

**Aula 1**

O processamento digital de imagens teve sua origem nas indústrias de jornais, transportando imagens usando cabo submarino de Londres a Nova York. Uma imagem pode ser definida como uma função bidimensional  $f(x,y)$ , onde  $x$  e  $y$  são as coordenadas dos pixels e a amplitude de  $f$  é chamada intensidade da imagem no local. Quando esses valores são finitos e discretos temos uma imagem digital. Podem ser aplicados vários processos na imagem para se tornar melhor o seu processamento ou a parte visual para os olhos humanos.

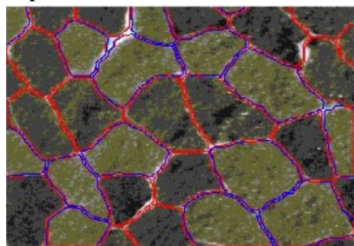
**Realce de contraste:** Primeiro é feito um histograma da imagem (quantidade de pixels em cada amplitude), assim sendo possível encontrar o pixel com menor tonalidade e leva-lo a 0 e o pixel de maior tonalidade leva-lo a 255, assim deixando a imagem mais suave.

**Correção de iluminação irregular:** A ideia é deixar a imagem toda com uma intensidade regular, isso é feito através do borramento, que consiste em pegar os vizinhos de um pixel, fazer a média e substituir o pixel que está sendo calculado. Após conseguir uma imagem borrada é aplicado um outro processo que consiste em pegar a imagem original e dividir cada um de seus pixels pelo pixel na posição relativa da imagem borrada. Assim se obtém uma imagem com iluminação regular.

**Redução de ruído aleatório:** Para suavizar os ruídos podemos aplicar vários tipos de filtros, mas o mais básico é o filtro da média, que acaba borrando a imagem.

Vários outros tipos de processamentos podem ser feitos nas imagens como redução de ruído periódico, redução de borramento devido ao movimento, correção de imagem desfocada, correção de foco em microscopia óptica, pseudo-cores, entre outros.

**Analisar imagens:** Isto é retirar informações quantitativas da imagem possibilitando fazer medidas automáticas ou semi-automáticas, realizar medidas impossíveis ou muito difíceis de se obter manualmente, obter medidas muito



mais rápido e realizar medidas mais precisas e acuradas. Por exemplo, podemos obter diagnósticos médicos observando as características de fibras musculares, onde as fibras rápidas são caracterizadas em vermelho e as lentas em azul. Uma distribuição não uniforme dessas fibras pode indicar atrofia neuropática.

**Aula 2**

O olho humano é capaz de ver uma pequena faixa de cores, que é chamado de espectro visível, assim não é necessário guardar tons de cores que é imperceptível pro olho humano, facilitando o processamento das imagens. Vale ressaltar que o olho não é capaz de distinguir tons de cores muito próximos, então podemos tirar vantagem desse fato para melhorar o processamento.

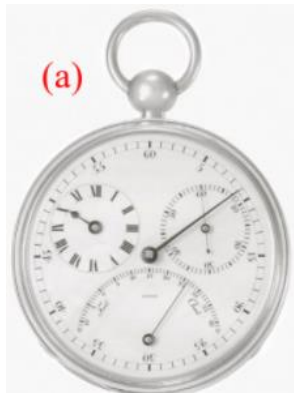
**Modelo de formação da imagem usando iluminação:** Função 2D de  $f(x,y)$ , onde  $0 < f(x,y) < \infty$ .  $F(x,y)$  é positiva escalar obtida por:  $f(x,y) = i(x,y)r(x,y)$ , com  $i(x,y)$  a intensidade da luz  $0 < i(x,y) < \infty$  e  $r(x,y)$  a reflectância  $0 < r(x,y) < 1$ .

**Representação de imagens digitais monocromáticas:** a imagem pode ser representada computacionalmente por uma matriz com  $M$  linhas e  $N$  colunas, onde cada elemento da matriz é chamado de pixel.

Equipamentos de captura de luz são normalmente matriciais e são capaz de transformar energia da luz em sinais elétricos, mas essa transformação é dependente de dois fatores: a amostragem, que é referente ao espaço; e a quantização, que é referente a intensidade.

**Conversão do sinal analógico para o digital:** O sinal analógico é formado por uma senóide para transforma-lo em sinal digital teríamos que obter os infinitos pontos da senóide, que é algo impossível de se representar computacionalmente. Para isso fazemos a amostragem da senóide pegando pontos com uma certa distância, quanto menor a distância maior a quantidade de pontos, logo maior a precisão da imagem digital. Com a mesma ideia fazemos a quantização, onde quanto menor a distância entre os valores possíveis de representar maior a nossa precisão. A imagem é representada com uma malha 2D, onde é capturado os valores ao longo do tempo, se multiplicarmos o número de linhas pelo número de colunas obtemos a resolução da imagem.

O tamanho dos detalhes que podem ser digitalizados depende do tamanho da grade, o que decorre do teorema da amostragem, que afirma que para que um sinal possa ser adequadamente reconstruído, a taxa de amostragem precisa ser maior que duas vezes a maior frequência presente no sinal. Esta taxa de amostragem mínima é chamada frequência de Nyquist, quando essa taxa amostragem é menor que a frequência de Nyquist ocorre um efeito chamado de alia-



sing ou foldover. Quanto maior a taxa de amostragem melhor a qualidade, mas maior o tamanho. Por exemplo com 1250 dpi (pixels por polegadas) a imagem fica ótimas, mas muito grande.

Uma imagem pode ser representada em um plano 3D, onde  $x$  e  $y$  é a posição do pixel e  $z$  é a intensidade, podendo variar de 0 a 255, com 0 sendo preto e 255 sendo branco.

Em uma imagem pode ocorrer saturação e ruído, onde saturação é altos valores de brilho, aproximados para o valor máximo. E ruídos são valores indesejados muito diferente dos demais, aparecendo como

uma granulação na textura.

Na quantização consideramos que o número de bins é  $2^k$ , assim o número de bits necessário para representar a imagem é  $M*N*k$ , por exemplo, é necessário 1MB para armazenar uma imagem 1024x1024, utilizando 256 tons de cinza ( $M=N=1024$ ,  $k=8$ ). Já uma imagem colorida é 3x esse valor.

**Interpolação de imagens:** Muito usada em tarefas como: ampliação (zooming) e encolhimento (shrinking), rotação e correções geométricas. Existe alguns métodos para fazer-la.

Vizinho mais próximo: Olha o pixel mais próximo na grade da imagem original e atribui sua intensidade para o novo pixel.

Interpolação bilinear: O valor do novo pixel é calculado a partir da média dos quatro vizinhos mais próximos. Essa técnica obtém um resultado melhor.

Interpolação bicúbica: O valor da intensidade atribuído ao ponto  $(x,y)$  é obtido usando:  $\sum_{i=0}^3 \sum_{j=0}^3 a_{ij} x^i y^j$ . Sendo está a técnica que obtém o melhor resultado.

Por exemplo: a) imagem reduzida a 72 dpi e expandida de volta a 3692x 2812 usando interpolação vizinho mais próximo.

(b) Imagem encolhida e expandida usando interpolação bilinear.

(c) Como em (b) mas usando interpolação bicúbica.



**Relação entre pixels:** Um pixel  $p$  tem quatro vizinhos com coordenadas  $(x+1,y)$ ,  $(x-1,y)$ ,  $(x,y+1)$  e  $(x,y-1)$ , sendo chamado de vizinhança-4 de  $p$ ,  $N_4(p)$ . Os vizinhos da diagonais de  $p$  tem coordenadas  $(x+1,y+1)$ ,  $(x+1,y-1)$ ,  $(x-1,y+1)$  e  $(x-1,y-1)$ ,  $N_D(p)$ . A união de  $N_4(p)$  e  $N_D(p)$  é chamado de vizinhança-8,  $N_8(p)$ .

