Similar Image Retrieval

A Comparison Among SIFT, FAST, ORB and MSER

Guilherme Defreitas Juraszek,
Alexandre Gonçalves Silva
College of Technological Science
Santa Catarina State University (UDESC)
Joinville, SC - Brazil
guilherme.defreitas@gmail.com,
alexandre@joinville.udesc.br

Milton Roberto Heinen
Campus Bagé
Federal University of Pampa (UNIPAMPA)
Bagé, RS - Brazil
milton.heinen@unipampa.edu.br

***Abstract*—With the increase number of digital images available in the Internet there is a need for systems that provides efficient ways to retrieve similar images without rely on additional text information. This paper uses the descriptors extractor algorithms SIFT, FAST, ORB and MSER to retrieve similar images from a dataset. The result showed that MSER feature extractor combined with SIFT descriptors presented the best result than the other combinations evaluated.**

Keywords—image retrieval; sift descriptors; mser feature extractor

I. INTRODUÇÃO

Com o crescimento no volume de informação digital disponível cresce também a necessidade de ferramentas que facilitem a busca por informações relevantes em base de dados cada vez maiores. Informações textuais são adicionadas diariamente em redes sociais, blogs e sites de comércio eletrônico. Mecanismos de busca baseados em texto são amplamente estudados e diversas ferramentas estão disponíveis para consultas baseadas em textos mas quando se trata de busca baseada em conteúdo a partir de uma imagem não existem tantas opções. A popularização de equipamentos como câmeras e celulares equipados com câmeras resultou em um grande volume de informações no formato de imagens e vídeos disponibilizadas na internet. O objetivo deste trabalho é propor um método de busca de imagens semelhantes em uma base de dados a partir da consulta utilizando outra imagem. Nenhuma informação textual complementar será utilizada para auxílio na classificação. O trabalho realizado mostra um algoritmo capaz de identificar imagens semelhantes em um conjunto de imagens a partir de uma imagem consultada. Para avaliar a eficiência foram analisadas as combinações de algumas técnicas de extração de pontos chave e de descritores de acordo com Tabela 1. Na seção 2 serão apresentados os trabalhos encontrados na literatura e algumas técnicas existentes para extração de pontos chave e características. Na seção 3 a metodologia utilizada para a realização do experimento é detalhada. Na seção 4 é descrito o algoritmo proposto e as demais configurações relacionadas ao experimento. Na seção 5 é apresentado o resultado comparativo entre os dados obtidos no processamento do algoritmo utilizando a combinação das diferentes técnicas de extração de pontos chave e descritores. Ao final do trabalho são apresentadas as observações e a conclusão.

II. TRABALHOS RELACIONADOS

A busca por conteúdo em imagens é um assunto de grande interesse por grandes empresas no setor de mecanismos de buscas na internet. Liu [1] descreve em seu artigo um algoritmo para a criação de agrupamentos (clusters) contendo milhares de imagens semelhantes utilizando *k-nearest neighbour*. Além da utilização de informações retiradas da própria imagem, os mecanismos de busca utilizam informações sobre a credibilidade da página na qual esta inserida, adquirida através do conteúdo textual, atualizações, links de referências entre outros fatores como um importante indicativo na hora de agrupar as imagens e retornar buscas [2].

Além de auxiliar na busca por imagens em uma pesquisa algumas técnicas de agrupamento e identificação de padrões são usadas com o objetivo de identificar e remover conteúdo adulto dos resultados das buscas. Esta análise leva em consideração questões como a coloração da pele e detecção de rostos [3] e classificação ocorre por uma máquina de vetores de suporte treinada com os dados extraídos das imagens.

Diante do intenso dinamismo encontrado na internet onde conteúdos são atualizados a todo momento, alguns trabalhos sugerem a utilização de técnicas de aprendizado incrementais onde o algoritmo melhora o seu desempenho conforme novas imagens vão sendo adicionadas no decorrer do tempo sem a necessidade de realizar uma análise total em toda a base de dados. Tavares [4] mostra a utilização de um classificador baseado na floresta de caminhos ótimos utilizando realimentação por relevância na recuperação de imagens por conteúdo de maneira eficiente e eficaz.

