

ACELERAÇÃO DE DESEMPENHO PARA SISTEMAS EMBARCADOS EM DRONES COM PROCESSAMENTO DE IMAGENS EM TEMPO REAL

Remo C. Carnevalli¹, Elcio H. Shiguemori¹, Lamartine N. F. Guimaraes¹

¹Instituto Estudos Avançados – IEAv
São José dos Campos - SP, Brasil - 12228-001

RESUMO

As aplicações com uso de Veículos Aéreos Não Tripulados (VANT) equipados com câmeras estão expandindo e exigindo cada vez mais respostas rápidas e apuradas para auxiliar nas tomadas de decisões. Este artigo aborda uma proposta de metodologia visando melhorar o desempenho de um sistema de processamento de imagens obtidas por VANTs visando-se aplicações que requerem processamento em tempo real. Recursos que utilizam processamento paralelo, redes neurais e alguns fatores de otimização para sistemas embarcados são apresentados visando-se a aceleração da aplicação e também extraindo-se a melhor performance oferecida pelo ambiente computacional. Determinar o cenário que se está trabalhando e encontrar fatores que possam contribuir com o desempenho final da aplicação tem relação com esta proposta. Os resultados apresentam os ganhos de desempenho no tratamento das imagens capturadas.

Palavras-chave — Vant, imagem, processamento, embarcado, desempenho.

ABSTRACT

Applications using Unmanned Aerial Vehicles (UAV) and equipped with cameras to capture images are expanding and increasingly demanding quick and accurate responses to assist in decision making. This article discusses a proposed methodology to be followed in order to improve the performance of an image processing system obtained by UAVs with real-time processing. Resources using parallel processing, neural networks and some optimization factors for embedded systems are presented aiming at application acceleration and also extracting the best performance offered by the computational environment. Determining the scenario that is being worked on and finding factors that may contribute to the final performance of the application is directly related to this proposal. The results section presents the performance gains in the treatment of the images obtained.

Key words — UAVs, image, processing, embedded, performance.

1. INTRODUÇÃO

O uso de sistemas de imageamentos embarcados em Satélites e Aeronaves têm aumentado nos últimos anos, por exemplo, a quantidade de satélites pode quintuplicar na próxima década [1] e o emprego de aeronaves não tripuladas também tem aumentado rapidamente. Nos últimos anos, aplicações nas áreas de sensoriamento remoto e visão computacional tem sido bem requisitadas. Em [2] está apresenta-se a necessidade do uso de veículos aéreos não tripulados na agricultura digital. Em [3] os drones são utilizados no monitoramento e diagnóstico de linhas de distribuição e transmissão na indústria de energia. Já em [4], a tecnologia de drones para processamento de imagem é aplicada ao longo de fronteiras, onde os custos de operação podem ser reduzidos e também reduzir o número de horas de patrulhamento.

As diversas aplicações em diferentes áreas têm contribuído para este aumento. Neste cenário, variados tipos de sensores de imageamento podem ser considerados, por exemplo, os embarcados em satélites e aeronaves (tripuladas ou não tripuladas) como os sensores ópticos, termais, hiperspectrais, Radar de Abertura Sintética e LIDAR [5]. Um desafio, então, é empregar e desenvolver tecnologias para processar este grande volume de dados para extração automática de informações, haja vista que em diversas situações, as análises são feitas manualmente ou com participação humana. Em [6] foi proposta uma rede de detecção de objetos em sistemas embarcados utilizando redes neurais convolucionais. Outros estudos concentram-se no uso de técnicas de visão computacional para detectar e categorizar objetos de tráfego em vários ambientes [7]. As técnicas de visão computacional em conjunto com redes neurais Perceptron de Múltiplas Camadas (MLP) têm sido usadas para pré-processar as imagens e facilitar a extração automática de informações, conforme apresentado em [8]. Com o avanço da tecnologia com a necessidade de processamento de grande volume de dados no menor tempo possível, soluções que envolvem alto desempenho computacional têm sido necessárias. Para abordar estes problemas, tecnologias têm sido combinadas e empregadas para este modelo de processamento, entre elas, Inteligência Artificial, Visão Computacional, Redes Neurais Artificiais e técnicas de Processamento Paralelo no intuito de explorar ao máximo a capacidade de processamento da arquitetura

escolhida, bem como, reduzir o tempo de espera no processamento.