**Conectividade:** Pixels são conectados se são vizinhos e se os níveis de cinza satisfazem uma certa similaridade, por exemplo  $V=\{33,...,46\}$ .

Conectividade-4: dois pixels com valores em  $V$  estão no conjunto  $N_4(p)$ .

Conectividade-8: dois pixels com valores em  $V$  estão no conjunto  $N_8(p)$ .

Conectividade-m: dois pixels  $p$  e  $q$  com valores em  $V$ , onde  $q$  está no conjunto  $N_4(p)$  ou  $q$  está no conjunto  $N_D(p)$  e o conjunto  $N_4(p) \cap N_4(q)$  é vazio.

**Caminho digital:** de um pixel  $p$  com coordenadas  $(x,y)$  a um pixel  $q$  com coordenadas  $(s,t)$  é uma sequência de pixels distintos com coordenadas  $(x_0,y_0)$ ,  $(x_1,y_1)$ , ...,  $(x_n,y_n)$ , onde  $(x_0,y_0) = (x,y)$  e  $(x_n,y_n) = (s,t)$  e pixels  $(x_i,y_i)$  e  $(x_{i-1},y_{i-1})$  são adjacentes para  $1 \leq i \leq n$ . Neste caso,  $n$  é o comprimento do caminho. O grau do caminho depende do tipo de adjacência.

**Medidas de distância:** Para pixels  $p$ ,  $q$  e  $z$  com coordenadas  $(x,y)$ ,  $(s,t)$  e  $(v,w)$ , respectivamente,  $D$  é uma função distância ou métrica se:

– (a)  $D(p,q) \geq 0$  ( $D(p,q) = 0$  iff  $p = q$ ),

– (b)  $D(p,q) = D(q,p)$ , e

– (c)  $D(p,z) \leq D(p,q) + D(q,z)$ .

A distância euclidiana entre  $p$  e  $q$  é definida por  $D_e(p,q) = [(x-s)^2 + (y-t)^2]^{1/2}$

Distância  $D_4$  (city-block distance):  $D_4(p,q) = |x-s| + |y-t|$ .

Distância  $D_8$  (chessboard distance):  $D_8(p,q) = \max(|x-s|, |y-t|)$ .

**Operações aritméticas:** Adição:  $p+q$ , fazer a média da imagem e reduzir o ruído, por exemplo, reduzir ruídos de uma imagem;

Subtração:  $p-q$ , remover fundo, ou detectar objetos que estão em uma imagem e não estão na outra, por exemplo, melhoramento das diferenças em imagens;

**Multiplicação:** corrigir sombras e níveis de cinzas irregulares, por exemplo,  $f(x,y)$  é a imagem perfeita,  $g(x,y)$  é a imagem sombreada e  $h(x,y)$  é o padrão de sombreado, é possível obter a imagem  $f(x,y)$  multiplicando  $g(x,y)$  pelo inverso de  $h(x,y)$ ;

**Divisão:** corrigir sombras e níveis de cinzas irregulares.

**Operações geométricas:** sendo expressa por  $(x,y) = T\{(v,w)\}$ , onde T pode ser translação, rotação, aumento ou diminuição da escala. Tomar cuidado pois a contração e expansão causa um warping.

**Ajuste de cor e brilho:** Pode-se sobrepor uma imagem na outra, sendo possível observar traços de ambas, isso é chamado de dissolve. Ajustando a imagem para se encaixarem utilizamos as operações espaciais e radiométricas juntas, obtendo um Murphing.

Podemos operar os pixels individualmente, por exemplo para obter a inversa da imagem, mas também podemos operá-los em vizinhança, por exemplo na operação de média.

Podemos achar a probabilidade de cada intensidade ocorrer em uma imagem, com isso podemos obter sua média, variância e desvio padrão.

### Aula 3

Uma cor é definida pelo comprimento da sua onda. As cores podem ser descritas como combinações das cores primárias vermelho (R), verde (G) e azul (B) e também através das cores secundárias magenta (M), ciano (C) e amarelo (Y).

As imagens coloridas são representadas por três canais (R,G,B), onde cada pixel é um vetor  $3 \times 1$ . Cada pixel pode ter uma quantidade de bits dividida para os canais. Quando se há poucas cores a imagem não tem uma transição suave de para cada cor, assim podemos usar o Dithering para criar mais cores.

As cores são diferenciadas por brilho, matiz e saturação. Matiz e saturação juntas compõem a cromaticidade, e então, uma cor pode ser caracterizada por seu brilho e cromaticidade.

Há vários modelos de cores. Onde cada um é mais adequado dependendo da aplicação, por exemplo, monitores coloridos usam R,G,B e impressoras coloridas usam CMY. O modelo RGB tem 16777216 tons de cores, quando  $R=G=B$  obtém-se os tons de cinza.

Podemos usar a representação CIE-RGB, sistema aditivo, para representar as cores, sendo sua intensidade em cinza calculada por  $(R+G+B)/3$  ou  $0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B$ . O modelo CMY é definido por  $(1,1,1)-(R,G,B)$ . O modelo YIQ é usado na transmissão de televisão onde Y é a intensidade, I e Q é a cromaticidade. Modelo YUV é parecido com o YIQ mudando o modo de transformar de RGB para tal.

Em questão de precisão, o modelo mais fiel seria o HSI-HSV-HSL, pois ele desacopla a informação de brilho da informação de cores (matiz e saturação). Quanto mais saturada menos cinza na cor, a luminosidade é independente da saturação e a cor é a cor base.

### Aula 5

**Realce de imagem:** É o processamento que melhora uma imagem de acordo com a finalidade específica, análise visual melhora visualmente, análise computacional melhora as feições a serem extraídas. Os pixels podem ser processados pontualmente ou considerando sua vizinhança. Uma operação pontual é definida por  $G(x,y) = T[f(x,y)]$ , onde T é a transformação, por exemplo a binarização, que transforma a imagem em duas cores, preta e branco.

**Imagem negativas:** consiste em inverter os valores de nível de cinza. Podem ser aplicadas em exames médicos para enxergar melhor detalhes.

**Compressão da escala dinâmica:** Mapeia os valores para uma faixa mais adequada usando uma função logarítmica  $c \cdot \log(1+r)$ .

**Funções potências:**  $S = cr^y$ , onde c e y são constantes positivas.

**Fatiamento de tons de cinza:** Uma parte dos tons é transformada, por exemplo os tons de uma imagem entre 100 e 150 são maximizados recebendo 255.

As imagens podem ser fatiadas em camadas de cada bit, com a imagem sendo a soma das mesmas.

**Equalização de histogramas:** A equalização serve para tornar a imagem uniforme, a equalização é dada por  $\max(0, ((N^{\circ} \text{ de Tons de Cinza Total} \cdot \text{Freq. acumulada}) / (N^{\circ} \text{ de Colunas} \cdot N^{\circ} \text{ de Linhas})) - 1]$ .

**Especificação de histogramas:** Oferece a possibilidade de se processar a imagem no sentido de atender um histograma previamente estabelecido, somando a imagem original com a equalizada.

**Equalização de histogramas local:** Aplica a equalização considerando apenas uma vizinhança ao redor do pixel.

**Subtração de imagens:** Subtraindo pixel a pixel de duas imagens obtemos o realce de suas diferenças.

**Média de imagens:** Serve para minimizar ruídos, somando imagens ruidosas e dividindo pela quantidade de imagens.

**Máscara de convolução:** um filtro  $g(a,b)$  é deslocado sobre a imagem  $f(m,n)$  e obtendo o resultado  $h(x,y)$ .

**Filtro passa-baixa:** suaviza imagens, atenuando ruídos, passando uma matriz usando vizinhança-4 ou 8. Essa técnica elimina altas frequências, que são correspondentes as bordas.

**Filtro da mediana:** é uma técnica melhor que o filtro da média, onde pega uma vizinhança, a ordena e pega o elemento central.

