movido para a posição A' e B é movido para B' . A projeção afim exige que sejam selecionados quatro pares de pontos de controle, para resultar um sistema de oito equações a oito incógnitas. Os pares $A-A'$ e $B-B'$ são duas escolhas óbvias. Os outros dois pontos escolhidos, neste exemplo, são os dois cantos restantes do quadrado (que permanecerão inalterados). Se tivéssemos selecionado mais de quatro pontos de controle, um ajuste por mínimos quadrados seria necessário para determinar os melhores valores para a transformação.

Na prática, um programa para a solução simultânea de um sistema de equações é utilizado para calcular os valores dos coeficientes. Então, entrando com as coordenadas X' e Y' da imagem destino, calcula-se os valores correspondentes de X e Y na imagem original. O nível de cinza do ponto de coordenadas (X,Y) é então atribuído à posição (X',Y') na imagem destino. Este processo de mapeamento pode ser facilmente executado em paralelo, pois cada ponto na imagem resultante depende de apenas um ponto da imagem original.

A figura 28 mostra um exemplo de *warping* utilizando imagem monocromática.



(a)

(b)

Figura 28 - Exemplo de *warping* de uma imagem monocromática utilizando padrão (*template*) em forma de losango.

Exercício resolvido

Baseando-se na figura 27, dadas as coordenadas originais dos vértices do quadrado e as coordenadas desejadas para o quadrado após o *warping*, indicadas na tabela a seguir e ilustradas na figura 29, calcular os valores dos coeficientes a , b , c , d , e , f , i e j correspondentes à transformação desejada.

Ponto de controle	Coordenadas originais (X,Y)	Coordenadas após <i>warping</i> (X',Y')
1	(0,0)	(2,2)
2	(4,4)	(3,3)
3	(4,0)	(4,0)
4	(0,4)	(0,4)

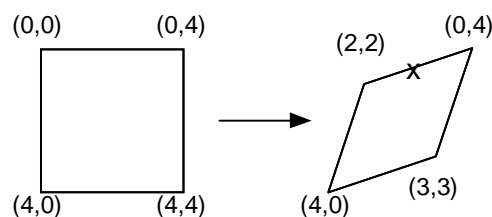


Figura 29 - Coordenadas dos pontos de controle antes e depois do *warping*.

Solução:

Entrando com estes valores para as coordenadas e resolvendo o sistema de equações resultante (eqs. 2.12 e 2.13), obtemos os seguintes valores para os coeficientes:

$$\begin{array}{llll}
 a = 0,75 & b = -0,25 & c = 1,00 & d = -0,25 \\
 e = 0,75 & f = 1,00 & i = 0,00 & j = 0,00
 \end{array}$$

Para verificar se os coeficientes calculados estão corretos, podemos escolher um ponto na imagem modificada, por exemplo o ponto de coordenadas (0,5, 2,5), indicado com uma cruz na figura 29. Calculando os valores de X e Y correspondentes a este ponto na imagem original, obteremos o par (0, 2), como esperado.

|

2.5.7 Cropping, cutting e pasting

Recortar e colar trechos de imagens para compor novas imagens são operações corriqueiras de manipulação de imagens. Existem três formas de se recortar uma imagem. A primeira e mais simples consiste em utilizar uma região retangular, definida pelas coordenadas de dois de seus vértices. A segunda consiste em utilizar uma figura geométrica regular qualquer ou um polígono, regular ou não. A terceira e mais complexa consiste em se permitir delimitar a área de recorte 'à mão livre' utilizando o mouse ou dispositivo equivalente. O detalhamento de tais operações, por se enquadrarem mais no contexto de manipulação de imagens, foge ao objetivo do texto.

Leitura complementar

Para maiores detalhes sobre os aspectos de interpolação necessários à implementação de diversas transformações geométricas apresentadas nesta seção, recomendamos as subseções 4.3.2 e 14.5.1 de [Pratt 1991] e o capítulo 12 de [Lindley 1991].

Dawson [Dawson 1987] traz fragmentos de código em C para a execução de transformações geométricas sobre imagens, particularmente rotação e *resizing*. Em outro artigo

Técnicas de Modificação de Histograma

Este capítulo é inteiramente dedicado à definição e utilização do conceito de histograma. A Seção 3.1 conceitua histograma e dá exemplos de histogramas de imagens. Na Seção 3.2 uma técnica ponto a ponto de processamento de imagens, a transformação de intensidade, é apresentada. As seções 3.3 a 3.5 apresentam técnicas de modificação de histograma de imagens monocromáticas. Finalmente, a Seção 3.6 introduz o conceito de limiarização (*thresholding*) de imagens, como exemplo de utilização da informação contida em um histograma.

3.1 Conceito de histograma

O histograma de uma imagem é simplesmente um conjunto de números indicando o percentual de pixels naquela imagem que apresentam um determinado nível de cinza. Estes valores são normalmente representados por um gráfico de barras que fornece para cada nível de cinza o número (ou o percentual) de pixels correspondentes na imagem. Através da visualização do histograma de uma imagem obtemos uma indicação de sua qualidade quanto ao nível de contraste e quanto ao seu brilho médio (se a imagem é predominantemente clara ou escura).

Cada elemento deste conjunto é calculado como:

$$p_r(r_k) = \frac{n_k}{n} \quad (3.1)$$

onde:

$$0 \leq r_k \leq 1$$

$k = 0, 1, \dots, L-1$, onde L é o número de níveis de cinza da imagem digitalizada;

n = número total de pixels na imagem;

$p_r(r_k)$ = probabilidade do k -ésimo nível de cinza;

n_k = número de pixels cujo nível de cinza corresponde a k .

Exemplo

Os dados da tabela 1 correspondem a uma imagem de 128 x 128 pixels, com 8 níveis de cinza. O número de pixels correspondentes a um certo tom de cinza está indicado na segunda coluna, enquanto as respectivas probabilidades $p_r(r_k)$ aparecem na terceira coluna. A representação gráfica equivalente deste histograma é mostrada na figura 1.

Um histograma apresenta várias características importantes. A primeira delas é que cada $p_r(r_k)$ fornece, como sugere a notação, a probabilidade de um pixel da imagem apresentar nível de cinza r_k . Portanto, um histograma nada mais é que uma função de distribuição de probabilidades e como tal deve obedecer aos axiomas e teoremas da teoria de probabilidade. Por exemplo, é possível verificar que na tabela 1 a soma dos valores de $p_r(r_k)$ é 1, o que já era esperado.

Tabela 1 - Exemplo de histograma.

Nível de cinza (r_k)	n_k	$p_r(r_k)$
0	1120	0,068
1/7	3214	0,196
2/7	4850	0,296
3/7	3425	0,209
4/7	1995	0,122
5/7	784	0,048
6/7	541	0,033
1	455	0,028
Total	16384	1



Figura 1 - Exemplo de histograma para imagem com oito níveis de cinza.

A figura 2 apresenta cinco exemplos de tipos de histogramas freqüentemente encontrados em imagens. O histograma da figura 2(a) apresenta grande concentração de pixels nos valores mais baixos de cinza, correspondendo a uma imagem predominantemente escura. Na figura 2(b) os pixels estão concentrados em valores próximos ao limite superior da escala de cinza, caracterizando uma imagem clara. Na parte (c) da figura, os pixels estão agrupados em torno de valores intermediários de cinza, correspondendo a uma imagem de brilho médio. Nas figuras (a), (b) e (c) a maioria dos pixels está concentrada em uma estreita faixa da escala de cinza, significando que as imagens correspondentes apresentam baixo contraste. A figura 2(d) corresponde a uma imagem com pixels distribuídos ao longo de toda a escala de cinza. É comum dizer que uma imagem com estas características apresenta um bom contraste. A figura 2(e) mostra um histograma tipicamente bimodal, isto é, apresentando duas concentrações de pixels, uma delas em torno de valores escuros e outra na região clara do histograma. Pode-se dizer que a imagem correspondente apresenta alto contraste entre as duas concentrações, uma vez que elas se encontram razoavelmente espaçadas.¹

Para verificar a relação entre imagens e respectivos histogramas, a figura 3 mostra cinco imagens monocromáticas cujos histogramas são aqueles da figura 2.

¹ Convém observar que os conceitos de alto e baixo contraste neste caso somente estão relacionados ao espaçamento médio entre as raías do histograma. Já o termo 'bom contraste' deve ser utilizado com cautela, pois pode exprimir distribuição equitativa das raías ao histograma ao longo da escala de cinza – como foi utilizado neste caso – ou uma opinião subjetiva sobre a qualidade de uma imagem, que não poderia ser extraída somente da observação de seu histograma.

