

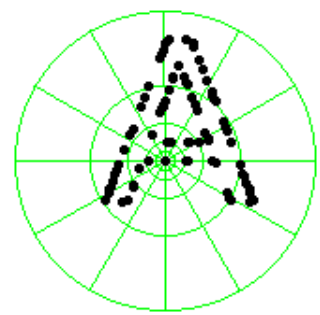
CASAMENTO DE FORMAS COM SHAPE CONTEXT

ALUNO: DIEGO MIRA DAVID - SUPERVISOR: RONALDO FUMIO HASHIMOTO
TRABALHO DE FORMATURA SUPERVISIONADO - INSTITUTO DE MATEMÁTICA E ESTATÍSTICA

INTRODUÇÃO

Este trabalho consiste no estudo e implementação dos algoritmos para casamento de formas que utilizam o descritor *shape context*, proposto por **S. Belongie** e **J. Malik** inicialmente em “**Matching with Shape Contexts**”.

SHAPE CONTEXT



Considere uma amostra de pontos do contorno. Para cada ponto p_i da amostra é calculado um **histograma** das coordenadas relativas dos demais pontos do contorno. É utilizado um diagrama log-polar (ilustrado na imagem ao lado), particionado em 60 *bins*, de forma a tornar o descritor mais sensível às posições mais próximas de p_i . O histograma h_i , que é o *shape context* de p_i , incorpora informação global da forma ao armazenar a distribuição dos pontos nos *bins* e é calculado da seguinte forma:

$$h_i(k) = \#\{q \neq p_i : (q - p_i) \in \text{bin}(k)\}$$

O custo para casar um ponto p_i com um ponto q_j , que é uma medida da diferença entre os dois *shape contexts*, é dado por:

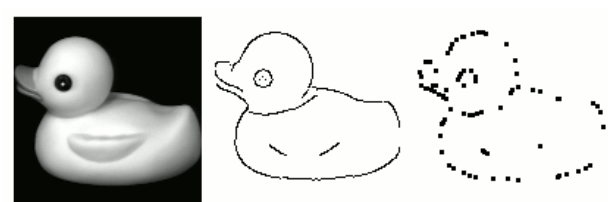
$$C(p_i, q_j) = \frac{1}{2} \sum_{k=1}^K \frac{[h_i(k) - h_j(k)]^2}{h_i(k) + h_j(k)}$$

Uma extensão (*Generalized Shape Contexts*) ainda mais rica do descritor considera vetores tangentes t_i de comprimento unitário para indicar a direção do contorno em cada ponto p_i .

$$\hat{h}(k) = \sum_{q_i \in Q} t_i \text{ onde } Q = \{q \neq p_i : (q - p_i) \in \text{bin}(k)\}$$

Dessa forma, cada *bin* carrega um vetor com a direção dominante da contorno dentro do *bin*. O custo para casar um ponto p_i com um ponto q_j é dado por:

$$C(p_i, q_j) = \sum_{k=1}^K \frac{\|\hat{h}_i(k) - \hat{h}_j(k)\|^2}{\|\hat{h}_i(k) + \hat{h}_j(k)\|}$$



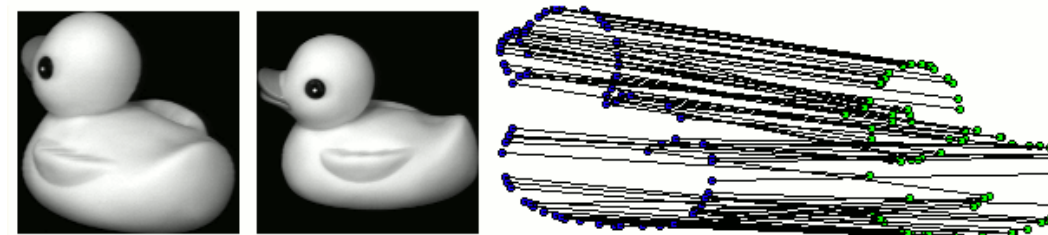
A imagem ao lado ilustra uma amostra de pontos do contorno. Foi implementado o algoritmo **Canny** de detecção de bordas.

CORRESPONDÊNCIA ENTRE OS PONTOS

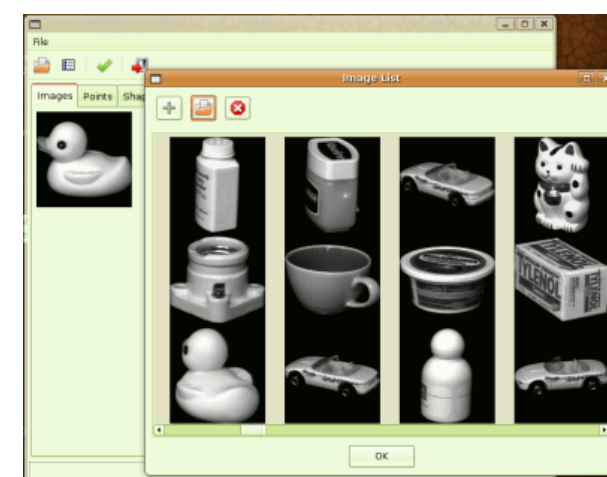
A correspondência entre os pontos das duas formas é obtida resolvendo um problema de atribuição de tarefas (*assignment problem*) com custos C_{ij} . Dado o conjunto dos custos C_{ij} de todos os pares de pontos p_i de uma forma e q_j da outra forma, desejamos encontrar uma **permutação** π que minimize o custo total, dado por:

$$H(\pi) = \sum_i C(p_i, q_{\pi(i)})$$

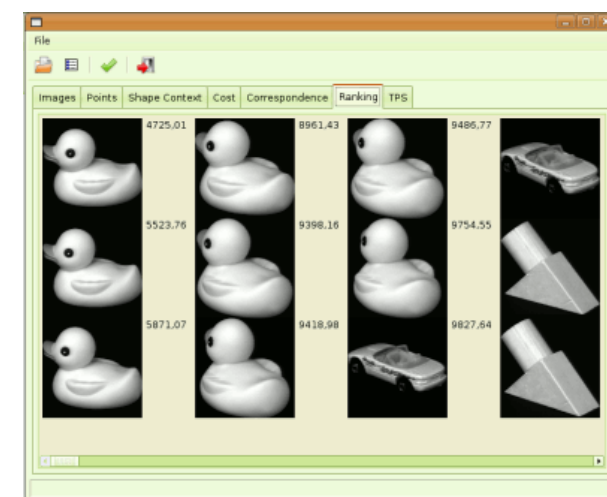
Foi implementado o algoritmo Kuhn-Munkres (também conhecido como Método Húngaro), um algoritmo de otimização que resolve o *assignment problem* em tempo $O(n^3)$.



O SOFTWARE DESENVOLVIDO



Dados um conjunto de imagens e uma imagem de consulta, o programa contrói um ranking com as imagens do conjunto mais semelhantes à imagem de consulta. O programa também fornece informações para acompanhar o funcionamento do método utilizado.

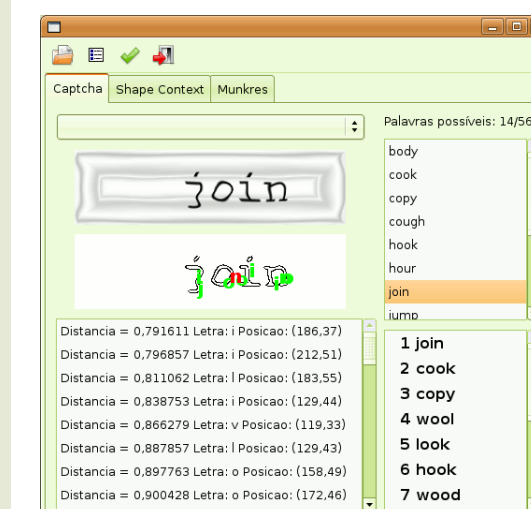


As imagens ao lado ilustram o funcionamento do programa para algumas imagens do dataset *Columbia Object Image Library (COIL-20)*. Na primeira tela estão as imagens de entrada e na segunda tela está o ranking com as imagens mais semelhantes.

- ▶ Linguagem: C++
- ▶ Biblioteca gráfica: wxWidgets
- ▶ Coleção de bibliotecas: Boost

CAPTCHAS

CAPTCHA é acrônimo de “*Completely Automated Public Turing test to tell Computers and Humans Apart*”. Trata-se de um teste que visa distinguir humanos de computadores para impedir acessos automatizados a um sistema. O uso deste dispositivo muitas vezes incorre em problemas de usabilidade e acessibilidade.



Atualmente, diversos projetos têm demonstrado que captchas visuais baseados em caracteres podem se quebrados por computadores. Neste trabalho foi estudado o algoritmo para quebrar o captcha EZ-Gimpy descrito em “**Recognizing Objects in Adversarial Clutter: Breaking a Visual CAPTCHA**”.

Algoritmo:

1. Localizar possíveis letras em diversas localizações.
2. Construir um grafo com todas as letras consecutivas que podem ser usadas para formar uma palavra.
3. Dentre todos os caminhos possíveis no grafo, são selecionadas apenas palavras reais com auxílio de um dicionário. Estas palavras são então avaliadas em um processo mais caro computacionalmente.

RESULTADOS

- ▶ Neste trabalho foi possível estudar e aplicar os conceitos que envolvem o uso do descritor *Shape Context*. Foi construído um software que permite analisar e testar o funcionamento do descritor.
- ▶ Maiores informações sobre o trabalho e resultados alcançados logo estarão disponíveis no site:

<http://www.linux.ime.usp.br/~diegomd/mac499/>

REFERÊNCIAS

- ▶ “Matching with Shape Contexts”, S. Belongie and J. Malik. CBAIVL 2000
- ▶ “Recognizing Objects in Adversarial Clutter: Breaking a Visual CAPTCHA”, G. Mori e J. Malik. CVPR 2003