Um dos propósitos de borrar uma imagem é obter uma representação aproximada dos objetos que a compõem. Assim, pixels acima de um certo valor de tom de cinza (objetos) podem ser facilmente separados de pixels abaixo deste valor (fundo). Assim ficando mais fácil de processá-la. Desse modo podemos utilizar a binarização ou limiarização que consiste em atribuir 0 a valores menores que x e manter seu valor para maiores ou iguais. A binarização pode separar os objetos do fundo, pode ocorrer que os objetos fiquem com buracos, assim preenchendo os pixels brancos que estão rodeados de pretos por preto, separa os objetos removendo pixels pretos com vários vizinhos brancos e faz uma varredura pixel por pixel até encontrar um preto, o marca e faz o processo de inundação para ignorar os pixels que pertencem a ele.

**Filtro de auto reforço:** Passa-alta é obtido através da imagem original – o passa-baixa, para reforçar os detalhes da imagem fazemos  $A_{original} - pass\_baixa$ , assim quando o A cresce os detalhes ficam mais claros.

**Deteção de bordas:** As bordas são usadas pela visão humana para separar os objetos e reconhece-los. As bordas são regiões nas imagens que acontecem uma mudança de intensidade em um intervalo de espaço e em uma direção. Contornos (boundary) é uma linha fechada formado pelas bordas (edges) do objeto, nem sempre sendo possível obtê-lo através das bordas.

Aguçamento (sharpening): Realça os detalhes finos das imagens. Enquanto os filtros de suavização utilizam uma operação equivalente a uma integral, os filtros de aguçamento utilizam uma operação equivalente a uma derivada. A primeira derivada será  $f(x+1,y) - f(x,y)$  e a segunda será  $f(x+1,y) - 2f(x,y) + f(x-1,y)$ .

-1	0	0	-1
0	1	1	0

Usando a primeira derivada Roberts desenvolveu um filtro capaz de detectar bordas através da convolução. Onde se usa duas matrizes  $2 \times 2$  e as somas, obtendo as bordas.

0	1	0	1	1	1
1	-4	1	1	-8	1
0	1	0	1	1	1

Laplaciano também desenvolveu um método para obter as bordas utilizando matrizes  $3 \times 3$ , onde se pode escolher usar vizinhança-4 ou 8, com esse método também é possível realçar as bordas aumentando um no

centro, assim a imagem original será somada as bordas.

Uma outra técnica desenvolvida por Sobel é utilizar os gradientes das imagens onde pode se obter até a direção das bordas através da fórmula  $\tan^{-1}(\text{derivada parcial de } y / \text{derivada parcial de } x)$ . A intensidade da borda é dada pela magnitude do gradiente  $\sqrt{(\text{derivada parcial de } x)^2 + (\text{derivada parcial de } y)^2}$ . Os gradientes podem ser obtidos pelas convoluções:

As

$$\frac{\partial f}{\partial x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad \frac{\partial f}{\partial y} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

As detecções de bordas podem gerar ruídos, para minimizá-los podemos utilizar a técnica de limiarização.

### Aula 6

O processamento de imagens no domínio da frequência pode remover ruídos periódicos após a redução de um sinal com uma frequência bem definida.

Uma função  $f(x)$  é chamada periódica se ela é definida para todo real x e se existe um número positivo p tal que:  $f(x+p) = f(x)$ . O número p é chamado período da função  $f(x)$ . O gráfico da função é obtido por repetições periódicas do gráfico em um intervalo de tamanho p.

**Série de Fourier:** Uma série de funções simples como cosseno e seno pode ser escrita como, o difícil é encontrar os coeficientes  $a_n$  e  $b_n$ .

$$a_0 + \sum_{n=1}^{\infty} a_n \cdot \cos nx + b_n \cdot \sin nx$$

$A_0$ ,  $a_n$  e  $b_n$  são os coeficientes de Fourier e são definidos como:



$$a_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) dx$$

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(nx) dx$$

$$b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(nx) dx$$

**Números complexos:**  $z_1 = a+bi$  e  $z_2 = c+di$ , com  $i = \sqrt{-1}$ ,  $a$  e  $c$  são reais e  $b$  e  $d$  são imaginário.

Operações:

Soma:  $a+bi + c+di = (a+c) + (b+d)i$

Subtração:  $a+bi - c+di = (a-c) - (b+d)i$

Multiplicação:  $a+bi * c+di = (ac-bd) + (ad+bc)i$

Divisão:  $a+bi / c+di = (ac+bd)/(cc+dd) + (cb-ad)/(cc+dd)i$

Exponencial:  $e^{(a+bi)} = e^a \cos(b) + e^a \sin(b)i$

Forma polar:  $a = r \cdot \cos X$ ,  $b = r \cdot \sin X$ ,  $z = r(\cos X + i \sin X)$ ,  $X = \arctang(b/a)$ , assim  $e^{iy} = \cos y + i \sin y$


Transformada de Fourier: Obtém o domínio de frequência de uma imagem. Pode ser aplicada quando uma em imagens amostradas em uma matriz quadrada  $M=N$ , através da fórmula:

$$F(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \exp[-j2\pi(ux + vy) / N] \quad \text{para } u, v = 0, 1, \dots, N-1$$

Após transformar a imagem em sua frequência é possível transformá-la de volta para a imagem utilizando a fórmula:

$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} F(u, v) \exp[j2\pi(ux + vy) / N] \quad \text{para } x, y = 0, 1, 2, \dots, N-1$$

Exemplo



$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \exp[-j2\pi ux / N]$$

$u=0 \quad F(0) = \frac{1}{4} \sum_{x=0}^3 f(x) \exp[0]$

$$= \frac{1}{4} [f(0) + f(1) + f(2) + f(3)] = \frac{1}{4} (2 + 3 + 4 + 4) = 3,25$$

$u=1 \quad F(1) = \frac{1}{4} \sum_{x=0}^3 f(x) \exp[-j2\pi x / 4]$

$$= \frac{1}{4} (2e^0 + 3e^{-j\pi/2} + 4e^{-j\pi} + 4e^{-j3\pi/2})$$

Aplicando a Fórmula de Euler

$$= \frac{1}{4} [2(\cos 0 + j \sin 0) + 3(\cos \pi/2 - j \sin \pi/2) + 4(\cos \pi - j \sin \pi) + 4(\cos 3\pi/2 - j \sin 3\pi/2)]$$

$$= \frac{1}{4} [2(1+0j) + 3(0-1j) + 4(-1-0j) + 4(0+1j)]$$

$$= \frac{1}{4} [2 - 3j - 4 + 4j]$$

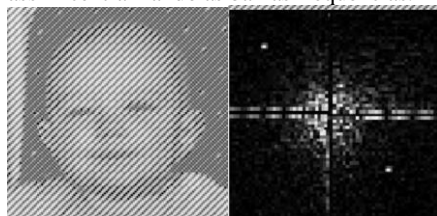
$$= \frac{1}{4} [-2 + j]$$

O espectro de Fourier pode ser calculado a partir da magnitude de cada um dos termos da transformada.

Isto é:  $F(0) = 3.25 + 0j$ ,  $F(1) = -2/4 + 1/4j$ ,  $F(2) = -1/4 + 0j$ ,  $F(3) = -2/4 - 1/4j$ .

A escala do espectro de Fourier normalmente é muito mais alta do que a reproduzida com fidelidade, para consertar essa escala e ficar mais visível, ou seja, aumentar o tamanho do espectro usamos uma técnica que compensa a dificuldade na exibição.

A imagem original do espectro aparece com as baixas frequências nos cantos das imagens, então ela é dividida em quatro e desloca os pedaços na diagonal, assim centralizando as baixas frequências.