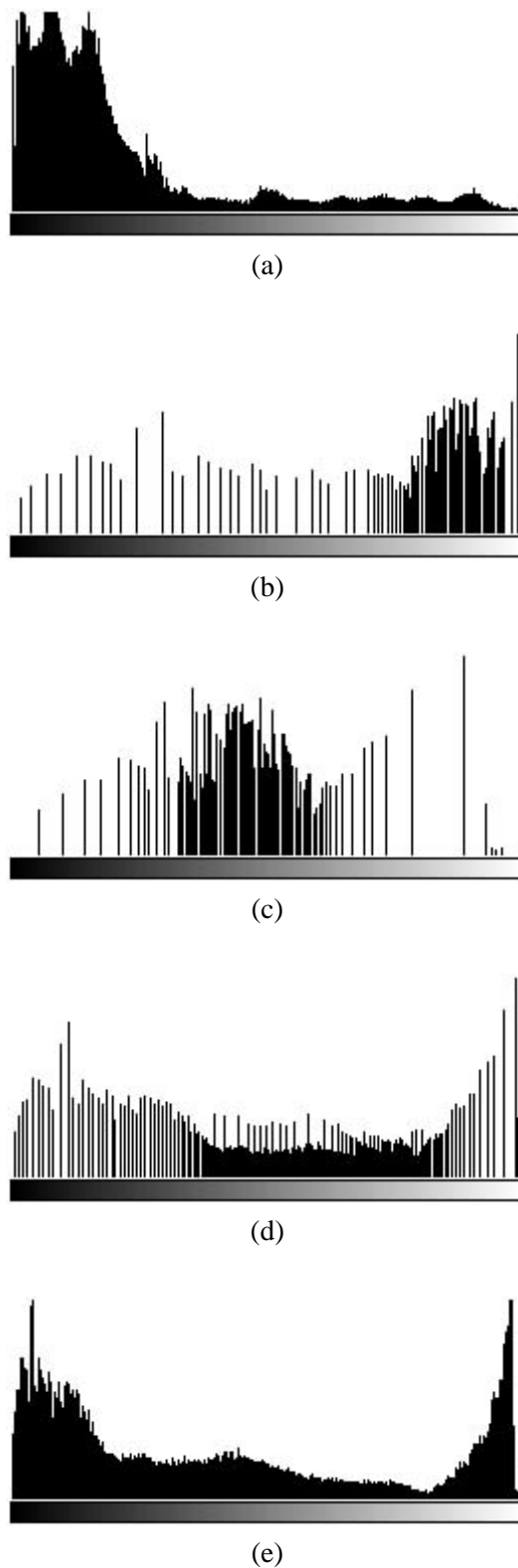


Figura 2 - Exemplos de histogramas.

O conceito de histograma também é aplicável a imagens coloridas. Neste caso, a imagem é decomposta de alguma forma (por exemplo, em seus componentes R, G e B) e para cada componente é calculado o histograma correspondente. A figura 4 (ver Seção *Figuras Coloridas*) mostra um exemplo de imagem colorida e seus histogramas R, G e B.

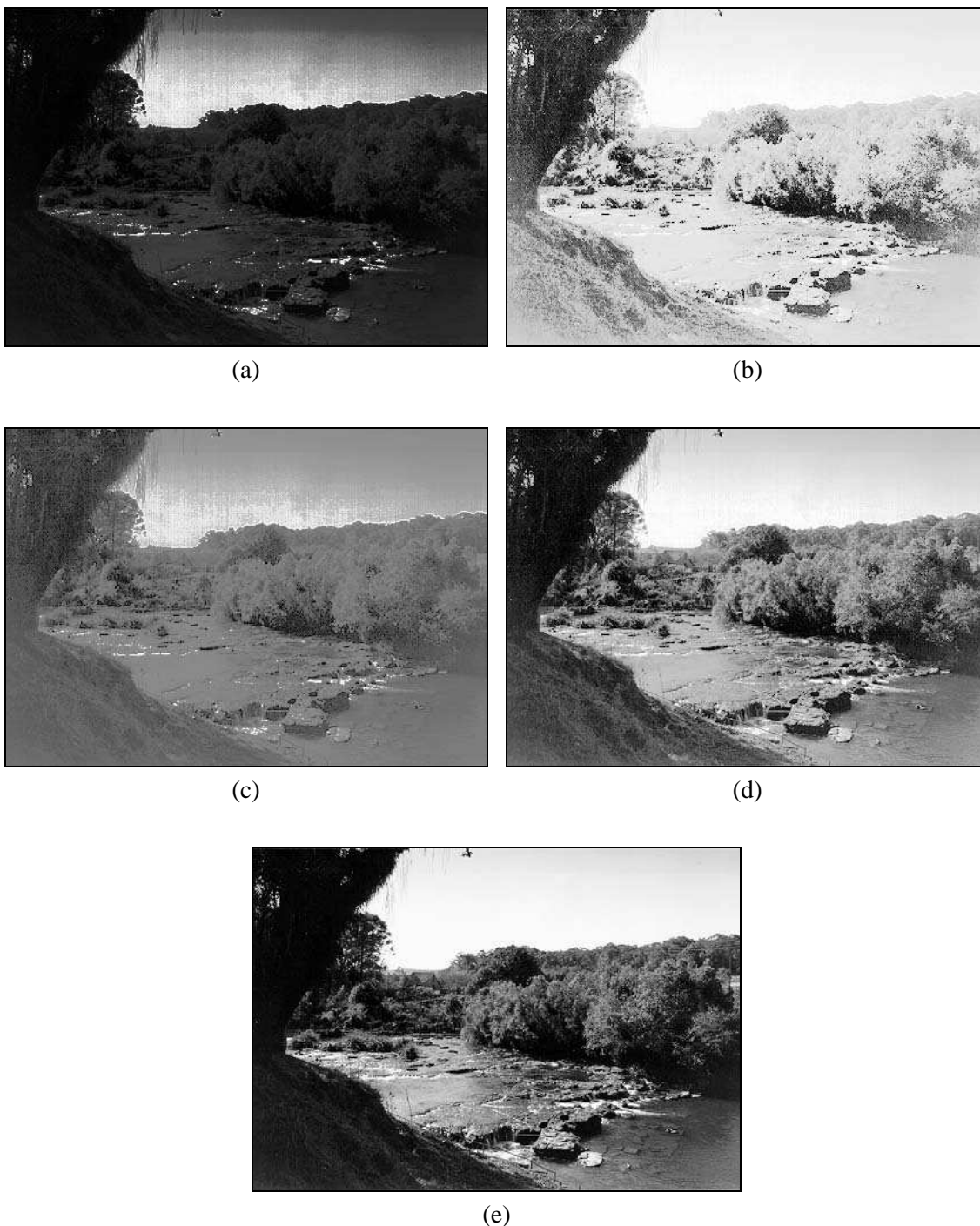


Figura 3 - Imagens correspondentes aos histogramas da figura 2.

Para computar o histograma de uma imagem monocromática, inicializa-se com zero todos os elementos de um vetor de L elementos, onde L é o número de tons de cinza possíveis. Em seguida, percorre-se a imagem, pixel a pixel, e incrementa-se a posição do vetor cujo índice corresponde ao tom de cinza do pixel visitado. Após toda a imagem ter sido percorrida, cada elemento do vetor conterá o número de pixels cujo tom de cinza equivale ao índice do elemento. Estes valores poderão ser normalizados, dividindo cada um deles pelo total de pixels na imagem.

Concluindo esta Seção, convém enfatizar que, embora o histograma de uma imagem forneça diversas informações qualitativas e quantitativas sobre ela (e.g. nível de cinza mínimo, médio e máximo, predominância de pixels claros ou escuros etc.), outras conclusões de caráter qualitativo (e.g. qualidade subjetiva global da imagem, presença ou não de ruído etc.) somente

podem ser extraídas dispondo-se da imagem propriamente dita. Tal fato pode ser confirmado a partir de uma análise das figuras 2 e 3.

Leitura complementar

Para uma revisão da teoria básica de probabilidade, sugerimos [Ross 1994].

O capítulo 9 de [Lindley 1991] e o artigo de Dawson [Dawson 1987] apresentam código-fonte em C para cálculo e exibição de histogramas de imagens monocromáticas.

O capítulo 3 de [Pavlidis 1982] apresenta algoritmos para obtenção e equalização do histograma de uma imagem monocromática.

3.2 Transformações de intensidade

As técnicas de modificação de histograma são conhecidas como técnicas ponto-a-ponto, uma vez que o valor de tom de cinza de um certo pixel após o processamento depende apenas de seu valor original. Em contraste, nas técnicas de processamento orientadas a vizinhança, o valor resultante depende também, de alguma forma, dos pixels que circundam o elemento de imagem original.

Diversas técnicas de modificação da distribuição dos pixels na escala de cinza podem ser implementadas a partir do conceito de transformações de intensidade, apresentado formalmente a seguir.

Seja uma variável f , representando o nível de cinza dos pixels na imagem a ser processada. Por simplicidade, assumiremos inicialmente que a escala de cinza é normalizada, ou seja,

$$0 \leq f \leq 1,$$

onde $f = 0$ representa um pixel preto e $f = 1$ indica pixel branco.

Para qualquer f no intervalo $[0, 1]$, denominaremos transformações de intensidade as funções do tipo

$$g = T(f) \tag{3.2}$$

que mapearão cada pixel de tom de cinza f da imagem original em um novo tom de cinza, g , na imagem destino. Estas funções devem satisfazer duas condições:

- (i) devem retornar um único valor para cada valor distinto de f e devem crescer monotonicamente no intervalo $0 \leq f \leq 1$
- (ii) $0 \leq T(f) \leq 1$ para $0 \leq f \leq 1$.

Um exemplo de função que satisfaz estes critérios é dado na figura 5. O efeito desta transformação não-linear de intensidade sobre a imagem é um aumento de seu contraste.