2. MATERIAL E MÉTODOS

2.1. Processamento de imagem em tempo real

O emprego de tecnologias de alto desempenho computacional para processamento de dados de imageamento mostra-se como o *estado da arte* para extração automática de informações a partir de grandes quantidades de imagens no menor tempo possível [9]. Em [10] propõe-se o desenvolvimento de um sistema de visão computacional, que utiliza imagens capturadas e processadas em tempo de voo para determinar a localização de um VANT (Veículo Aéreo Não Tripulado). [11] propõe um método capaz de estimar a posição geográfica de um VANT durante um voo noturno. Em [12] utilizou-se redes neurais artificiais e técnicas de visão computacional para estimação da posição de um VANT. No entanto, pouco tem-se analisado e explorado o uso de tecnologias de alto desempenho computacional em sistemas embarcados com processamento em tempo real. O processamento de imagens com uso de técnicas de inteligência artificial tem sido feito em diferentes hardwares: *Desktops*, *Workstation*, *Clusters* e também em computadores de maior porte. Este mesmo modelo de processamento vem sendo transferido para computadores de menor porte conhecidos como SBC (single board computer) tais como: Raspberry PI, Odroid, Zybo, Jetson entre outros. Acelerar o desempenho computacional pode ser vital para uma operação embarcada e principalmente se estiver ocorrendo em voo autônomo.

2.2. Hardware de Desenvolvimento



Figura 1. Raspberry PI model B

Neste trabalho foi considerada a SBC Raspberry PI. Trata-se de uma arquitetura que segue o padrão *Reduced Instruction Set Computer* (RISC) com um número menor de instruções, resultando assim em custos menores de desenvolvimento, menor dissipação de calor e consumo energético, fatores estes fundamentais para serem adicionados em sistemas embarcados. Esta arquitetura utiliza um processador ARM Broadcom BCM2837 de 1.2GHz 64 bits Quad Core, GPU 400 MHz videocore IV, 1 GB de memória RAM, slot micro SD e fonte de alimentação 5V/3A. Ela foi escolhida porque pode ser facilmente embarcada em drones.

2.3. Cenário computacional e os fatores de otimização

Este artigo visa explorar a plataforma computacional SBC em um sistema embarcado para processamento de imagens em tempo real. O objetivo é avaliar cada setor e encontrar fatores que possam ser otimizados visando acelerar o desempenho computacional durante um voo autônomo. Estes fatores serão extraídos de cada elemento que compõe o cenário computacional embarcado em um VANT. Sendo assim, os setores são apresentados na Figura 2. A proposta da visualização do cenário computacional em camadas segue o modelo OSI [13] para rede de computadores, onde o processo de análise tem início pelas camadas mais baixas e segue a análise e otimização até o tratamento do código fonte e os dados coletados.

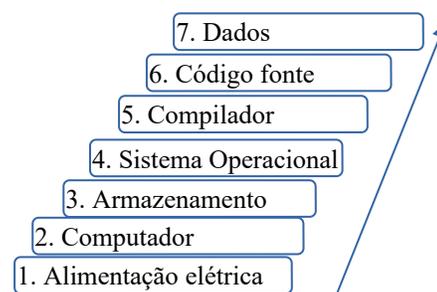


Figura 2. Cenário computacional - visão por camadas

Sabe-se que a relação entre cada setor ocorre de forma natural, portanto cada um deles pode contribuir com fatores de otimização sabendo que este fator contribui diretamente no desempenho dos demais camadas, colaborando assim para atingir a melhor desempenho computacional. Então para atingir a melhor performance pode se considerar que o melhor resultado é dependente da integração dos fatores otimizados, cuja análise é proposta neste artigo.

2.4. Implementação

A parte experimental teve início com a seleção do hardware e instalação e configuração do sistema operacional. Como a proposta foi de trabalhar com elementos de código aberto por permitir portabilidade e principalmente otimização, optou-se pelo Linux Ubuntu. Para estudo de caso, foi selecionado o algoritmo desenvolvido por [10] voltado no processamento de dados LIDAR. A metodologia aplicada utiliza fusão de dados por um filtro de partículas, visão computacional, odometria visual e registro automático de imagens. Estas técnicas empregadas por [10] passam a ser interessantes para o estudo de caso, pois a aplicação também está direcionada para ambientes embarcados e seu processamento exige bastante do processador local. As fases deste algoritmo são: captura das imagens, conversão da imagem multibanda para monocromático, filtragem da imagem para atenuação de ruídos, ajustes de escala e