**Propriedades da transformada de Fourier:**

- Separabilidade: a função 2D pode ser dividida em somatórias separadas 1D;
- Translação: se somado valores nas coordenadas de  $f(x,y)$  impactará nas coordenadas de  $F(u,v)$  e o contrário também ocorrerá;
- Periodicidade e simetria conjugada: a transformada de Fourier e sua inversa são periódicas com período  $N$ .
- Rotação: Se transformado as coordenadas da função e da transformada em coordenadas polares  $x = r \cdot \cos \theta$ ,  $y = r \cdot \sin \theta$ ,  $u = w \cdot \cos \phi$ ,  $v = w \cdot \sin \phi$

$w = \text{sen} \theta$ , logo  $f(x,y)$  e  $F(u,v)$  ficam  $f(r,\theta)$  e  $F(w,\phi)$ , assim qualquer alteração no ângulo de uma das funções alterará a outra.

- Distributiva e mudança de escala: a transformada e sua inversa são distributivas para a adição, mas não para a multiplicação.
- Valor médio: a média da transformada é relacionada a transformada, onde:  $\frac{1}{N} F(0,0)$ ;

- Laplaciano: Com o operador laplaciano podemos realçar as bordas de uma imagem, a transformada aplicada em laplaciano é definido por:  $-(2\pi)^2(u^2 + v^2)F(u, v)$ ;

- Convolução: A convolução de  $f(x)*g(x)$  é igual ao produto da transformada de ambas, ou seja,  $F(x).G(x)$ , o contrário também pe valido  $f(x).g(x) = F(x)*G(x)$

- Correlação: sendo as funções  $f(x)$  e  $g(x)$  denotadas por  $f(x) \circ g(x)$ , é definida por :

Sendo  $*$  o conjunto dos imaginários.

$$f(x) \circ g(x) = \int_{-\infty}^{\infty} f^*(\alpha) g(x + \alpha) d\alpha$$

**Transformada rápida de Fourier (FFT):** A transformada de Fourier é extremamente custosa, com uma operação de complexidade  $N^2$ , decompondo a transformada de Fourier adequadamente podemos chegar a uma complexidade  $N \log_2 N$ , que é muito menos custoso, essa decomposição é chamada de Transformada rápida de Fourier (FFT). Para obtê-la é preciso representar a transformada de uma maneira diferente:

$$F(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) W_N^{ux} \quad \text{onde } W_N = e^{-\frac{j2\pi}{N}} \text{ e } N = 2^n \text{ ou } N \text{ é par}$$

A função acima então é dividida em duas, uma parte par e uma parte ímpar, e daí é obtida a seguinte equação:

$$f(u + M) = \frac{1}{2} (F_{\text{par}}(u) - F_{\text{impar}}(u) W_{2M}^u)$$

Essa fórmula tem como intuito substituir o processo de convolução, em que o processo é feito da seguinte forma: faz a transformada rápida da imagem e da máscara, separadamente, depois a matriz máscara recebe o mesmo tamanho da imagem, preenchendo os campos vazios com 0, após isso é feito a multiplicação  $F(u)$  por  $G(u)$ , que gera uma matriz e como ela é feito o processo da transformada inversa para gerar uma imagem final.

**Transformada de Walsh:** Se  $N = 2^n$ , a transformada discreta de Walsh de uma função  $F(x)$ , denotado por  $W(x)$ , é obtida pela fórmula:

$$W(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) \prod_{i=0}^{n-1} (-1)^{b_i(x) b_{n-1-i}(u)}$$

Onde  $b_k(z)$  é o  $k$ -ésimo bit na representação binária de  $z$ .  $b_0(z)=0$ ,  $b_1(z)=1$ ,  $b_2(z)=1$ .

$$f(x) = \sum_{u=0}^{N-1} W(u) \prod_{i=0}^{n-1} (-1)^{b_i(x) b_{n-1-i}(u)}$$

E sua inversa é

Estendendo para 2D:

$$W(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \prod_{i=0}^{n-1} (-1)^{[b_i(x) b_{n-1-i}(u) + b_i(y) b_{n-1-i}(v)]}$$

$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} W(u, v) \prod_{i=0}^{n-1} (-1)^{[b_i(x) b_{n-1-i}(u) + b_i(y) b_{n-1-i}(v)]}$$

**Transformada de Hadamard:** A somatória do expoente é executada através da aritmética binária.  $N = 2^n$ . Fórmula para 1D:

$$H(u) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) (-1)^{\sum_{i=0}^{n-1} b_i(x) b_i(u)}$$

2D:

$$H(u, v) = \frac{1}{N} \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) (-1)^{\sum_{i=0}^{N-1} [b_i(x) b_i(u) + b_i(y) b_i(v)]}$$

$$f(x, y) = \frac{1}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} H(u, v) (-1)^{\sum_{i=0}^{N-1} [b_i(x) b_i(u) + b_i(y) b_i(v)]}$$

**Transformada cosseno discreta:** É frequentemente usada para fazer a compressão de imagens. O padrão JPEG usa essa técnica.

Definida por, em 2D:

$$C(u, v) = \alpha(u) \alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \cos \left[ \frac{(2y+1)v\pi}{2N} \right]$$

Para  $u = 0, 1, 2, \dots, N-1$ .

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} \alpha(u) \alpha(v) C(u, v) \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \cos \left[ \frac{(2y+1)v\pi}{2N} \right]$$

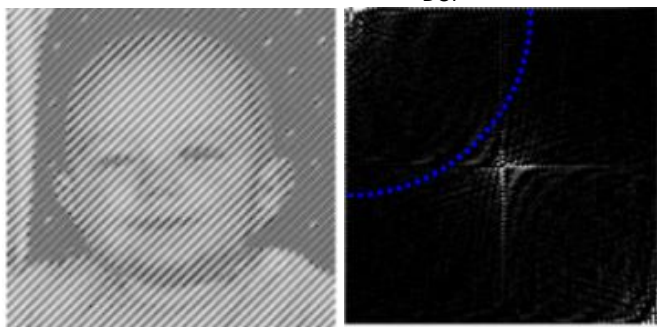
E sua inversa, para  $x = 0, 1, 2, \dots, N-1$ .

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{N}} & \text{para } u = 0 \\ \sqrt{\frac{2}{N}} & \text{para } u = 1, 2, \dots, N-1. \end{cases} \quad \text{Onde, para ambas a é:}$$



**Filtro no domínio da frequência:** Temos o filtro ideal que usa a ideia de binarização, ou seja, 1 se  $D(u, v) \leq D_0$  ou 0 se  $D(u, v) > D_0$ . Já o filtro de butterworth é parecido com a binarização, mas onde tem-se uma suavização dos valores entre 1 e 0, tornando seu gráfico uma curva.

DCT



Passa-Baixa, Corte= 77

Inversa DCT



## Aula 7

Os filtros realizados no domínio da frequência, basicamente, é feito um pré-processamento na imagem, depois é feita a transformada de fourier, é

feito o filtro no espectro, faz a inversa da transformada de fourier para obter a imagem filtrada e se faz um pós-processamento.

O espectro da frequência pode ser representado por um gráfico 3D, onde  $x$  e  $y$  é a posição e o  $z$  é a intensidade. Assim podemos perceber que as frequências no centro são as baixas e as mais distantes são as altas, sabendo dessa informação podemos aplicar vários filtros para termos resultados que se adequem ao que se quer.

**Passa-baixa:** remove as altas frequências, assim removendo os detalhes das imagens, como as bordas. Para aplicar o filtro usamos uma circunferência centrada no meio do espectro, quanto maior o raio, mais se preserva da imagem, se for muito pequeno começa a ter perdas significativas, pois começa a remover as baixas frequências.

**Passa-alta:** remove as baixas frequências, assim obtendo os detalhes da imagem. Pode ser usado para a detecção de bordas.

Podemos realçar os detalhes das imagens somando as altas frequências com a imagem original.

**Ideal lowpass filters:** Faz a binarização da imagem, mantendo somente as baixas frequências.

**Butterworth lowpass filters:** Aplica a binarização, mas tendo valores intermediários entre 0 e 1.

**Gaussian lowpass filters:** é parecido com a técnica de Butterworth, mas a inclinação da descida de 1 para 0 é definida pela constante  $D$ , onde quanto maior mais suave é a curva.