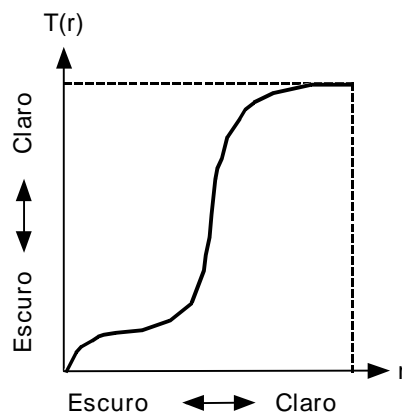


Figura 5 - Exemplo de transformação de intensidade.

As transformações de intensidade podem ser lineares ou não-lineares. As transformações lineares podem ser genericamente descritas pela equação:

$$g = c \cdot f + b \quad (3.3)$$

onde o parâmetro c controla o contraste da imagem resultante, enquanto b ajusta seu brilho. A figura 6 apresenta diversos exemplos de transformações lineares e seus respectivos valores de c e b .

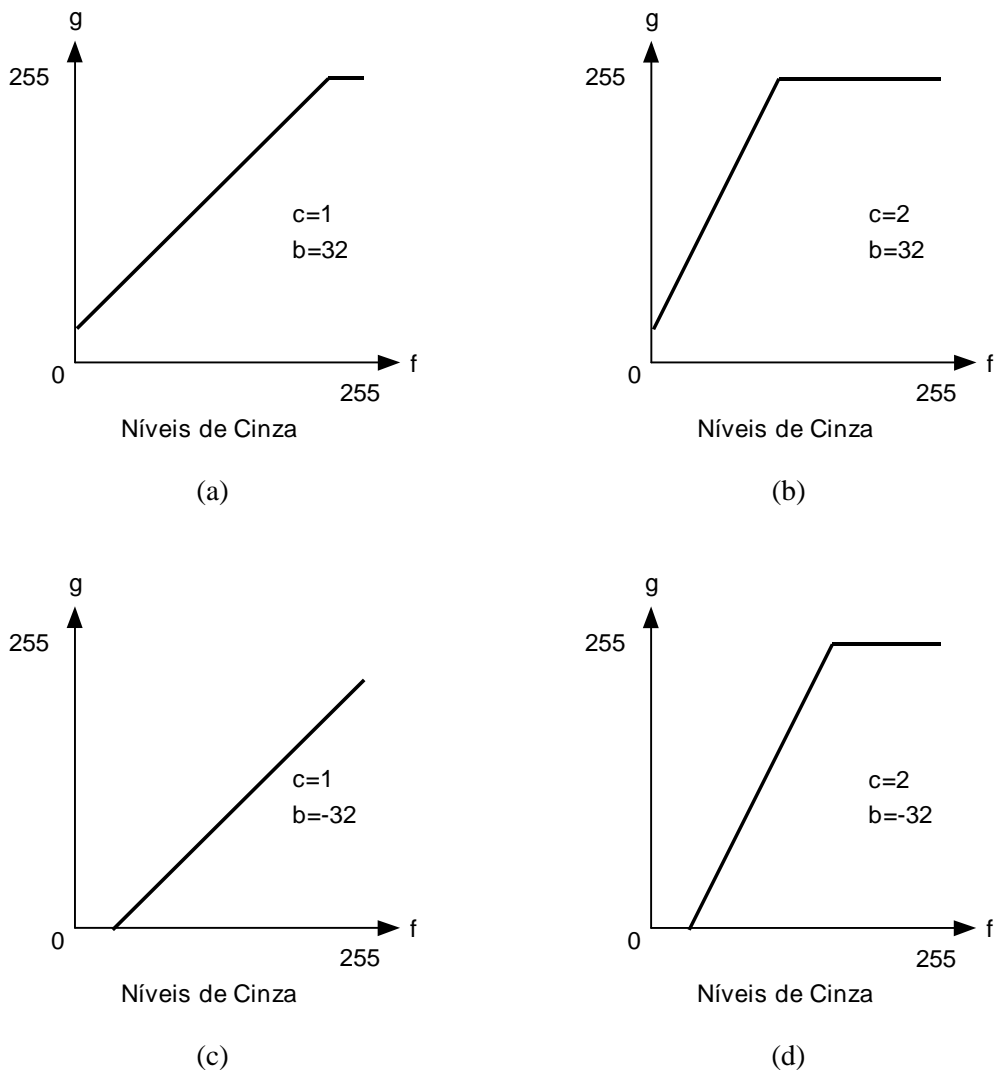


Figura 6 - Exemplos de transformações de intensidade lineares.

As transformações não-lineares podem ser descritas por equações tais como:

$$g = 31,875 \cdot \log_2 (f + 1) \quad (3.4)$$

produzindo o resultado mostrado na figura 7. Nos aplicativos para processamento de imagens disponíveis atualmente, freqüentemente estas transformações são especificadas de forma interativa pelo usuário, utilizando o mouse ou dispositivo equivalente e 'desenhando' a curva desejada.

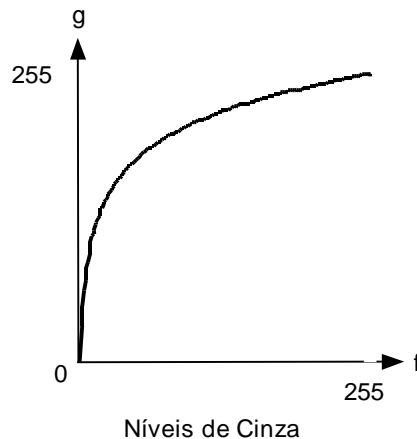


Figura 7 - Exemplo de transformação de intensidade não-linear.

O conceito de transformação de intensidade linear pode ser utilizado para implementar uma função que automaticamente expande a escala de tons de cinza de uma imagem para que ela ocupe todo o intervalo possível. Esta função recebe o nome de autoescala. Para um sistema que opera com imagens com 256 níveis de cinza, uma função de autoescala pode ser implementada calculando, para cada pixel com tom de cinza f , o nível de cinza resultante g , pela equação:

$$g = \frac{255}{f_{\max} - f_{\min}} (f - f_{\min}), \quad (3.5)$$

onde f_{\max} e f_{\min} são, respectivamente, os níveis máximo e mínimo de cinza presentes na imagem original.

Leitura complementar

O capítulo 9 de [Lindley 1991] e o artigo de Dawson [Dawson 1987] apresentam código-fonte em C para cálculo de algumas transformações ponto-a-ponto discutidas nesta Seção.

O capítulo 5 de [Galbiati, Jr. 1990] contém inúmeros exemplos de funções de transformação de intensidade.

3.3 Equalização de histograma

A equalização de histograma é uma técnica a partir da qual se procura redistribuir os valores de tons de cinza dos pixels em uma imagem, de modo a obter um histograma uniforme, no qual o número (percentual) de pixels de qualquer nível de cinza é praticamente o mesmo. Para tanto, utiliza-se uma função auxiliar, denominada função de transformação. A forma mais usual de se equalizar um histograma é utilizar a função de distribuição acumulada (*cdf* - *cumulative distribution function*) da distribuição de probabilidades original, que pode ser expressa por:

$$s_k = T(r_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p_r(r_j) \quad (3.6)$$

onde:

$$0 \leq r_k \leq 1$$

$$k = 0, 1, \dots, L-1$$

A inversa desta função é dada por:

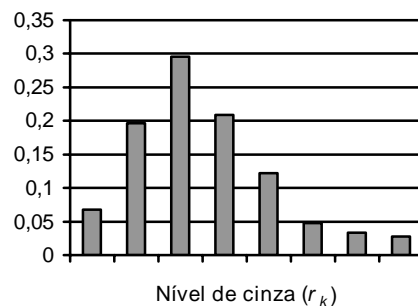
$$r_k = T^{-1}(s_k) \quad p/0 \leq s_k \leq 1 \quad (3.7)$$

e embora ela não seja necessária no processo de equalização de histograma, será importante no método descrito na Seção seguinte. Convém notar que outras funções de transformação, que não a *cdf*, podem ser especificadas.

Exercício resolvido

Seja o histograma da tabela 1, ilustrado graficamente na figura 1, ambas reproduzidas a seguir para maior facilidade. Equalizá-lo utilizando a função de distribuição acumulada e plotar o histograma resultante.

Tabela 1 - Histograma original		
Nível de cinza (r_k)	n_k	$p_r(r_k)$
0	1120	0,068
1/7	3214	0,196
2/7	4850	0,296
3/7	3425	0,209
4/7	1995	0,122
5/7	784	0,048
6/7	541	0,033
1	455	0,028
Total	16384	1



Solução:

Utilizando a *cdf* como função de transformação, calculamos:

$$\begin{aligned}
 s_0 &= T(r_0) = \sum_{j=0}^0 p_r(r_j) \\
 &= p_r(r_0) \\
 &= 0,068
 \end{aligned}$$

De forma similar,

$$\begin{aligned}
 s_1 &= T(r_1) = \sum_{j=0}^1 p_r(r_j) \\
 &= p_r(r_0) + p_r(r_1) \\
 &= 0,264
 \end{aligned}$$

$$\begin{array}{lll}
 s_2 = 0,560 & s_3 = 0,769 & s_4 = 0,891 \\
 s_5 = 0,939 & s_6 = 0,972 & s_7 = 1.
 \end{array}$$

Esta função está plotada na figura 8.