Um bom algoritmo de identificação de pontos de interesse deve ser capaz de reconhecer e extrair descritores que são invariáveis a iluminação, rotação, escala e translação do objeto a ser identificado na imagem.

Pontos chaves ou *keypoints* são áreas de uma imagem com características salientes que se repetem em imagens de diferentes perspectivas de um mesmo objeto.

Mikolajczyk at al. [5] realizam um comparativo entre diversas técnicas para detecção de regiões de interesse em imagens de diferentes perspectivas e mostram um bom desempenho do algoritmo MSER (*Maximally Stable Extremal Regions*) comparado a algoritmos como Harris e Hessian.

A. *SIFT – Scale Invariant Feature Transform*

O algoritmo SIFT fornece um método para extração de características distintas e invariantes para o reconhecimento de pontos em um objeto em imagens de diferentes ângulos. Os descritores extraídos são invariantes a escala e rotação e possuem uma boa tolerância a ruídos, distorções decorrentes de diferentes perspectivas e mudanças de iluminação [6]. Lowe, criador do SIFT, ainda descreve uma abordagem para identificação de objetos utilizando um comparativo com um banco de descritores extraídos de outras imagens usando um algoritmo de vizinhos próximos. A implementação do algoritmo é dividida em duas partes, o detector e o descritor. As etapas de processamento são:

- Detecção de extremos: Nesta etapa o algoritmo identifica possíveis pontos de interesse utilizando a função de diferença de Gaussianas aplicadas a diversas escalas da imagem. Este procedimento permite encontrar pontos de interesse invariantes a escala e a orientação.
- Localização de pontos chave: Para cada candidato encontrado na etapa anterior são determinadas a localização e escala e métricas de estabilidade são calculadas resultando na escolha dos pontos mais estáveis.
- Atribuição de orientação: Uma ou mais orientações são atribuídas a cada ponto chave escolhido de acordo com o gradiente local da imagem.
- Extração do descritor: Os gradiente locais ao redor do ponto de interesse são mensurados e uma representação simplificada é extraída.

B. *FAST - Features from Accelerated Segment Test*

FAST é um algoritmo de detecção de cantos proposto com o objetivo de identificar pontos chaves em imagens com uma velocidade muito superior ao do SIFT [7]. Não possui informações sobre a orientação dos descritores e é altamente sensível a ruídos. O algoritmo propõe uma melhora de desempenho utilizando aprendizado de máquina e criação de uma árvore de decisão. A árvore é transformada em código C e compilada.

C. *ORB – Oriented Fast and Rotated Brief*

Algoritmo proposto por Rublee et al. [8] como um possível candidato para substituição do SIFT em ambientes de baixo poder de processamento ou aplicações que necessitem de processamento em tempo real. O ORB é invariante a rotação e possui um bom grau de tolerância a ruídos na imagem. O algoritmo utiliza as técnicas FAST e BRIEF realizando algumas melhorias. Adiciona informações de orientação no FAST e no BRIEF, a adição desta funcionalidade permite o cálculo das variantes de correlação e orientação, resolvendo um dos pontos fracos do BRIEF original que é a falta de invariância em rotações. Um dos principais problemas descritos pelo autor é a falta de robustez do algoritmo em relação a variações de escala.

D. *MSER – Maximally Stable Extremal Regions*

Algoritmo proposto por Matas et al. [9] com o objetivo de desenvolver um método robusto diante de mudanças de perspectiva. Originalmente utilizado para detecção de características e alinhamento de imagens estéreo. O algoritmo localiza pontos extremos na imagem buscando identificar regiões conexas da imagem a partir da intensidade do brilho dos pixels. O algoritmo aplica limiares de diferentes valores e detecta regiões de bordas com grande variação de intensidade.