O high-boost Filtering fortalece a imagem original (altas e baixas) e subtrai a baixa.

A filtragem homomorfica trata a imagem como uma combinação de iluminação e reflectância. A vantagem é que se obtém um controle sobre as componentes de iluminação e reflectância, fazendo uma compressão da escala dinâmica e realce de contraste simultâneos.

## Segundo Bimestre

### Aula 8

Restauração de imagem: é um processo que pega uma imagem que foi degradada de alguma forma, seja por um ruído ou que envolve algum tipo de espalhamento de luz. Ele tem como objetivo estimar a imagem real, pois não é possível obtê-la exatamente igual, pois essa informação foi perdida pela degradação. Esse processo é útil em várias áreas como astronomia, radioastronomia, imagens médicas, entre outros.

Uma imagem degradada é obtida pela convolução de uma função de espalhamento  $h$  sobre uma imagem  $f$ , adicionando um ruído  $b$ , obtendo a imagem degradada  $g$ .

Supondo que  $h$  seja uma matriz identidade, então só ficamos com os ruídos. O ruído nunca é conhecido, mas pode ser caracterizado pela média e variância. O ruído-branco tem espectro de frequência uniforme, já o gaussiano não. A função densidade de probabilidade de uma variável gaussiana  $z$  é:

$$p(z) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(z-\mu)^2/2\sigma^2}$$

$z$  é a intensidade;  
 $\mu$  é o valor médio;  
 $\sigma$  é o desvio padrão.

Assim 70% dos valores estarão no intervalo:  $[u-\sigma, u+\sigma]$  e 95% dos valores estarão no intervalo:  $[u-2\sigma, u+2\sigma]$ .

Existem outras funções densidade de probabilidade, como Rayleigh, Erlang (Gama), Exponencial, Uniforme, Impulsivo (sal e pimenta).

O ruído sal e pimenta é adicionar pixels em posições aleatórias, podendo assumir valores sal = branco ou pimenta = preto.

O ruído com padrão regular é algo que se repete periodicamente na imagem.

Para diminuí-lo é preciso calcular a transformada de Fourier, remover os picos incomuns e calcular a inversa da transformada.

Filtros de ruídos no domínio da imagem:

**Filtro da média:**

Aritmética  $\hat{f}(x, y) = \frac{1}{mn} \sum_{(s,t) \in S_{xy}} g(s, t)$

Geométrica  $\hat{f}(x, y) = \left[ \prod_{(s,t) \in S_{xy}} g(s, t) \right]^{\frac{1}{mn}}$

Harmônica  $\hat{f}(x, y) = \frac{mn}{\sum_{(s,t) \in S_{xy}} \frac{1}{g(s, t)}}$

Contra-harmônica  $\hat{f}(x, y) = \frac{\sum_{(s,t) \in S_{xy}} g(s, t)^{2+1}}{\sum_{(s,t) \in S_{xy}} g(s, t)^2}$

Filtro do mínimo pega uma região 3x3 e coloca o pixel mínimo, assim predominando as áreas mais escuras.

O filtro máximo, faz basicamente a mesma coisa, porém pega o pixel máximo, assim predominando as áreas mais claras.

Os filtros adaptivos pegam as características locais da imagem e aplica o filtro que melhor se adapta para o caso.

Supondo:  $g(x,y)$  o valor da imagem com ruído em  $x,y$ ;  $\sigma^2n$  a variância do ruído que corrompe  $f(x,y)$  para formar  $g(x,y)$ ;  $mL$  a média local dos pixels;  $\sigma^2L$  a variância local dos pixels.

O filtro deve: 1) Se  $\sigma^2n$  for zero, o filtro retorna  $g(x,y)$ , pois o ruído é zero  
2) Se  $\sigma^2L$  for alta em relação a  $\sigma^2n$ , deve retornar  $g(x,y)$ , pois deve ser uma região de bordas;

3) Se  $\sigma^2L$  e  $\sigma^2n$  são iguais, retornar o valor da média da região.

O efeito que um sistema de aquisição de imagem tem em uma fonte pontual de imagem é conhecido por função de espalhamento do ponto (PSF - point spread function). Para determinar a função de espalhamento de uma imagem  $f$  tendo a degradada  $g$ ,  $g=h*f$ , utilizamos a transformada de Fourier para obter  $H$ , então  $H = G/F$ .

Podemos usar a PSF gaussiana para experimentar  $h$  desconhecidos:

$$h(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Para determinar a qualidade de  $H$  usamos a minimização de  $\|g-h*f\|^2$ .  
Podemos usar o olho humano para determinar a qualidade visível da imagem, assim experimentando vários  $h$  até ter um resultado satisfatório.

Filtragem inversa: assume a degradação como um filtro invariante no espaço onde  $g = f*h$ .

Um modelo melhor é  $g=f*h+n$ , onde  $n$  é o ruído, assim a transformada de

$$F(\omega_1, \omega_2) = \frac{G(\omega_1, \omega_2) - N(\omega_1, \omega_2)}{H(\omega_1, \omega_2)}$$

Fourier fica: , fazendo a transformada inversa obtém a imagem.

$$F(\omega_1, \omega_2) = \frac{|H(u, v)|^2}{|H(u, v)|^2 + K} G(u, v)$$

Já no filtro de Wiener

Onde  $H$  é a transformada da função de degradação e  $K$  é uma constante (deve ser a variância do ruído).

Quando se tem um grupo de objetos e uma imagem borrada, usamos os momentos para decidir a qual objeto a imagem se refere.

Quando se tem um movimento relativo entre o sensor e o objeto, a imagem sofre um borramento. Para se fazer a desconvolução é necessário fazer linha por linha. Onde a linha da imagem borrada  $g[m]$  corresponde a linha da

$$g[m] = \sum_k h[m-k].f[k]$$

imagem original  $f[m]$ , assim:

Transformação geométrica: Procura encontrar a deformação sofrida pela imagem, usando pontos de controle, para então, realizar a transformação inversa. Dada uma imagem original  $f(x,y)$  e a sua degradada  $g(i,j)$ , com  $i = r(x,y)$  e  $j = s(x,y)$ ,  $r(x,y) = c_1x + c_2y + c_3xy + c_4$ ,  $s(x,y) = c_5x + c_6y + c_7xy + c_8$ , ou seja  $i = c_1x + c_2y + c_3xy + c_4$  e  $j = c_5x + c_6y + c_7xy + c_8$ .

### Aula 9

A segmentação de imagens serve para separar objetos da imagem, os algoritmos são baseados em duas propriedades dos níveis de cinza: similaridade e descontinuidade.

Similaridade: baseado em limiarização, crescimento de regiões e divisão/fusão de regiões.

Descontinuidade: dividem a imagem quando há uma significativa mudança nos níveis de cinza.

A limiarização e binarização são as abordagens mais simples da segmentação.

A técnica P-TILE é usada para achar o limiar da imagem, para isso precisa-se saber a área do objeto, com a  $p\%$  de ocupação do objeto na imagem o limiar  $T$  é o valor acima que resulta em  $p\%$  (supondo o objeto com altos e o fundo com valores baixos).

Pelo método de OTSU o limiar é obtido classificando a imagem em duas classes, que são objeto e o fundo.

O método de PUN define duas entropias para a imagem, do objeto e do fundo, então o valor limiar é o valor máximo da soma das entropias.

No método de KAPUR, SAHOO e WONG é parecido com o de PUN, mas usa uma fórmula diferente para calcular a entropia.

Podemos usar esses conceitos para aplicar a limiarização localmente.

Na técnica de split and merge, a imagem é dividida até que as regiões tenham apenas uma tonalidade, cor ou textura, então regiões similares e adjacentes são unidas.

No crescimento de regiões pega um ponto do objeto e dele pontos adjacentes similares são unidos, se repetindo até não ser mais possível.