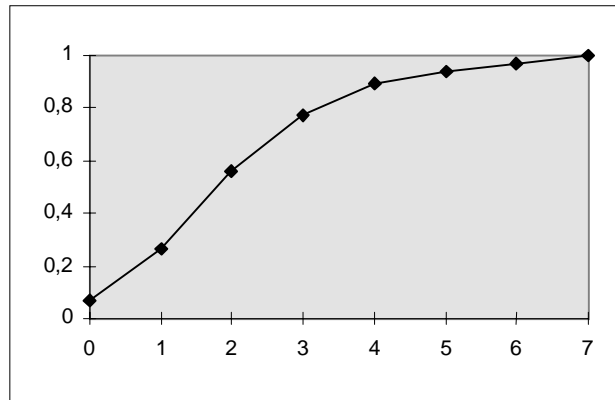


Figura 8 - Função de transformação utilizada para a equalização.

Como a imagem foi quantizada com apenas 8 níveis de cinza, cada valor s_j deverá ser arredondado para o valor válido (múltiplo de $1/7$) mais próximo. Desta forma,

$$\begin{array}{llll}
 s_0 \approx 0 & s_1 \approx 2/7 & s_2 \approx 4/7 & s_3 \approx 5/7 \\
 s_4 \approx 6/7 & s_5 \approx 1 & s_6 \approx 1 & s_7 \approx 1.
 \end{array}$$

Concluindo o mapeamento, verificamos que o nível original $r_0 = 0$ foi mapeado para $s_0 = 0$ e portanto a raia correspondente não sofreu alteração. Já os 3214 pixels que apresentavam tom de cinza $1/7$ foram remapeados para $s_1 = 2/7$. Similarmente, os pixels com tom de cinza $2/7$ foram modificados para $4/7$, aqueles com $r = 3/7$ passaram a $5/7$ e os de $4/7$ mapearam em $6/7$. Convém observar, entretanto, que as três raiais correspondentes aos pixels com tons de cinza $5/7$, $6/7$ e 1 foram somadas em uma só raia, com tom de cinza máximo, isto é, 1 .

Agrupando os resultados na tabela 2, teremos o histograma após a equalização, mostrado graficamente na figura 9.

Tabela 2 - Histograma equalizado

Nível de cinza (s_k)	n_k	$p_s(s_k)$
0	1120	0,068
1/7	0	0,000
2/7	3214	0,196
3/7	0	0,000
4/7	4850	0,296
5/7	3425	0,209
6/7	1995	0,122
1	1780	0,109
Total	16384	1

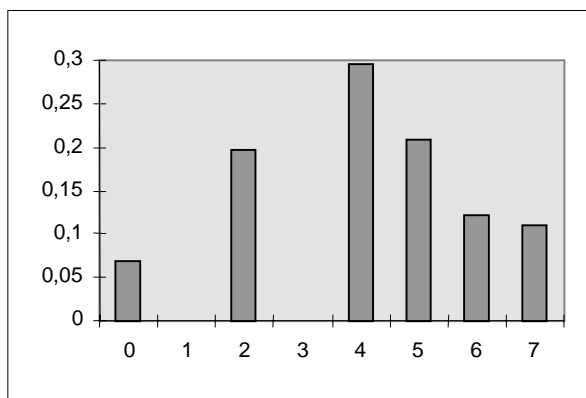


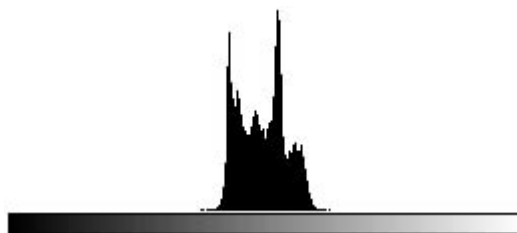
Figura 9 - Histograma equalizado.

Pode-se notar que o histograma equalizado, apesar de estar longe de ser perfeitamente plano, apresenta melhor distribuição de pixels ao longo da escala de cinza em relação ao original.

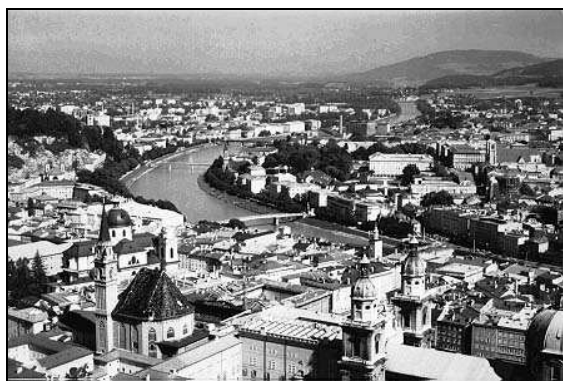
A figura 10 apresenta um exemplo de aplicação da técnica de equalização de histograma para aumentar o contraste de uma imagem 446 x 297 com 256 tons de cinza. A parte (a) apresenta a imagem original, cujo histograma é plotado na figura 10(c). A parte (d) mostra o histograma equalizado, correspondente à imagem da figura 10(b).



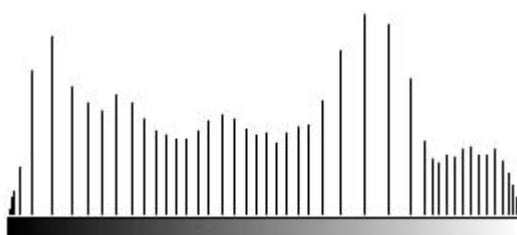
(a)



(b)



(c)



(d)

Figura 10 - Aplicação da equalização de histograma a imagens com baixo contraste.

As técnicas de obtenção e equalização de histogramas também podem ser aplicadas a trechos de imagens, por exemplo, janelas $n \times m$. Estas técnicas locais servem principalmente para realçar detalhes sutis de pequenas porções da imagem.

3.4 Especificação direta de histograma

Apesar de sua grande utilização em situações de aprimoramento de contraste de imagens, a equalização de histograma apresenta como principal limitação o fato de não permitir a especificação de nenhum parâmetro, a não ser a função de transformação, que, como vimos na Seção anterior, costuma ser a *cdf* da distribuição de probabilidade original. Existem situações, entretanto, em que seria desejável poder especificar que tipo de mudança se deseja sobre o histograma. Nestes casos, uma das possíveis técnicas é a especificação direta de histograma.

Dada uma imagem (e seu histograma original) e o novo histograma desejado, o procedimento da especificação direta de histograma consiste em:

- 1) equalizar os níveis da imagem original usando a *cdf* discreta:

$$s_k = T(r_k) = \sum_{j=0}^k \frac{n_j}{n} = \sum_{j=0}^k p_r(r_j) \quad (3.8)$$

- 2) equalizar a função densidade de probabilidade discreta (isto é, o histograma) desejada(o):

$$v_k = G(z_k) = \sum_{j=0}^k p_z(z_j) \quad (3.9)$$

- 3) aplicar a função de transformação inversa:

$$z = G^{-1}(s) \quad (3.10)$$

aos níveis obtidos no passo 1.

Exercício resolvido

Seja novamente o histograma da tabela 1. Deseja-se modificar este histograma de modo que a distribuição de pixels resultante seja aquela da tabela 3, a seguir.

Tabela 3 - Histograma desejado

Nível de cinza (z_k)	n_k	$p_z(z_k)$
0	0	0,000
1/7	0	0,000
2/7	0	0,000
3/7	1638	0,100
4/7	3277	0,200
5/7	6554	0,400
6/7	3277	0,200
1	1638	0,100
Total	16384	1

Solução:

O histograma após equalização já foi calculado em exercício resolvido anterior e seus resultados estão na tabela 2.

O próximo passo consiste em obter a *cdf* da distribuição de probabilidade desejada. Seguindo o mesmo raciocínio utilizado para o cálculo da *cdf* do histograma original, encontramos:

$$\begin{array}{llll} v_0 = 0 & v_1 = 0 & v_2 = 0 & v_3 = 0,1 \\ v_4 = 0,3 & v_5 = 0,7 & v_6 = 0,9 & v_7 = 1. \end{array}$$

O último passo – e o mais difícil de entender quando se estuda este assunto pela primeira vez – é a obtenção da inversa. Como estamos lidando com níveis discretos, a obtenção da função inversa consistirá basicamente em procurar, para cada valor de s_k , o valor de v_k que mais se aproxima dele. Por exemplo, o valor de v_k que mais se aproxima de $s_1 = 2/7 \approx 0,286$ é $G(z_4) = 0,3$ ou seja, $G^{-1}(0,3) = z_4$. Portanto, os pixels que após a equalização do histograma original foram reposicionados no tom de cinza s_1 serão mapeados para o tom de cinza z_4 . Em outras palavras, os 3214 pixels que apresentavam originalmente tom de cinza $1/7$ e que foram remapeados para $s_1 = 2/7$ devido à equalização, serão trasladados novamente para $z_4 = 4/7$ por força da especificação direta de histograma. Procedendo de forma similar para os demais valores de s_k , teremos:

$$\begin{array}{ll} s_0 = 0 \rightarrow z_2 & s_1 = 2/7 \approx 0,286 \rightarrow z_4 \\ s_2 = 4/7 \approx 0,571 \rightarrow z_5 & s_3 = 5/7 \approx 0,714 \rightarrow z_5 \\ s_4 = 6/7 \approx 0,857 \rightarrow z_6 & s_5 = 1 \rightarrow z_7 \\ s_6 = 1 \rightarrow z_7 & s_7 = 1 \rightarrow z_7 \end{array}$$

Neste caso, assumimos que o algoritmo de cálculo da inversa, para um dado valor de s_k , percorreria os diversos valores de v_k , armazenando o índice do último valor que tenha resultado na menor diferença encontrada. Se o algoritmo possuir outra forma de solucionar 'empates', o nível s_0 poderá mapear em z_0 ou z_1 . A tabela 4 resume os histogramas original e desejado, suas respectivas *cdfs* e o processo de mapeamento descrito acima.