III. METODOLOGIA

O experimento foi realizado utilizando uma base de imagens pré segmentadas manualmente. A segmentação manual foi realizada com o objetivo de separar o objeto de interesse do restante da imagem. A separação é realizada através de um arquivo de máscara que demarca o local do objeto de interesse na imagem conforme mostrado na Fig. 1. Durante o processamento as imagens analisadas são multiplicadas pela negativa da máscara binária, resultando apenas na imagem do objeto de interesse. O experimento foi realizado com uma base de treinamento pré segmentada de 240 itens divididos em 15 categorias. Para a pesquisa foram utilizadas 3 imagens contendo objetos de cada categoria. Todas as imagens de pesquisa distintas das existentes no grupo de imagens de treinamento. Foram avaliados se as duas primeiras imagens retornadas do resultado da busca correspondiam a mesma classe do objeto pesquisado. Se os dois objetos retornados forem do mesmo tipo do objeto da imagem pesquisada, é marcado o valor 1. Se apenas um dos objetos for do mesmo tipo do objeto pesquisado é marcado o valor 0.5. Se nenhum dos dois objetos retornados forem do mesmo tipo do objeto pesquisado é informado o valor 0. Ao término de todas as avaliações todos os valores são somados. O conjunto de técnicas com a maior pontuação corresponde ao melhor desempenho na busca de imagens semelhantes. Para cada combinação, mostrada na Tabela I, foram executadas as etapas de treinamento e de busca por cada uma das imagens do grupo de pesquisa.

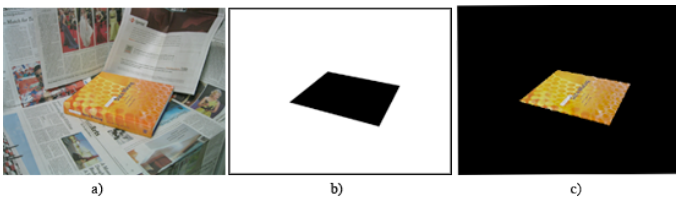

Fig. 1 - a) Imagem original b) Máscara c) Imagem segmentada

Tabela I - Combinações avaliadas.

Algoritmo de detecção de pontos chaves	Algoritmo de extração de descritores
SIFT	SIFT
FAST	SIFT
ORB	ORB
MSER	SIFT

IV. ALGORITMO PROPOSTO

O algoritmo proposto é composto por duas partes distintas. Na primeira ocorre o processamento de toda a base de treinamento. Esse processamento consiste nas etapas de aquisição, pré processamento, segmentação e extração de características. Na etapa de aquisição é realizada a leitura recursiva de um diretório contendo as imagens de treinamento especificado. Todas as imagens do diretório e subdiretórios são lidas. Na etapa de pré processamento são realizadas as conversões em tipo de cores exigidas para a execução das etapas seguintes. Na etapa de segmentação é realizada a leitura da imagem de máscara criada manualmente para cada uma das imagens da base e realizada a subtração da imagem sendo analisada. Essa subtração resulta em uma nova imagem contendo apenas o objeto de interesse. Na última etapa da parte de treinamento ocorre a extração dos pontos chaves e dos descritores da imagem de acordo com a técnica testada. O processamento de treinamento resulta em um vetor contendo a localização da imagem e os descritores extraídos desta imagem.