A maneira mais comum de detectar descontinuidades é utilizando uma máscara de convolução adequada. Mas linhas de um pixel são muito incomuns nas imagens. A abordagem mais comum para a detecção de descontinuidades significantes nos níveis de cinza é a detecção de bordas.

Filtros de aguçamento: usa-se operações equivalentes a derivadas para realçar bordas. Para encontrar as bordas podemos usar técnicas de Sobel, Roberts, Prewitt descritas na Aula 5.

NEVATIA E BABU - sugere a aplicação de máscaras particulares, que apresentam uma maior resposta para bordas com inclinações específicas.

0	-1	0	-1	-1	-1
-1	4	-1	-1	8	-1
0	-1	0	-1	-1	-1

Podemos usar a detecção Laplaciana onde usa vizinhança-4 ou 8, mas não da a direção das bordas e é muito sensível a ruídos, normalmente é aplicada uma suavização antes.



Para remover o ruído após usar algoritmos baseados em descontinuidade podemos aplicar uma limiarização da imagem.

Segmentação de Imagens: Primeiro é aplicada uma suavização na imagem para reduzir os ruídos, aplicamos a detecção de bordas, limiarizamos as bordas para eliminar bordas insignificantes, afinamos as bordas para uma melhor definição e por fim conectamos pixels que fazem parte da mesma linha.

O método mais conhecido para deixar bordas com um pixel de espessura é o Zhang-Suen, que opera em imagens binárias, baseado no eixo central.

Afinamento por supressão Não máxima: Mantém o pixel de borda com a tonalidade mais alta (pixel mais forte), mesmo que ele não esteja no centro da borda.

Ligação de bordas: após a detecção de bordas podem haver linhas com descontinuidades por causa de ruídos, então essa técnica tem o objetivo de tornar as linhas mais regulares. Podemos fazer isso processando localmente e os pixels que apresentam similaridades são conectados.

Conexão: Varre os pixels e os rotula, pixels que tem a mesma direção recebem o mesmo rótulo, pode-se verificar a magnitude, guarda o ponto inicial, ponto final, direção e número de pixels de uma reta. Podemos considerar 8 ou 12 direções. Podemos usar a técnica de inundamento para fazer a conexão de retas.

O algoritmo Canny é ótimo para detecção de bordas, deixando-as finas e sem quebras. Ele tem alguns passos: Redução de ruído gaussiano, cálculo dos gradientes por Sobel, afinamento das bordas por supressão não máxima, limiarização dupla e Histeresys (transforma pixels fracos em fortes) que consistem em analisar a intensidade dos pixels em relação à sua vizinhança se tem um pixel vizinho forte, ele é mantido, se todos os vizinhos são fracos, ele é descartado.

Podemos fazer a ligação de bordas e detecção de fronteiras globalmente através do método da transformada de Hough em círculos consiste em passar uma circunferência em cada ponto, o ponto com maior intersecções é o centro dos pontos iniciais. Podemos executar essa técnica em retas também onde: Dada a equação da reta com os parâmetros x e y,  $y = ax + b$ ,  $b = y_i - ax_i$ . Cada ponto  $p_i$ , de coordenadas  $(x_i, y_i)$ , gera uma equação  $b = y_i - ax_i$ . A interseção das retas no novo espaço a-b definem os valores de a e b (da reta a ser determinada no espaço x-y. Mas se tivermos vários pontos alinhados encontraremos linhas que não pertencem ao mesmo segmento.

Utilização de movimentos na segmentação:

$$d_{ij}(x,y)=\begin{cases} 1 & \text{se } |f(x,y,t_i)-f(x,y,t_j)| > \theta \\ 0 & \text{caso contrário} \end{cases}$$

Diferença cumulativas:

Imagem de referência: Deve se ter uma imagem que possa ser comparada. Caso não a tenha é necessário construir uma, a partir de um conjunto de imagens com objetos em movimento.

### Aula 10

Quando se cria uma imagem digital podemos obter uma grande quantidade de dados, sendo difícil e custoso o seu processamento, então foram desenvolvidas técnicas de compressão de imagens.

Armazenamento de imagem é guardar a imagem em mídias magnéticas ou em arquivos. Transporte de imagem é a transferência eletrônica através de um link. Alguns fatores devem ser levados em consideração para a compressão da imagem: Quanto deve ser diminuído os dados da imagem, distorções causadas pela compressão e a complexidade para comprimir, abrir e salvar a imagem. Compressão significa diminuir uma imagem sem perdas do conteúdo, já a compactação aceita perdas. Pode-se aplicar os métodos no domínio do espaço ou da frequência. A mais usada no domínio da frequência é a transformada discreta do cosseno, utilizado na JPEG.

Redundância: Quando se tem unidades de transporte com a mesma informação, temos a redundância relativa  $R_D = 1 - (1/C_R)$ , onde  $C_R = n_1/n_2$  é a taxa de compressão.

Utilizando o histograma da imagens, e  $r_k$  no intervalo  $[0,1]$  é os níveis de cinza e  $p_r(r_k) = nk/n$ , com  $k = 0, \dots, L-1$  é a probabilidade de cada nível acontecer. Assim o número médio de bits para representar a imagem é:

$$L_{\text{médio}} = \sum_{k=0}^{L-1} l(r_k) p_r(r_k)$$

Codificação de comprimento variável: Usa menos bits para tons que mais ocorrem (>probabilidade) e usa mais bits para tons que menos ocorrem

(<probabilidade).

Redundância interpixel: É a diferença entre um pixel e o próximo. Quando fazemos o histogramas e percebemos padrões frequentes, então podemos dizer que há uma alta redundância interpixel.

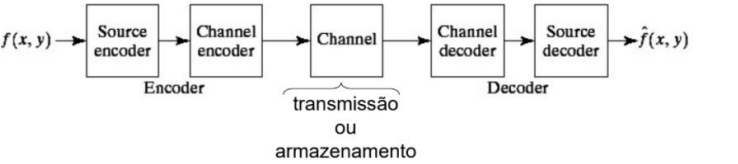
Para reduzir a redundância podemos calcular a diferença entre os pixels adjacentes e representa-los por ela. Nomeando-se mapeamento, se podemos reconstruir a imagem original com esse mapeamento o chamamos de reversível.

Quando temos uma imagem onde a diferença do pixel para o próximo é pequena, fica mais fácil de armazená-la, quando os pixels são iguais é só armazenar o valor do pixel e quantas vezes ocorre.

Ex.

supondo a imagem **[88|88|88|88|88|88|88|88|88]** → 9 bytes  
toma-se as diferenças **[88|0|0|0|0|0|0|0|0]**  
guarda-se as quantidades **[88|9]** → 2 bytes

Desenhos tem uma redundância maior do que em imagens. A compressão de imagem consiste em dois blocos, o codificador que reduz a imagem e o decodificador que a reconstrói



A da teoria da informação diz que a geração da informação pode ser modelada como um processo probabilístico, que pode ser medido de maneira que concorde com nossa intuição. Assim, um evento aleatório E que ocorra com probabilidade P(E) contém:

$$I(E) = \log \frac{1}{P(E)} = -\log P(E)$$

log de base 2;

I(E) é chamada auto-informação de E. Quando transferimos a auto-informação dizemos que a fonte está conectada ao usuário por um canal de informação (internet). A fonte de informações gera uma sequência aleatória de símbolos  $\{a_1, a_2, \dots, a_j\}$  denominado alfabeto fonte. A informação média é chamada de entropia da fonte, definida

$$H(z) = - \sum_{j=1}^J P(a_j) \log P(a_j)$$

Chamamos de estimativa de primeira ordem. Estimativa de segunda ordem: Para melhorar o resultado agrupamos pixels e encontramos a frequência, de acordo que o tamanho do agrupamento se aproxima do infinito a estimativa se aproxima do valor real.