Tabela 4 - Resumo da especificação direta de histograma

k	$p_r(r_k)$	s_k		v_k	$p_z(z_k)$
0	0,068	0		0,00	0,000
1	0,196	2/7		0,00	0,000
2	0,296	4/7		0,00	0,000
3	0,209	5/7		0,10	0,100
4	0,122	6/7		0,30	0,200
5	0,048	1		0,70	0,400
6	0,033	1		0,90	0,200
7	0,028	1		1,00	0,100

A tabela 5 apresenta os valores obtidos para o histograma resultante. Para uma comparação visual entre o histograma desejado e o obtido, plotamos cada um deles nas figuras 11 e 12, respectivamente.

Tabela 5 - Histograma obtido

z_k	$p_z(z_k)$
0	0,000
1/7	0,000
2/7	0,068
3/7	0,000
4/7	0,196
5/7	0,505
6/7	0,122
1	0,109
Total	1

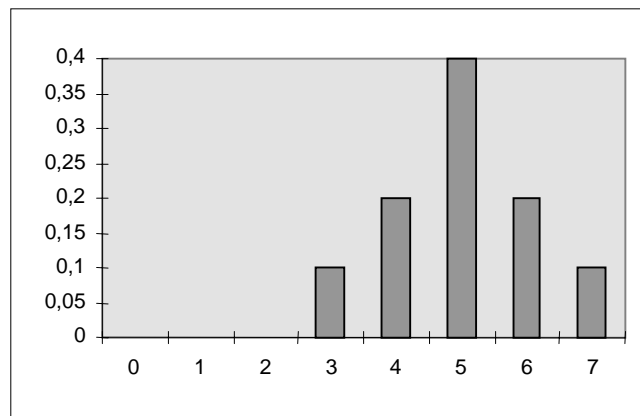


Figura 11 - Histograma desejado.

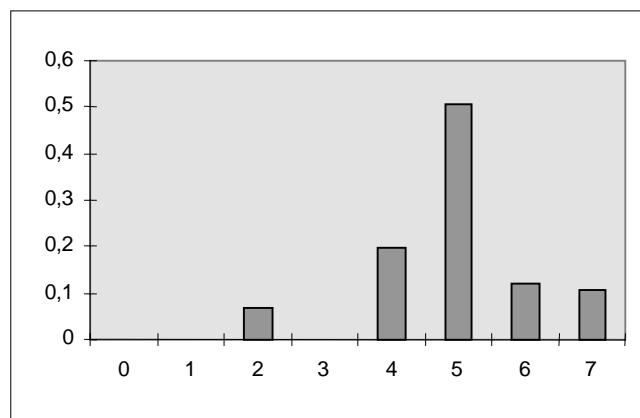


Figura 12 - Histograma obtido.

Pode-se notar que o histograma obtido aproxima-se, dentro do possível, do histograma desejado.

A figura 13 apresenta um exemplo de aplicação da técnica de especificação direta de histograma aplicada a uma imagem 443 x 298 com 256 tons de cinza. A parte (a) apresenta a imagem original, cujo histograma é plotado na figura 13(c). A parte (d) mostra o histograma desejado, enquanto a figura 13(e) mostra o histograma obtido, que corresponde à imagem da figura 13(b).

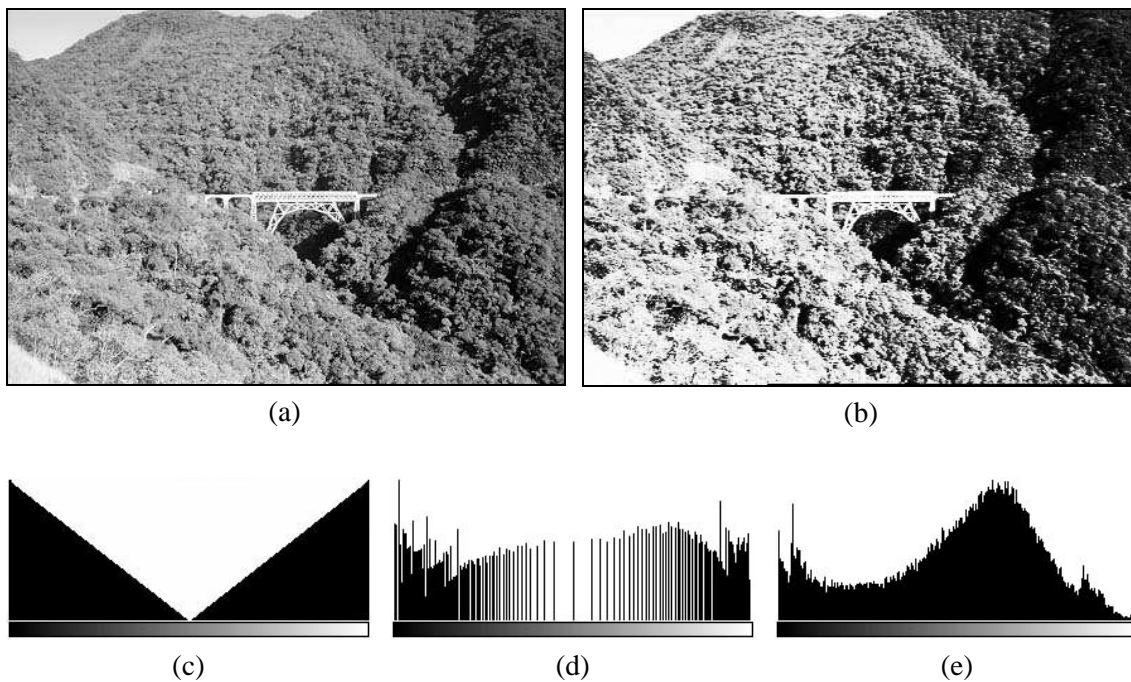


Figura 13 - Exemplo de aplicação da especificação direta de histograma.

Leitura complementar

Em [Woods e Gonzalez 1981] são descritas algumas técnicas de modificação de histograma, aplicadas em um sistema de realce de imagens em tempo real.

3.5 Outras técnicas

Existem inúmeras outras técnicas de processamento de imagens a partir de modificações de seus respectivos histogramas. Apresentamos a seguir os principais aspectos de algumas delas.

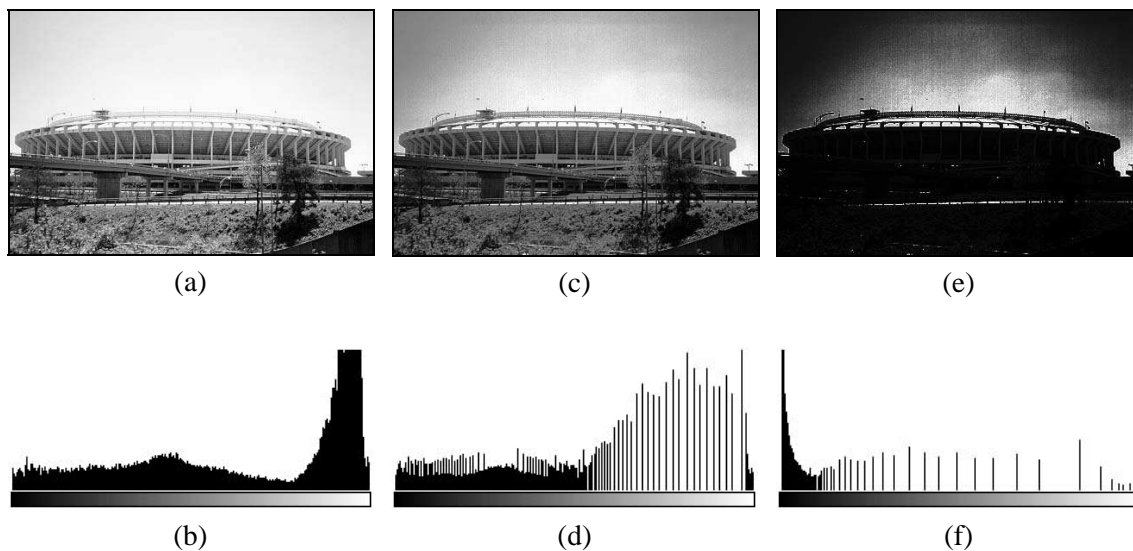


Figura 14 - Comparação entre equalização e hiperbolização de histograma: (a) imagem original, (b) histograma original, (c) imagem após equalização de histograma, (d) histograma equalizado, (e) imagem após hiperbolização, (f) histograma correspondente à imagem (e).