A segunda parte consiste na busca de imagens semelhantes a partir de uma imagem sendo pesquisada. Para efetuar a busca são realizadas novamente as etapas de pré processamento, segmentação e extração de descritores na imagem sendo pesquisada. É utilizada exatamente a mesma rotina usada na etapa de treinamento. O resultado é um vetor contendo o caminho da imagem pesquisada e os descritores extraídos. Após a extração dos descritores ocorre a comparação utilizando força bruta (*BruteForceMatcher* implementado no OpenCV) entre cada um dos descritores existentes no vetor de treinamento. Este método compara os descritores da imagem pesquisada com todos os descritores de todas as imagens armazenadas utilizando o algoritmo *k-nearest neighbour*. O resultado de processamento de cada comparação de força bruta é um vetor contendo os descritores compatíveis encontrados e suas respectivas distâncias. É calculada a distância mínima e máxima entre os descritores compatíveis e os descritores com distância mínima multiplicados por 2,2 são armazenados em um novo vetor como bons candidatos. O valor 2,2 foi obtido empiricamente durante os testes do experimento. O resultado final é um vetor ordenado por quantidade de descritores que são bons candidatos. As imagens com a maior quantidade de bons candidatos são mostradas como sendo semelhantes a imagem pesquisada conforme a Fig. 2.

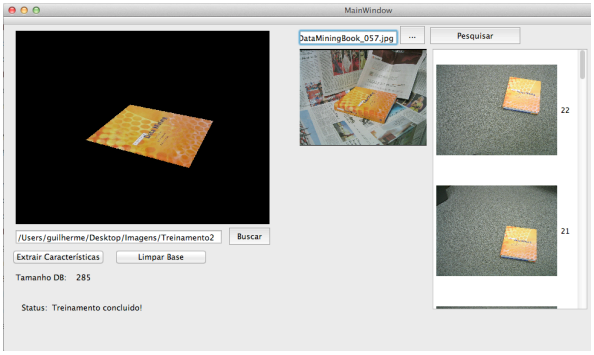

Fig. 2 - A esquerda, imagem pesquisada após subtração da máscara de segmentação. No centro, a imagem pesquisada. A direita, o resultado da busca.

V. EXPERIMENTO

Para desenvolvimento do programa foram utilizadas as seguintes tecnologias:

- Linguagem de programação: C++
- Bibliotecas: OpenCV 2.4.2 e QT 4.7.2

Foi utilizada a base de imagens do SIVAL[1] em conjunto com algumas imagens registradas pelo próprio autor do artigo. Todas as imagens foram segmentadas manualmente com o objetivo de separar o objeto de interesse do restante da imagem. Todas as imagens foram divididas em duas bases, uma chamada de treinamento e outra de verificação. A base de treinamento possui 240 imagens divididas em 15 categorias com fotos tiradas em locais, orientações e condições de iluminação diferentes para cada categoria. A base de verificações possui 3 amostras para cada uma das 15 categorias em orientações e condições de iluminação diferentes.

VI. RESULTADOS

Os resultados obtidos mostram um bom desempenho na identificação de imagens semelhantes utilizando a combinação das técnicas MSER para detecção de pontos chaves e SIFT para a extração de descritores. Este conjunto mostrou que apenas em um dos casos não foi retornada nenhuma imagem da mesma classe e em outras duas pesquisas foram retornadas apenas uma imagem da mesma classe. Em todas as outras pesquisas o algoritmo retornou corretamente duas imagens da mesma classe da imagem pesquisada.

A utilização dos algoritmos MSER para detecção de pontos chaves e o algoritmo SIFT para a extração de descritores mostrou o melhor desempenho na base de imagens analisada identificando imagens semelhantes com 91.1% de sucesso nas pesquisas realizadas. Em segundo lugar ficou a combinação dos algoritmos FAST e SIFT com 90% de acerto.

O pior desempenho ocorreu na utilização do ORB onde o resultado da análise somou apenas 33 pontos mostrado na

[1] http://accio.cse.wustl.edu/sg-accio/SIVAL.html

Tabela II, 19.5% a menos do que o melhor desempenho alcançado pela combinação MSER e SIFT.