rotação, extração de bordas através de redes neurais artificiais e correlação entre imagem captura com imagem armazenada. Este código foi desenvolvido em C/C++ e o compilador utilizado foi o *GNU Compiler Collection*. Com os elementos básicos já definidos, partiu-se para o monitoramento do tempo de execução do software e encontrar partes do código com processamento massivo, onde poderiam ser otimizados e implementados técnicas de processamento paralelo. Como o hardware em uso utiliza memória compartilhada, optou-se por explorar o paralelismo utilizando parâmetros do *OpenMP* (Open Multi-Processing). Com esta implementação, houve ganhos de aceleração, mas observou-se que era possível obter melhores resultados. A avaliação do percentual de ganho depende da necessidade da aplicação. Visando ampliar o ganho de desempenho, partiu-se para a análise do cenário computacional, com o objetivo de extrair novos fatores de otimização que pudessem contribuir na melhoria de desempenho. Neste sentido é de vital importância explorar fatores que influenciam diretamente no desempenho, como utilizar uma fonte de alimentação adequada, trabalhar com processadores de baixo consumo de energia, ajustar a frequência da CPU, escolher o cartão de memória adequado a múltiplas operações de I/O, otimizar o sistema operacional, implementar acesso direto a memória e acesso direto a disco, dividir processamento com GPGPU quando possível, evitar no código fonte chamadas de função e variáveis do tipo ponto flutuante (double) e por fim, definir o tamanho e tipo da imagem que será manipulada.

3. RESULTADOS

Os resultados apresentados neste trabalho utilizam um conjunto de imagens provenientes de voos reais obtidas por um VANT em tempo de voo. Importante ressaltar que aplicados no estudo de caso demonstrado abaixo ocorrem quatro tipos de processamentos diferentes, e a partir do segundo modelo de processamento passa a ser incrementado um fator de aceleração. Sendo assim, cada valor no gráfico representa o tempo de processamento em função dos fatores de aceleração escolhidos.

| | |
|-----------|--|
| Serial | Somente em CPU |
| Paralelo | CPU + Paralelismo |
| Híbrido | CPU + Paralelismo + GPU |
| Otimizado | CPU + Paralelismo + Otimização na Compilação + GPU |

Tabela 1. Modelos de processamento utilizados.

3.1 Estudo de Caso

São apresentados a aceleração de desempenho, com uso da metodologia proposta, em 3 fases fundamentais do algoritmo de registro automático de imagem, rotação da imagem, extração de bordas e um processo de correlação

entre imagens. Na tabela 2 são apresentados as especificações técnicas adotadas para a realização do estudo de caso. A Figura 3 traz exemplos das imagens coletadas e nas figuras 4, 5 e 6 são os resultados alcançados com a metodologia proposta.

| | |
|-------------------------------|-----------------------------|
| VANT: Phanton3 | Tamanho em Pixel: 360 x 288 |
| Sensor Ótico Termal: Flir Duo | Tamanho do arquivo: 103 KB |
| Quantidade de imagens: 1593 | Tipo do arquivo: PGM |
| Tipo da Imagem: Raster | SBC: Raspberry PI Model B |

Tabela 2. Quesitos do experimento



Figura 3. Exemplo das imagens capturadas.

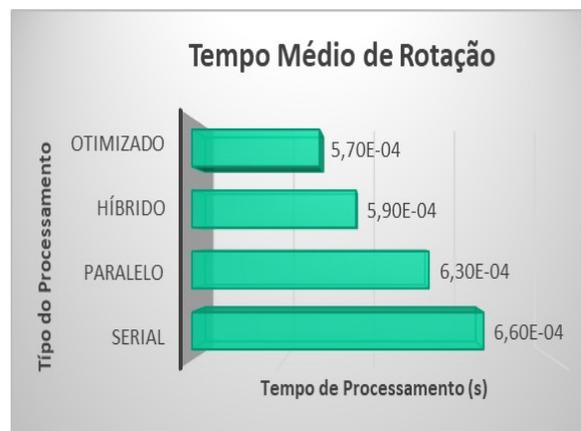


Figura 4. Ganho de desempenho – Rotação de imagens

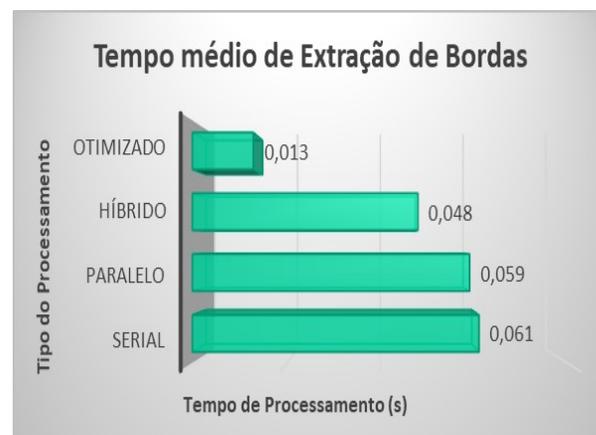


Figura 5. Ganho de desempenho – Extração de bordas .