3.5.1 Hiperbolização

Esta técnica, proposta originalmente por Frei [Frei 1977], consiste na modificação da imagem original através de uma função de transferência tal que produza à saída uma imagem cujo histograma tenha forma hiperbólica. Ela é empregada principalmente para corrigir a equalização do histograma levando-se em conta a resposta visual do olho humano, que é considerada logarítmica. Ou seja, a equalização do histograma seria processada em duas etapas, sendo a primeira a aplicação da técnica de hiperbolização e a segunda executada pela retina. A figura 14 mostra um exemplo de aplicação desta técnica, comparando-a com a equalização de histograma.

3.5.2 Hiperbolização quadrática

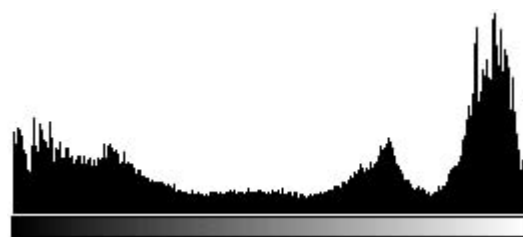
Em [Cobra et al. 1992], Cobra, Costa e Menezes propõem uma nova abordagem à hiperbolização de histograma, à qual denominaram hiperbolização quadrática de histograma.

Esta técnica é baseada em um modelo do sistema visual periférico humano que leva em conta o fato de que o olho humano se acomoda à intensidade média da cena observada e não à intensidade dos pixels individuais, como subentende o modelo utilizado por [Frei 1977]. Como resultado, obtém-se uma distribuição mais espaçada dos níveis de cinza, com menor concentração na região escura do histograma.

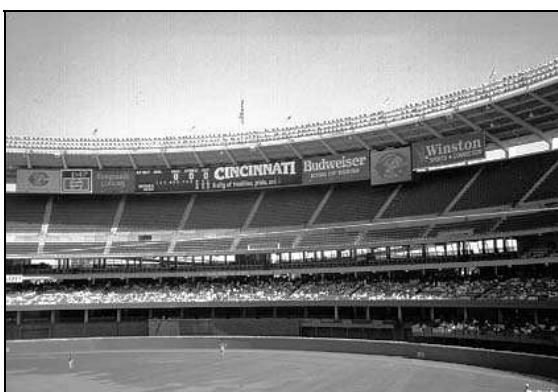
A figura 15 ilustra o uso desta técnica, mostrando na parte (a) a imagem original, na parte (b) seu histograma, e nas figuras 15(g) e 15(h) o resultado da hiperbolização quadrática sobre a imagem e o histograma correspondente. Para efeito comparativo, também são apresentados os resultados da equalização de histograma – figuras 15(c) e 15(d) – e da hiperbolização – figuras 15(e) e 15(f).



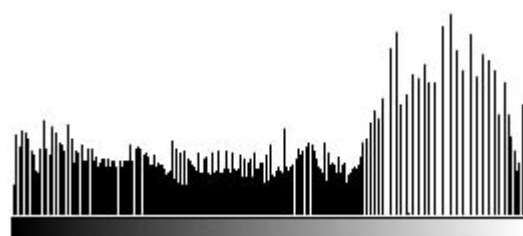
(a)



(b)



(c)

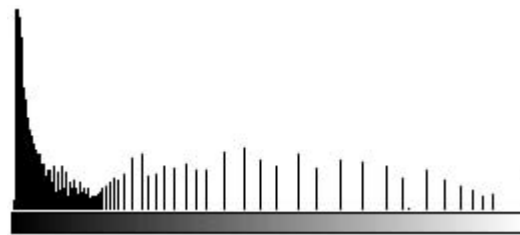


(d)

Figura 15 - Exemplo de hiperbolização quadrática de histograma.



(e)



(f)



(g)

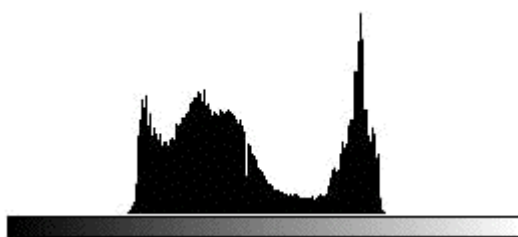


(h)

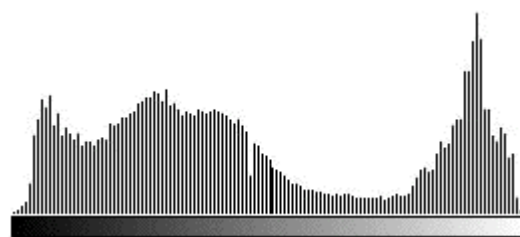
Figura 15 - Continuação.

3.5.3 Expansão de histograma (*Input cropping*)

Nesta técnica, o histograma original de uma imagem é modificado de tal forma que parte dele é expandida para ocupar toda a faixa de cinza da imagem. A figura 16 ilustra esquematicamente o processo e a figura 17 mostra um exemplo de utilização desta técnica para aprimoramento de contraste de uma imagem.



(a)



(b)

Figura 16 - Expansão de histograma.

3.5.4 Compressão de histograma (*Output cropping*)

A técnica de compressão de histograma, como o próprio nome sugere, modifica o histograma original de uma imagem de tal forma que suas raías passam a ocupar apenas um trecho da faixa total de cinza, produzindo como resultado uma redução de contraste na imagem. A figura 18 ilustra esquematicamente o processo enquanto a figura 19 mostra um exemplo de aplicação desta técnica a imagens monocromáticas.



Figura 17 - Exemplo de aplicação da técnica de expansão de histograma.

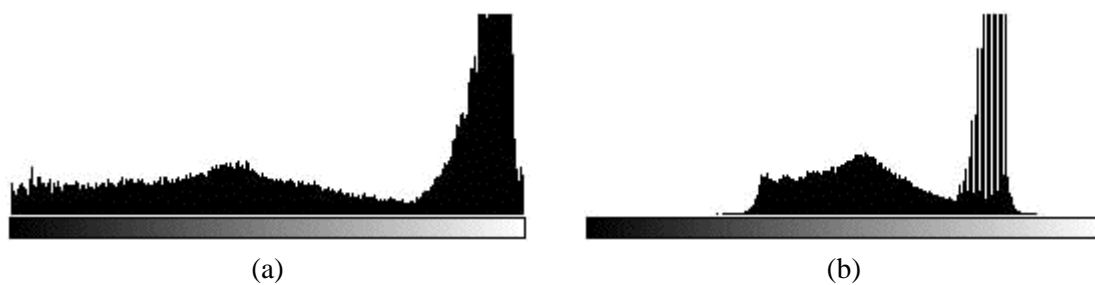


Figura 18 - Compressão de histograma.

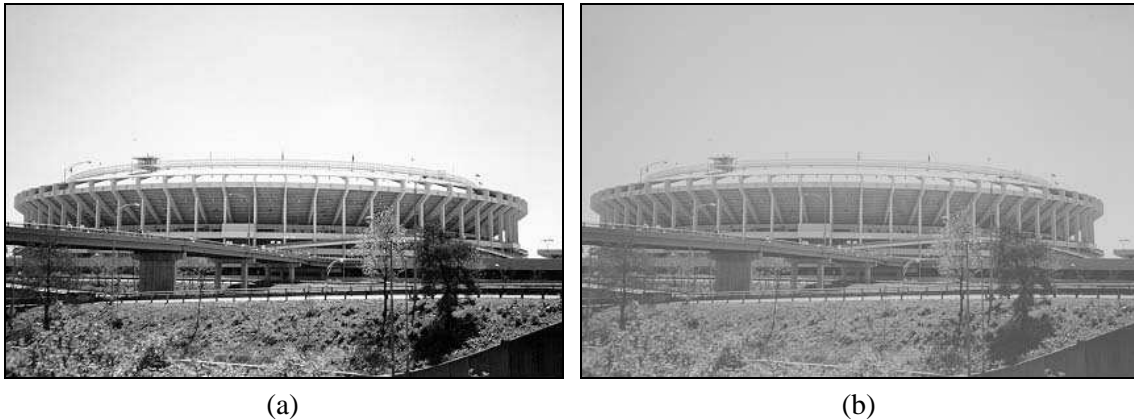


Figura 19 - Exemplo de aplicação da técnica de compressão de histograma.

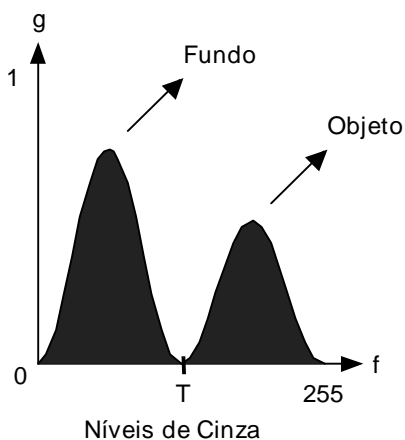
Leitura complementar

O capítulo 10 (seções 10.1 e 10.2) de [Pratt 1991] apresenta exemplos de diversas técnicas de modificação de histograma.

