

Monitoramento Automatizado de Ambientes

Marcelo D. Lopes^{1,2}, Antonio C. Sobieranski¹, Eros Comunello¹, Aldo von Wangenheim¹

¹Laboratório de Processamento de Imagens e Computação Gráfica LAPIX,
Departamento de Informática e Estatística - Universidade Federal de Santa Catarina, sala 230
CEP 88040-900 - Florianópolis - SC - Brasil

²Ciência da Computação - Universidade do Vale do Itajaí
CEP 88122-000 São Jose - SC - Brasil

{marcelo,antonio,eros}@incod.ufsc.br, awangenheim@inf.ufsc.br

Abstract. *This paper presents the preliminary results of an automated computer vision system for the environment monitoring context. For this purpose, we used the OpenSURF method to extract local features of predefined patterns from static images, and stored into a database. The proposed approach was applied over video sequences captured sequentially from three cameras, and therefore the patterns identified are converted into objects of interest in a monitored environment.*

1. Introdução

O campo de vídeo monitoramento apresenta uma crescente demanda em aplicações automatizadas de segurança pública, tais como detecção de intrusão, detecção de estacionamento indevido de veículos e detecção de bagagem desacompanhada. Uma abordagem para aumentar a capacidade e o desempenho destes sistemas é a incorporação de técnicas para classificação de objetos, permitindo uma maior inferência durante a análise das cenas [Gurwicz et al. 2011].

O desenvolvimento das tecnologias envolvidas na construção de dispositivos para captura, processamento e armazenagem de imagens tem permitido a diminuição de seu custo, portanto a pesquisa para automação de sistemas de monitoramento justifica-se uma vez que a parte mais dispendiosa de todo o processo tornou-se manter recursos humanos para acompanhar as imagens [Collins et al. 2000].

O presente trabalho tem por objetivo desenvolver e implementar um sistema capaz de monitorar um ambiente a partir de N câmeras, onde um padrão previamente estabelecido possa ser identificado de forma automatizada. Com isto, o dispendioso trabalho de monitoramento em múltiplos monitores ou ambientes que é realizado manualmente pode ser automatizado, e um determinado objeto de interesse (e.g. estampa de uma camiseta, padrão objetos, bolsas, etc) pode ser encontrado computacionalmente e focado para uma câmera de interesse em um menor tempo. Esta etapa definiu quatro módulos que foram implementados e estão compreendidos entre aquisição, modelo, extração e interface.

2. Materiais e Métodos

O experimento foi executado empregando um computador com processador Intel Core i5, com 8 GB de memória RAM. Os dispositivos de captura utilizados foram três webcams Logitech QuickCam Orbit AF. Já para a programação foi utilizada a linguagem C++.

Algumas bibliotecas foram empregadas para a construção do protótipo. No módulo de captura foram utilizadas as bibliotecas OpenCV e libwebcam para capturar os frames e alterar os parâmetros de configuração das câmeras. No módulo interface foi empregado o framework Qt para a construção das telas. Para a criação e persistência dos modelos foi empregada a bibliotecas Boost. Já para a extração e comparação dos padrões foi utilizada a biblioteca OpenSURF, baseada no método para detecção e descrição de características locais em imagens apresentado por [Bay et al. 2008].

O protótipo implementado permite a criação de modelos que representam objetos específicos bem como a sua persistência em disco. Durante a análise é possível escolher um dos modelos salvos previamente para ser identificado, quando ocorre um reconhecimento positivo é pintado um quadrilátero ao redor do objeto de interesse como pode ser observado na Figura 1.

Como o sistema mostra a visualização de uma câmera por vez outra ação que ocorre na identificação positiva é a mudança de câmera em destaque, desta forma ocorre a troca automatizada das imagens em exibição na interface do usuário se o padrão de interesse for encontrado em uma câmera que não está em evidência.



Figure 1. Reconhecimento positivo

3. Considerações Finais

O protótipo desenvolvido apresenta um processamento sequencial no modo análise apresentando taxas de processamento entre 8 e 12 FPS (Frames per Second - Quadros por Segundo), o próximo passo no desenvolvimento irá fornecer um processamento concorrente para cada câmera, empregando o conceito de GPGPU (General Purpose Computing on Graphics Processing Units - Processamento de Propósito Geral em Processadores Gráficos).

References

- Bay, H., Ess, A., Tuytelaars, T., and Vangool, L. (2008). Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359.
- Collins, R., Lipton, A., Kanade, T., Fujiyoshi, H., Duggins, D., Tsin, Y., Tolliver, D., Enomoto, N., and Hasegawa, O. (2000). A system for video surveillance and monitoring. Technical Report CMU-RI-TR-00-12, Robotics Institute, Pittsburgh, PA.
- Gurwicz, Y., Yehezkel, R., and Lachover, B. (2011). Multiclass object classification for real-time video surveillance systems. *Pattern Recognition Letters*, 32(6):805 – 815.