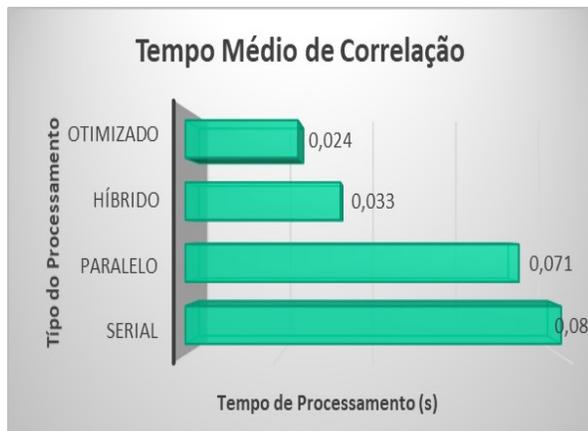


Figura 6. Ganho de desempenho - Correlação de imagens.

4. CONCLUSÕES

Mesmo considerando um ambiente computacional reduzido com as *Single Board Computer* baseadas nos processadores ARM destinados a sistemas embarcados onde ocorre uma série de restrições, os resultados mostraram-se promissores indicando que é possível aplicar técnicas para aceleração de desempenho onde ocorre processamento de imagem em tempo real. Com base nesta metodologia fica claro que não se pode atingir o melhor desempenho da aplicação sem que cada uma das camadas do cenário computacional atinja suas melhores performance. O estudo de caso apresentado indica que a somatória dos fatores de aceleração, impulsionadas pela aplicação do processamento paralelo, podem trazer resultados expressivos chegando a melhorias de desempenho na ordem de 80% em relação ao processamento inicial.

5. REFERÊNCIAS

[1] Ryan-Mosley T, Winick E., Kakaes K. - The number satellites orbiting Earth could quintuple in the next decade MIT Technology Review - Jun 26, 2019.

[2] Arantes, B.H.T.; Arante, L.T.; Costa, E. M.; Ventura M.V.A. - Drone Aplicado na Agricultura Digital - Ipê Agronomic Journal - v. 3 n. 1 (2019)

[3] N. Jung, M. Choi and C. Lim, "Development of Drone Operation System for Diagnosis of Transmission facilities," 2018 21st (ICEMS), Jeju, 2018, pp. 2817-2821.

[4] Mirzaeinia, A; Hassanalian, M.; Lee K. - Drones for Borders Surveillance: AIAA Scitech 2020 Forum.

[5] K. Lee and J. Sim, "Cloud Removal of Satellite Images Using Convolutional Neural Network With Reliable Cloudy Image Synthesis Model," 2019 IEEE International Conference on Image Processing (ICIP), Taipei, Taiwan, 2019, pp. 3581-3585.

[6] Y. Sun, C. Wang and L. Qu, "An Object Detection Network for Embedded System," 2019 IEEE International Conferences on Ubiquitous Computing & Communications, Shenyang, China, 2019, pp. 506-512.

[7] Ahmad, A., Anisetti, M., Damiani, E. *et al.* Special issue on real-time image and video processing in mobile embedded systems. *J Real-Time Image Proc* 16, 1–4 (2019).

[8] M. A. Anupama, V. Sowmya and K. P. Soman, "Breast Cancer Classification using Capsule Network with Preprocessed Histology Images," 2019 International Conference on Communication and Signal Processing (ICCSP), Chennai, India, 2019, pp. 0143-0147.

[9] Gulo, C. A. S. J. Técnicas de Computação de Alto Desempenho para o Processamento e Análise Eficiente de Imagens Complexas (Programa Doutoral em Engenharia Informática) Faculdade de Engenharia da Universidade do Porto, 2015.

[10] Braga, J. R. G. ; Shiguemori, E. H. ; Campos Velho, H.F. Determining the trajectory of Unmanned Aerial Vehicles by a novel approach for the particle filter. *Mecanica Computacional*, v. 36, p. 683, 2018.

[11] Silva, W. Navegação autônoma de VANT em período noturno com imagens infravermelho termal. Dissertação (Mestrado em Computação Aplicada) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2016.

[12] Goltz, G. A. M.; Shiguemori, E. H. ; Campos Velho, Haroldo F. . UAV Position Estimation By Image Processing Using Neural Networks. Fortaleza - CE: Sociedade Brasileira de Redes Neurais, 2011.

[13] Ricardo, Ademar Felipe Fey E. Raul. Introdução Às Redes De Computadores: Modelos Osi E Tcp/ip. Clube de Autores, 2015, pp.37-45.