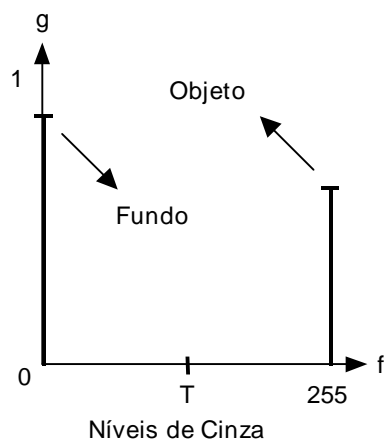
3.6 Limiarização (*Thresholding*)

O princípio da limiarização consiste em separar as regiões de uma imagem quando esta apresenta duas classes (o fundo e o objeto). Devido ao fato da limiarização produzir uma imagem binária à saída, o processo também é denominado, muitas vezes, binarização. A forma mais simples de limiarização consiste na bipartição do histograma, convertendo os pixels cujo tom de cinza é maior ou igual a um certo valor de limiar (T) em brancos e os demais em pretos, como ilustra a figura 20. No caso de níveis de cinza divididos basicamente em duas classes,

onde o histograma apresenta dois picos e um vale, a limiarização é trivial. Ainda assim, os efeitos decorrentes da escolha de um valor específico de limiar dentre os diversos pontos situados na região de vale podem ser analisados na figura 21.



(a)



(b)

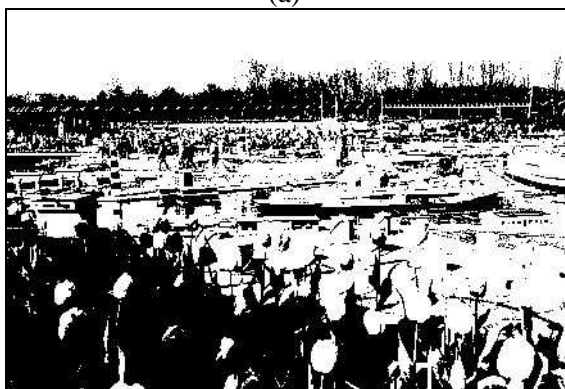
Figura 20 - Limiarização de uma imagem monocromática utilizando limiar T : (a) histograma original, (b) histograma da imagem binarizada.



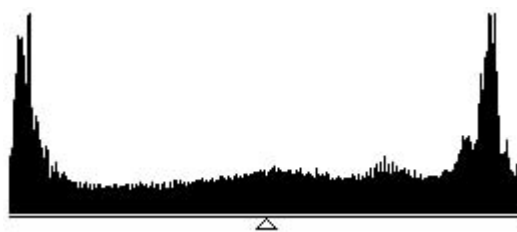
(a)



(b)

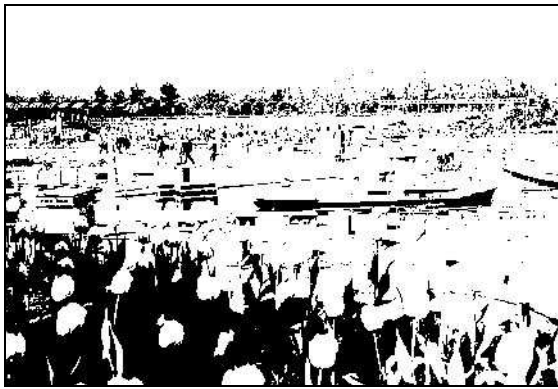


(c)

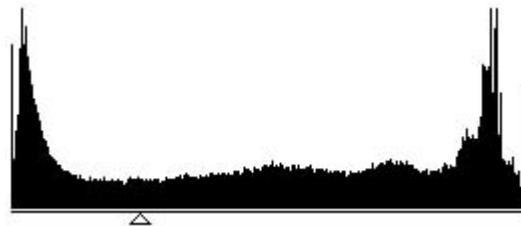


(d)

Figura 21 - Efeitos da escolha do valor de limiar na binarização de uma imagem de 256 tons de cinza. As imagens (c), (e) e (g) correspondem à bipartição dos histogramas (d), (f) e (h), respectivamente, nos limiares indicados, a saber: 128, 64 e 192.



(e)



(f)



(g)



(h)

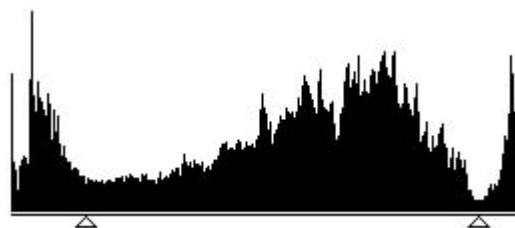
Figura 21 - Continuação.

Matematicamente, a operação de limiarização pode ser descrita como uma técnica de processamento de imagens na qual uma imagem de entrada $f(x,y)$ de N níveis de cinza produz à saída uma imagem $g(x,y)$, chamada de imagem limiarizada, cujo número de níveis de cinza é menor que N . Normalmente, $g(x,y)$ apresenta 2 níveis de cinza, sendo:

$$\begin{aligned} g(x,y) &= 1 \text{ se } f(x,y) \geq T \\ &= 0 \text{ se } f(x,y) < T \end{aligned} \quad (3.11)$$

onde os pixels rotulados com 1 correspondem aos objetos e os pixels etiquetados com 0 correspondem ao fundo (*background*) e T é um valor de tom de cinza pré-definido, ao qual denominamos limiar.

A figura 22(a) mostra um exemplo de histograma particionado utilizando dois valores de limiar: $T_1 = 37$ e $T_2 = 233$. As figuras 22(b) e 22(c) mostram a imagem original e a imagem após a limiarização.



(a)

Figura 22 - Exemplo de utilização de múltiplos limiares.



Figura 22 - Continuação.

A limiarização pode ser vista como uma operação que envolve um teste com relação a uma função T do tipo $T = T[x, y, p(x, y), f(x, y)]$, onde $f(x, y)$ é o tom de cinza original no ponto (x, y) e $p(x, y)$ indica alguma propriedade local deste ponto, por exemplo a média de seus vizinhos. Quando T depende apenas de $f(x, y)$, o limiar é chamado global; quando T depende de $f(x, y)$ e de $p(x, y)$, o limiar é chamado local. Se, além disso, T depende das coordenadas espaciais de (x, y) , o limiar é chamado dinâmico ou adaptativo.

3.6.1 Influência da iluminação

A iluminação desempenha um papel significativo no processo de limiarização, uma vez que provoca alterações no histograma original da imagem, eventualmente eliminando uma região de vale entre dois picos, naturalmente propícia para a definição de um limiar global.

Pode-se provar [Papoulis 1965] que, sendo $f(x, y) = i(x, y) \cdot r(x, y)$ e sendo $z(x, y) = \ln f(x, y) = \ln i(x, y) + \ln r(x, y) = i'(x, y) + r'(x, y)$, onde $i'(x, y)$ e $r'(x, y)$ são variáveis aleatórias independentes, o histograma de $z(x, y)$ é dado pela convolução do histograma de $i'(x, y)$ com o de $r'(x, y)$.

Uma técnica comum utilizada para compensar a não uniformidade da iluminação consiste em projetar o padrão de iluminação em uma superfície refletora branca. Isto nos dá uma imagem $g(x, y) = K \cdot i(x, y)$, onde K depende da superfície utilizada. Deste modo, para qualquer imagem $f(x, y) = i(x, y) \cdot r(x, y)$ obtida com a mesma função iluminação, simplesmente divide-se $f(x, y)$ por $g(x, y)$, obtendo-se uma função normalizada:

$$h(x, y) = \frac{f(x, y)}{g(x, y)} = \frac{r(x, y)}{K} \quad (3.12)$$

Logo, se $r(x, y)$ pode ser limiarizada utilizando o limiar T , então $h(x, y)$ poderá ser segmentada usando um limiar T/K .

A figura 23 ilustra as alterações causadas por modificações no padrão de iluminação na imagem binarizada resultante. Na parte (a) é apresentada a imagem original, cujo histograma é exibido na figura 23(e). O resultado da limiarização desta imagem com limiar $T = 128$ é mostrado na figura 23(c). Na coluna da direita são mostradas a imagem com padrão de iluminação alterado (b), seu histograma (f) e o resultado da limiarização com o mesmo limiar utilizado anteriormente (d).