Se houver diferença entre a primeira ordem e ordens mais elevadas, então significa que há redundância inter-pixel, podendo melhorar a compressão. De acordo que aumentamos a ordem fica computacionalmente mais caro. Em alguns casos só podemos usar a compressão livre de erros, as principais técnicas são: Codificação por tamanho variável (Codificação de Huffman), Codificação por planos de bits e Codificação previsora sem perdas. Codificação de Huffman, primeiro cria-se uma série de redução de fonte

Original source		Source reduction			
Symbol	Probability	1	2	3	4
$a_2$	0.4	0.4	0.4	0.4	0.6
$a_6$	0.3	0.3	0.3	0.3	0.4
$a_1$	0.1	0.1	0.2	0.3	
$a_4$	0.1	0.1	0.1		
$a_3$	0.06				
$a_5$	0.04				

Depois a codificamos, começando com a menor fonte continuando para trás, até a fonte original.

Original source			Source reduction			
Sym.	Prob.	Code	1	2	3	4
$a_2$	0.4	1	0.4	1	0.4	1
$a_6$	0.3	00	0.3	00	0.3	00
$a_1$	0.1	011	0.1	011	0.2	010
$a_4$	0.1	0100	0.1	0100	0.1	011
$a_3$	0.06	01010	0.1	0101		
$a_5$	0.04	01011				

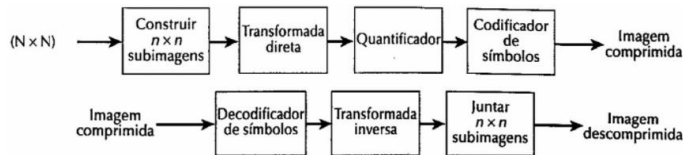
Então o tamanho médio é dado pela soma das probabilidades multiplicado pelo tamanho do código.

Codificação por planos de bits: Primeiro decompomos a imagem em várias imagens binárias, em seguida, comprimimos, usando uma técnica de compressão para dados binários, representando com o código gray.

Codificação binária: dividi-se a imagem até as regiões terem a mesma tonalidade.

Codificação de corrida unidimensional (RLE): processa em linha, armazenando a frequência e o valor.

Compressão com erro: Usando a transformada, mais comum a DCT



Quando temos imagens muito grandes a dividimos em blocos e processamos individualmente, quanto maior o bloco menor o erro, normalmente usa-se 8x8. A transformada resulta em números reais então a quantizamos para reduzir o número de bits utilizados.

$$Q[i][j] = 1 + (1 + i + j) * \text{fator\_de\_qualidade}$$

$$\text{Valor\_quantizado}[i][j] = \frac{\text{DCT}[i][j]}{Q[i][j]}$$

Fator\_de\_qualidade: 1 a 25

Utilizamos a curva de quantização para obter os valores que dividirão os coeficientes, os coeficientes menores são as altas frequências, que não causam grande impacto se omitidos.

Quando usamos a DCT guardamos até o último número diferente de 0, o resto é descartado. Para reconstruir a imagem preenchemos com 0 os valores que faltarem.

Essas compressões podem ser usadas em vídeos, aplicando em cada frame.

## Aula 11

Para representarmos e descrevermos objetos precisamos extrair medidas numéricas dos objetos que precisam ser classificados, objetos estes contidos nas imagens.

Podemos classificar-los pela sua altura, largura, perímetro, área, textura, cor, entre outros.

Código da cadeia: usado para representar retas conectadas, onde pode ter 4 ou 8 direções. A partir de um ponto inicial representamos os objetos em uma malha e o nomeamos de acordo com sua direção, assim tendo um código. O código pode mudar dependendo de onde se inicia, podemos normalizá-lo, rodando o código até se ter o menor número possível.

Podemos representar as bordas dos objetos como um polígono assim sendo possível encontrar o seu perímetro. Podemos usar o método de aproximações sucessivas, onde começamos com dois pontos mais distantes e traçamos uma reta, depois traçamos uma perpendicular, até se obter o resultado aceitável. Assinatura é a distância do centro até a borda de acordo com o seu ângulo, ela é invariante a translação, mas sofre efeitos de rotação e escala. Podemos normalizá-la em relação a rotação:

- Iniciar sempre no ponto mais afastado do centróide (quando existir um único ponto que satisfaça esta condição);
- Selecionar o ponto no eixo mais alongado;
- Obter a assinatura e depois tratá-la, do mesmo modo usado com o código cadeia.

O fecho convexo de um objeto é o menor conjunto convexo que é capaz de representá-lo. A fronteira pode ser particionada anotando os pontos que ela entra e sai do fecho. As razões entre os comprimentos adotados é invariante a

escala, rotação e translação, também se pode obter um grafo a partir dos pontos e proceder a análise sobre ele.

Para representá-los de uma forma qualquer podemos reduzi-lo a um gráfico através de seu esqueleto. O esqueleto pode ser obtido através de técnicas de afinamento, por exemplo, transformação do eixo médio (MAT).

Uma região R com borda B. Para cada ponto p em R, encontre seu vizinho mais próximo em B. Se p tem mais que um vizinho, diz-se que ele pertence ao esqueleto de R.

Afinamentos de imagens binárias usa-se Zhang e Suen, objeto = 0 e fundo = 1, elimina os pixels externos até ficar somente o esqueleto. Esse processo tem 2 passos.

1: Um pixel é marcado para ser eliminado se satisfizer 4 condições

a)  $2 \leq N(p1) \leq 6$

b)  $S(p1) = 1$

c)  $p2, p4, p6 = 0$

d)  $p4, p6, p8 = 0$

Onde:  $N(p1)$  é o número de vizinhos não nulos de  $p1$   $N(p1) = p2 + p3 + \dots + p8 + p9$  e  $S(p1)$  é o número de transições 0-1 na sequência.

Depois de percorrer todos os pontos devemos excluir os marcados.

2: Deve satisfizer as condições a e b e

c')  $p2, p4, p8 = 0$

d')  $p2, p6, p8 = 0$

para serem marcados e após processar todos, excluir os marcados.

Descritores de fronteiras:

Comprimento do contorno: conta os pixels do contorno, oferece uma aproximação do comprimento. No caso de uma fronteira representada pelo código da cadeia, os cálculos ficam facilitados, observando que pixels bordas horizontais e verticais tem tamanho 1 e bordas diagonais tem tamanho  $\sqrt{2}$ .

Diâmetro da fronteira: é a maior distância de dois pontos do contorno:

$$\text{Diâmetro}(B) = \max[D(p_i, p_j)]$$

Curvatura: é definida pela taxa de mudança da inclinação, podendo ser aproximada pela primeira diferença.

O número de forma é definido pela diferença de menor magnitude do código da cadeia. O primeiro número recebe a diferença entre o último e o primeiro elemento. Podemos rotacionar para obter o menor número, chamando de Shape  $n^\circ$ .

Descritores de Fourier: é fronteira digital com n pontos no plano xy, começando em um ponto  $x_0, y_0$ .

Onde cada número é complexo  $s(k) = x(k) + y(k)i$ .

$$a(u) = \frac{1}{N} \sum_{k=0}^{N-1} s(k) \exp[-j2\pi uk / N]$$

Se mantermos alguns termos iniciais, ainda se tem uma boa aproximação da fronteira.

Descritores regionais:

Área – número de pixels dentro da fronteira.

Perímetro - é o comprimento da fronteira.

Estas duas medidas são variantes em relação a escala

Compacidade - é definida por  $\text{perímetro}^2 / \text{área}$  é uma medida sem dimensão e também invariante a rotação.

Há outros descritores como, média e mediana dos níveis de cinza, valores mínimos e máximos, número de pixels com valor acima e abaixo da média.

Descritores topológicos: A topologia é o estudo das propriedades de uma forma que não é afetada por deformações, na medida que não ocorre divisão ou fusão da forma.

Podemos usar o número de componentes conexos como descritor, contando o número de objetos e de buracos, assim podendo obter o número de Euler: componentes – buracos.

