

# Trabalho 1

Implemente sua atividade sozinho sem compartilhar, olhar código de seus colegas, ou buscar na Internet. Procure usar apenas os conceitos já vistos nas aulas.

## Descritor visual usando autocorrelograma de cores

Nesse trabalho você deverá implementar e utilizar um descritor visual para representar imagens.

O correlograma de cores descreve a distribuição global da correlação entre a localização espacial de cores. Pode ser visto como uma tabela indexada de pares de cores  $P(c_i, c_j, d)$  especifica a probabilidade de encontrar um pixel de cor  $c_j$  a uma distância  $d$  de um pixel de cor  $c_i$  na imagem.

O descritor de autocorrelograma de cores da imagem é a diagonal principal da matriz do correlograma. Para uma dada imagem  $f$ , uma cor  $c_i$  e uma distância  $k$ :

$$\Gamma_{c_i}^k(f) \equiv |\{p_2 \in f_{c_i}, p_1 \in f_{c_i} \mid |p_1 - p_2| = k\}|, \quad (1)$$

que pode ser interpretado como o total da ocorrência de cores iguais entre pixels  $p_1$  e  $p_2$  a uma distância  $k$  um do outro, numa mesma imagem.

Para computar o descritor, escolhe-se um conjunto de distâncias  $k = [d]$  e uma métrica de dissimilaridade.

Utilizaremos a distância chessboard (distância de Chebyshev, norma  $L^\infty$ ) – kernel quadrado:

$$dChess(p, q) = \max(|p_x - q_x|, |p_y - q_y|), \quad (2)$$

ou seja, o máximo da distância nas direções  $x$  e  $y$  entre os pixels  $p$  e  $q$ .

## Tarefa

- Trabalho individual
- Data de entrega: 17/11/2011

Escreva um programa numa linguagem de sua escolha que realize as seguintes tarefas:

1. Abra duas imagens coloridas e carregue-as na memória principal
2. Reduza o espaço de cores para 64 cores (quantização).
3. Compute o autocorrelograma de cores das duas imagens para o conjunto de distâncias  $k = 1$  e  $3$  considerando a distância chessboard (norma  $L^\infty$ ). Não se esqueça de normalizar o descritor para que sua soma seja unitária.

4. Calcule a diferença entre os dois descritores usando a distância de Manhattan (norma  $L^1$ )
5. Imprima na tela: os descritores de cada imagem e a diferença entre eles.

**Atenção:** Os passos 3 e 4 devem ser implementados pelo aluno, sem usar função pronta de qualquer biblioteca para computar o correlograma, o autocorrelograma ou as distâncias.

## Requisitos

Devem obrigatoriamente ser implementadas as funções:

- `void color_autocorrelogram(IplImage *I, int k, double *desc, ...)` que retorne na variável `desc` o autocorrelograma de cores da imagem `I` para a distância `k`, podendo usar outros parâmetros se desejar.
- `double distChessboard(double *a, double *b, int N, ...)`, que retorne a distância de chessboard entre dois vetores `a` e `b`, ambos de tamanho `N`, podendo usar outros parâmetros se desejar.

Observação: a assinatura das funções acima usam a sintaxe da linguagem C, o que pode ser modificado se você usar linguagem de outra sintaxe.

## Instruções

O projeto será avaliado principalmente levando em consideração:

1. Processamento correto do autocorrelograma de cor (80%).
2. Implementação e uso correto da distância de Chessboard (10%).
3. Implementação e uso correto da distância de Manhattan, gerenciamento da memória, carregamento das imagens, cálculo e impressão da diferença (10%).

Dúvidas conceituais deverão ser colocadas nos horários de atendimento. Dificuldades em implementação, por favor, envie e-mail para a estagiária PAE com o assunto `[trab_cor]duvida`, anexando o código e especificando o problema.

A detecção de cópia de parte ou de todo código-fonte, de qualquer origem, implicará reprovação direta no trabalho. Partes do código cujas **ideias** foram desenvolvidas em colaboração com outro(s) aluno(s) devem ser devidamente documentadas em comentários no referido trecho. O que **NÃO** autoriza a cópia de trechos de código. Portanto, compartilhem ideias, soluções, modos de resolver o problema, mas **não o código**. Qualquer dúvida entrem em contato com o professor.