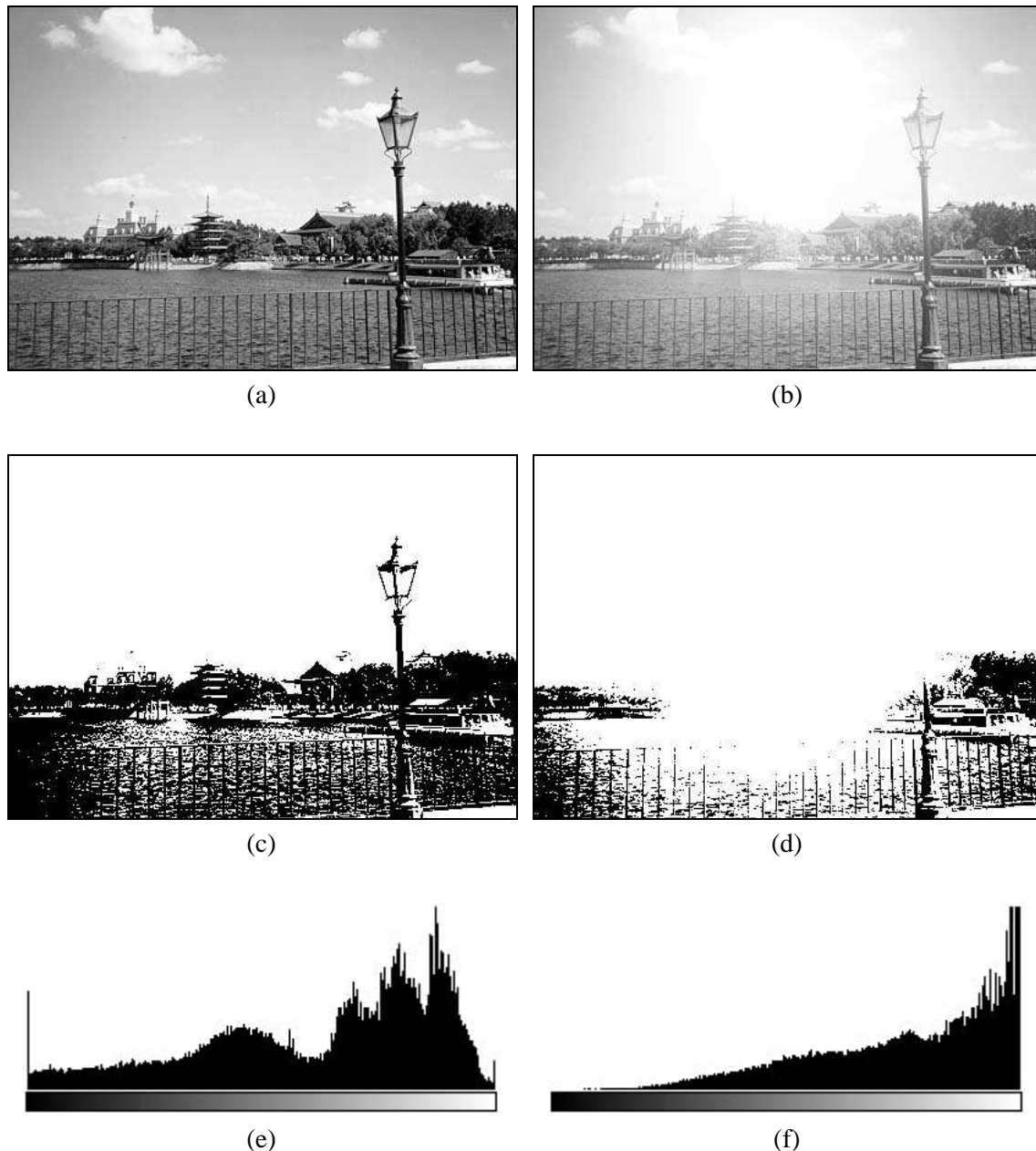


Figura 23 - Influência da iluminação no processo de limiarização.

3.6.2 Limiarização pelas propriedades estatísticas da imagem

Pelo exposto até aqui, assumiu-se que a escolha do valor de limiar é arbitrária e subjetiva. Sabendo que o histograma é uma representação gráfica da distribuição de probabilidade de ocorrência dos níveis de cinza em uma imagem, é lícito imaginar a possibilidade de uso de técnicas de cálculo do valor ótimo de limiar com base nas propriedades estatísticas da imagem.

Uma destas técnicas, denominada limiarização ótima, parte de uma imagem da qual se conhecem as principais propriedades estatísticas (supondo que sua distribuição de probabilidade é normal ou gaussiana), a saber:

μ_1 : média dos tons de cinza da região de interesse

μ_2 : média dos tons de cinza da região de fundo (background)

σ_1, σ_2 : desvios padrão

P_1, P_2 : probabilidade de ocorrência dos pixels pertencentes a esta ou aquela região.

Pode-se mostrar [Gonzalez e Woods 1992] que existe um valor ótimo de limiar, T , dado por uma das raízes da equação

$$AT^2 + BT + C = 0, \quad (3.13)$$

onde:

$$\begin{aligned} A &= \sigma_1^2 - \sigma_2^2 \\ B &= 2(\mu_1\sigma_2^2 - \mu_2\sigma_1^2) \\ C &= \mu_2^2\sigma_1^2 - \mu_1^2\sigma_2^2 + 2\sigma_1^2\sigma_2^2 \ln\left(\frac{\sigma_2 P_1}{\sigma_1 P_2}\right) \end{aligned} \quad (3.14)$$

Duas raízes reais e positivas indicam que a imagem pode requerer dois valores de limiar para obter uma solução ótima.

Se as variâncias forem iguais ($\sigma^2 = \sigma_1^2 = \sigma_2^2$), um único valor T é necessário:

$$T = \frac{\mu_1 + \mu_2}{2} + \frac{\sigma^2}{\mu_1 - \mu_2} \ln\left(\frac{P_2}{P_1}\right) \quad (3.15)$$

Se, além disso, as duas classes forem equiprováveis:

$$T = \frac{\mu_1 + \mu_2}{2} \quad (3.16)$$

o que está em acordo com o conceito intuitivo de que o valor ótimo de limiar quando as classes apresentam a mesma distribuição de probabilidade (os lóbulos são exatamente iguais) é o ponto médio entre as médias das classes.

Leitura complementar

O livro de Castleman [Castleman 1995] apresenta em detalhes outras alternativas de cálculo do valor ótimo de limiar.

Existem vários trabalhos científicos que abordam diferentes alternativas para a obtenção de uma melhor limiarização de uma imagem levando em conta seus parâmetros estatísticos. Estas técnicas pressupõem a determinação automática do melhor valor de limiar, ou seja, partem da premissa de que não haverá um operador humano que determine por tentativa e erro qual o valor de limiar mais adequado. Uma destas propostas, baseada na minimização da variância intra-grupo, encontra-se em [Otsu 1979] e está resumida em [Passariello e Mora 1995].

Em [Haralick e Shapiro 1992] encontra-se um resumo do método proposto por Kittler & Illingworth, o qual se baseia em assumir que o histograma é formado pela mistura de duas distribuições gaussianas, cujas médias e variâncias são conhecidas, no qual o objetivo é minimizar a chamada 'distância de informação de Kullback' [Passariello e Mora 1995]. De acordo com os resultados reportados em [Haralick e Shapiro 1992] com respeito a uma comparação entre o método de Otsu [Otsu 1979] e o método de Kittler-Illingworth, este último é o que produz melhores resultados.

Várias tentativas de estabelecimento de um valor adequado de limiar global (utilizando diversas técnicas de pré-processamento da imagem), sob os conceitos de precisão (*accuracy*) e reprodutibilidade, são mostradas em [Russ 1995].

Gómez-Allende [Gómez-Allende 1993] propõe um algoritmo original de limiarização baseado na busca de mínimos do histograma, no qual o histograma é submetido a uma filtragem

passa-baixas para reduzir as irregularidades causadas por objetos pouco relevantes e/ou ruído e, portanto, facilitar a detecção dos mínimos do histograma.

White e Rohrer [White e Rohrer 1983] descrevem um algoritmo de limiarização dinâmica implementado em hardware como parte de um sistema de Reconhecimento Óptico de Caracteres (OCR).

Mardia e Hainsworth [Mardia e Hainsworth 1988] propõem e comparam diversos algoritmos de limiarização espacial.

O capítulo 2 de [Haralick e Shapiro 1992], a Seção 7.3 de [Gonzalez e Woods 1992] e a Seção 5.1 de [Sonka et al. 1993] são dedicados ao problema da limiarização.

Exercícios Propostos

1. Que efeito uma transformação de intensidade $s = r^\gamma$ provocará em uma imagem monocromática com valores de níveis de cinza originais (r) normalizados na faixa de 0 a 1, caso $\gamma > 1$?

2. Assinalar V ou F conforme as proposições a seguir sejam verdadeiras ou falsas.

() A técnica de equalização de histograma aplicada a imagens digitais nunca produz à saída um histograma perfeitamente plano.

() Após um histograma ter sido equalizado, uma nova aplicação da técnica de equalização de histograma sobre a imagem não produzirá nela nenhuma alteração.

() A técnica de hiperbolização de histograma tem como principal vantagem sobre a equalização de histograma o fato de que a primeira leva em consideração as características não-lineares da curva de intensidade luminosa subjetiva versus intensidade luminosa física da imagem do olho humano.

() Ao recortar uma imagem em dois pontos quaisquer, o histograma da subcena resultante sempre será idêntico ao da imagem original, porque as raias verticais indicam a concentração de pixels em termos percentuais e, portanto, não dependem do número total de pixels da imagem.

3. Considere a imagem a seguir, representada por uma matriz 7 x 7, onde cada elemento da matriz corresponde ao nível de cinza normalizado do pixel correspondente, sendo 0 = preto, 1 = branco.

0	3/7	2/7	2/7	1/7	1/7	4/7
3/7	2/7	1/7	1/7	1/7	1/7	4/7
2/7	0	1	1/7	3/7	0	0
0	5/7	1/7	0	6/7	0	1/7
1/7	1/7	1/7	3/7	6/7	6/7	5/7
1/7	1/7	1/7	1/7	5/7	6/7	4/7
0	1	0	0	0	0	4/7

Pede-se:

- Calcular as probabilidades de cada nível de cinza e plotar seu histograma.
- Na imagem original predominam pixels claros ou escuros?