Redes poligonais, possuem uma interpretação simples do número de Euler, onde: W : número de vértices; Q : número de arestas; F : número de faces

Assim Euler:  $W - Q + F = C - H = E$ .

Descritores de texturas: algumas medidas podem dizer se a textura é suave, rugosa, ou regular. Usamos os momentos para a caracterização de texturas, sendo calculado como:

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y)$$

Podemos normalizá-lo com:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma} \quad \text{onde} \quad \gamma = \frac{p+q}{2} + 1$$

Momentos de Hu: podemos obter sete momentos que são invariantes a partir do segundo e terceiro momentos.

Abordagem estatística: Utiliza os momentos, obtidos a partir dos histogramas de tons de cinza, podemos normaliza-lo, assim se o valor for baixo é suave, médio rugoso e alto regular.

O uso de matrizes de co-ocorrência de níveis de cinza, determina as ocorrências de pares de pixels em uma disposição como: um pixel abaixo e a direita.

Abordagem espectral: O espectro de Fourier é ótimo para descrever padrões:

$$S(r) = \sum_{\theta=0}^{\pi} S_{\theta}(r)$$

$$S(\theta) = \sum_{r=1}^R S_r(\theta)$$

1 – Picos predominantes no espectro fornecem a direção dos padrões.  
2 – A posição dos picos fornece o período espacial fundamental dos padrões. 3 – A eliminação de componentes periódicas (filtragem) deixa elementos não periódicos, que podem ser descritos por técnicas estatísticas. Representando o espectro em coordenadas polares podemos obter dois descritores globais importantes somando os valores  $S_r(\theta)$  e  $S_{\theta}(r)$ .

Metodologias típicas: Digitaliza e binariza as imagens, aplica vários descritores e se observa qual melhor descreve os objetos.

## Aula 12

A morfologia é uma ferramenta para extrair características das imagens para representá-las e descrevê-las. Podem ser aplicadas técnicas de pré e pós-processamento, como: Filtragem morfológica, afinamento e poda.

Morfologia é baseado nos conjuntos, onde uma imagem binária com o objeto = 1 e fundo = 0, o conjunto dos pixels = 1 representa os objetos e os pixels = 0 representa o fundo.

Dilatação: denotada por :

Primeiro se obtém a reflexão de B, depois a

$$A \oplus B = \{x \mid (\hat{B})_x \cap A \neq \emptyset\}$$

reflexão  
translação

translação por x. O conjunto B pode ser visto como uma máscara de convolução. A dilatação ajuda a tampar buracos, por exemplo, em letras de textos. Elemento estruturante:

0	1	0
1	1	1
0	1	0

Erosão: Denotado por :

Todos os pontos x onde B reflexão, translado por x está contido em A.

Ajuda a separar objetos que estão colados

A dilatação e erosão são operações complementares, onde:

A complementação da Erosão de A por B é igual a dilatação da

complementação de A dilatada pela reflexão de B.

$$(A \ominus B)^c = A^c \oplus \hat{B}$$

A morfologia pode realizar remoção de ruídos, detecção de bordas, eliminação de objetos na imagem, localização de objetos na imagem, etc.

Abertura: suaviza o contorno de uma imagem, quebrando ligações estreitas e eliminando protuberâncias finas. Definida pela erosão de A por B, depois a dilatação do resultado por B.

$$A \circ B = (A \ominus B) \oplus B$$

Fechamento: suaviza os contornos, funde as quebras em linhas finas, elimina buracos e preenche fendas em um contorno. Definida pela dilatação de A por B, depois a erosão do resultado por B.

$$A \bullet B = (A \oplus B) \ominus B$$

A abertura e o fechamento podem excluir objetos pequenos. São operações complementares.

Interpretação geométrica abertura e fechamento: usando um elemento estruturante B redondo, a fronteira será dada, pelo mais perto que a bola consegue chegar da borda do objeto.

Transformada Hit-or-miss: detecta bordas, por exemplo, encontra a posição de um objeto específico. Supondo que a origem de cada forma está em seu centro e que W é uma pequena janela que cobre X.

$$A \circledast B = (A \ominus X) \cap [A^c \ominus (W - X)]$$

Aplicando assim, o elemento será somente o seu pixel central.

Extração de fronteira: Aplica a erosão no objeto, removendo sua borda, depois faz o resultado menos o elemento original, ficando só com a fronteira.

Preenchimento de regiões: é obtido reagindo a dilatação, complementação e interseções.

$$X_k = (X_{k-1} \oplus B) \cap A^c \quad k = 1, 2, 3, \dots$$

Com isso podemos eliminar pontos indesejados dentro de objetos.

Fecho convexo: podemos obtê-lo usando quatro elementos estruturantes  $B_i$ , com  $i = 1, 2, 3, 4$ . Repete-se esses passos:

$$X_k^i = (X_{k-1} \circledast B^i) \cup A \quad i = 1, 2, 3, 4 \quad \text{and} \quad k = 1, 2, 3, \dots$$

Seja  $D^i = X_{con}^i$ , sendo dado o fecho por  $C(A) = \bigcup_{i=1}^4 D^i$

Afinamento: denotado por  $A \otimes B = A - (A \circledast B)$

Onde se faz hit-or-miss de A por B, com o resultado faz menos A.

$$A \odot B = A \cup (A \circledast B)$$

Espessamento: denotado por

Onde se faz hit-or-miss de A por B, com o resultado faz união com A.

Esqueleto: é obtido através de múltiplas erosões, onde se faz até a erosão ser vazia e pega a erosão anterior.

Poda: é uma operação complementar ao esqueleto, onde pode ficar pixels soltos, a poda os remove. Ela consiste em eliminar as extremidades das imagens, mas ela também elimina as dos objetos. Após o processamento as extremidades dos objetos são recuperados. Sendo:

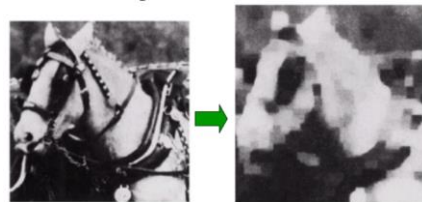
$$X_2 = \bigcup_{i=1}^8 (X_1 \circledast B^i) \quad \text{seguido de} \quad X_3 = (X_2 \oplus H) \cap A$$

No final faz  $X_4 = X_1 \cup X_3$ .

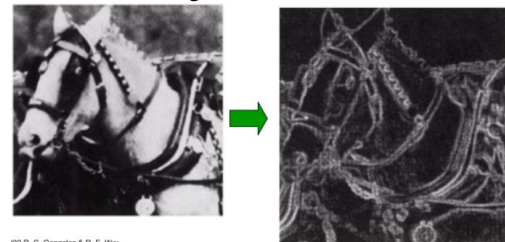
Dilatação em níveis de cinza: pega o elemento máximo de  $f(s-x, t-y) + b(x, y)$  e atribui ao pixel.

Erosão em níveis de cinza: pega o elemento mínimo de  $f(s+x, t+y) - b(x, y)$  e atribui ao pixel.

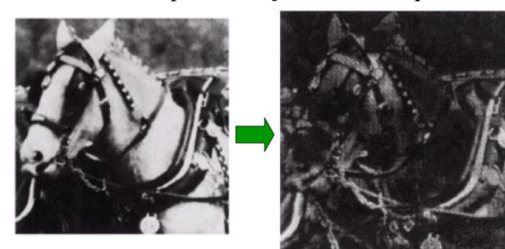
Suavização morfológica: Aplica uma abertura seguida de um fechamento.



Gradiente morfológico:  $g = (f \oplus b) - (f \ominus b)$



Transformada top-hat: realça os detalhes quando há sombreamento.



Segmentação por textura: Faz fechamentos seguidos, com elementos estruturantes cada vez maiores, assim quando o estruturante for do tamanho de objetos para serem eliminados, os elimina e deixa um fundo claro no lugar.