Tabela II - Resultados com base na metodologia definida na Seção III

Classe	Imagem de teste	SIFT	FAST + SIFT	ORB	MSER + SIFT
1	1	1	1	1	1
	2	1	1	0	1
	3	1	1	0	1
2	1	0	0.5	0.5	0
	2	0	1	1	1
	3	0	0	1	0
3	1	1	0.5	1	1
	2	0	0	1	0
	3	0	1	1	1
4	1	0	1	1	0.5
	2	0	1	0.5	1
	3	0	1	0	1
5	1	1	1	0	1
	2	1	1	0.5	1
	3	1	1	0	1
6	1	1	1	1	1
	2	1	1	1	1
	3	1	1	1	1
7	1	1	1	1	1
	2	1	1	0	1
	3	1	0.5	1	1
8	1	1	1	1	1
	2	1	1	0.5	1
	3	1	1	1	1
9	1	1	1	0	1
	2	1	1	1	1
	3	0.5	1	1	1
10	1	1	0.5	1	0.5
	2	1	1	1	1
	3	1	1	1	1
11	1	1	1	1	1
	2	1	1	0.5	1
	3	1	1	1	1
12	1	1	1	1	1
	2	1	1	1	1
	3	1	1	1	1
13	1	1	0.5	1	1
	2	1	1	1	1
	3	1	1	0	1
14	1	1	1	0.5	1
	2	1	1	1	1
	3	1	1	0	1
15	1	1	1	1	1
	2	1	1	1	1
	3	1	1	1	1
Total:		**36.5**	**40.5**	**33**	**41**

VII. CONCLUSÃO

A utilização dos algoritmos MSER para detecção de pontos chaves e o algoritmo SIFT para a extração de descritores mostrou o melhor desempenho na base de imagens analisada. A utilização do SIFT tanto para extração de pontos chave quanto para extração dos descritores apresentou um bom desempenho, porém seu desempenho foi bastante prejudicado em imagens onde existe baixa variação de textura como nos exemplos contendo imagens de maçãs, bananas e de uma esponja. Nessas situações o algoritmo SIFT não conseguiu extrair um grande número de descritores resultando em um baixo desempenho na hora da pesquisa. A utilização do algoritmo FAST para extração dos pontos chaves conseguiu suprir a deficiência anterior em imagens com texturas pouco diferenciadas mas resultou em um desempenho ligeiramente menor em algumas outras imagens identificadas com sucesso somente pelo SIFT, resultando no segundo melhor desempenho. O pior resultado foi observado na utilização do algoritmo ORB, um dos fatores que podem ter influenciado nesse baixo desempenho é o fato do algoritmo ORB ser altamente sensível a escala conforme descrito pelos próprios autores [8].

REFERÊNCIAS

[1] Liu, T. Rosenberg, C. Rowley, H. A. (2007). "Clustering Billions of Images with Large Scale Nearest Neighbor Search". In: IEEE Workshop on Applications of Computer Vision.

[2] Jing, Y. Baluja, S. (2008). "PageRank for Product Image Search". In: Proceedings of the 17th international conference on World Wide Web.

[3] Rowley, H. Jing, Y. Baluja, S. (2006). "Large Scale Image-Based Adult-Content Filtering". In: Conf. on Computer Vision Theory & Applications.

[4] Tavares, A. (2011). Recuperação de imagens por conteúdo baseada em realimentação de relevância e classificador por floresta de caminhos ótimos. Tese de Doutorado. Universidade Estadual de Campinas.

[5] Mikolajczyk, K. et al. (2006) "A Comparison of Affine Region Detectors", In: International Journal of Computer Vision, Edited by Springer Science. Netherlands.

[6] Lowe, D. (2004). Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision. Volume 60. Issue 2. Pages 91-110.

[7] Rosten, E.; Drummond, T. (2006) "Machine learning for high-speed corner detection". In European Conference on Computer Vision.

[8] Rublee, E. et al (2011). "ORB: an efficient alternative to SIFT or SURF. In: IEEE International Conference on Computer Vision.

[9] MATAS, J. CHUM, O. URBAN, M. PAJDLA, T. (2004). Robust wide-baseline stereo from maximally stable extremal regions. Image and Vision Computing. Elsevier. Volume 22. Issue 10. Pages 761-767.