

Estudo da Utilização de GPUs no Processamento de Imagens de Satélite

Study of GPUs in Satellite Image Processing

Victor Hugo de Moraes Danelichen^{*a}; Osvaldo Alves Pereira^a

^aUniversidade de Cuiabá, Programa de Pós-Graduação Stricto Sensu em Ciências Ambientais. MT, Brasil.

^{*}E-mail: victor.danelichen@kroton.com.br

Resumo

O uso e análise de inúmeras imagens de sensoriamento remoto têm demonstrado serem indispensáveis na constatação de mudanças na superfície terrestre. Em função do grande volume de dados e complexidade dos algoritmos, a análise de séries temporais tem sido uns dos problemas com maior necessidade de poder computacional na área de geotecnologias. Diante disso, o objetivo geral deste trabalho foi analisar diferentes estudos que utilizam GPUs (Graphic Processing Unit) para processar imagens de sensoriamento remoto com intuito de obter melhor desempenho. Para a elaboração deste trabalho foi necessário realizar consultas em artigos científicos e livros. A utilidade das GPUs nas principais áreas de sensoriamento remoto com uso de imagens de satélite (agricultura, biomas e mudança do uso e ocupação do solo) foi bem mais compreendida neste trabalho com a apresentação dos conhecimentos sobre sensoriamento remoto, geotecnologias e unidades de processamento de um hardware computacional. Este estudo demonstrou, também, que a utilização de GPUs surge como alternativa promissora na presente era tecnológica e avanço de futuros trabalhos que destinam entendimento das estruturas de núcleos gráficos de processamento com imagens de satélites, e que representam grande quantidade de dados “Big Data” em menor tempo.

Palavras-chave: Geotecnologia. Sensoriamento Remoto. Satélites. Unidades de Processamento.

Abstract

The use and analysis of numerous remote sensing images has proven to be essential in the observation of changes on the earth's surface. Due to the large volume of data and the complexity of the algorithms, the analysis of time series has been one of the problems with the greatest need for computational power in the field of geotechnology. Therefore, the general objective of this work was to analyze different studies that use GPUs (Graphic Processing Unit) to process remote sensing images to obtain better performance. For the elaboration of this work, it was necessary to consult scientific articles and books. The usefulness of GPUs in the main areas of remote sensing using satellite images (agriculture, biomes and change in land use and occupation) was better understood in this work with the presentation of knowledge about remote sensing, geotechnologies and processing units of computer hardware. This study also demonstrated that the use of GPUs emerges as a promising alternative in the present technological age and advances in future works aimed at understanding the structures of graphics processing cores with satellite images, and that represent a large amount of “Big Data” data in a smaller amount of time.

Keywords: Geotechnology. Remote sensing. Satellites. Processing Units.

1 Introdução

Desde a década de setenta com o lançamento da série Landsat (Land Remote Sensing Satellite) de satélites pela NASA (National Aeronautics and Space Administration) o estudo das mudanças da superfície terrestre, ecossistemas, florestas, crescimento de áreas urbanas e agrícolas vem chamando atenção da humanidade (SOARES *et al.*, 2020). O uso e análise de inúmeras imagens de sensoriamento remoto (satélites, drones, vants etc.) têm demonstrado serem indispensáveis na constatação dessas mudanças (BÉGUÉ *et al.*, 2018; DANELICHEN *et al.*, 2021).

Em função do grande volume de dados (imagens de satélites) e complexidade dos algoritmos, a análise de séries temporais (coleção de observações realizadas ao longo do tempo) tem sido uns dos problemas com maior necessidade

de poder computacional na área de geotecnologias e sensoriamento remoto (DANELICHEN *et al.*, 2020). No Brasil, por exemplo, o INMET (Instituto Nacional de Meteorologia) administra mais de 400 estações meteorológicas que geram milhões de dados diários, sem contabilizar estações automáticas, privadas, espalhadas pelo país e dados espaciais pelo INPE (Instituto Nacional de Pesquisas Espaciais) em parceria com a China (CARDIM *et al.*, 2018).

A capacidade de processamento de grande número de imagens de satélite com extenso armazenamento, nos últimos anos, vem ganhando avanço com a utilização geral das unidades de processamento gráficas (GPUs) em contraste com as unidades centrais de processamento (CPUs). Os processadores das GPUs são disponíveis pelas indústrias com vários núcleos baseados em uma arquitetura multicore que oferece ferramentas de paralelismo, o que proporciona

processamento mais veloz e ganho de desempenho (GOMES *et al.*, 2021).

Nesse contexto, a computação tem uma grande contribuição e desafio, a sanar o seguinte questionamento: o aumento na velocidade de cômputo dos processadores está ligada ao avanço da capacidade de processamento? A indústria comumente vem tentando resolver este problema aumentando grande número, dentro dos processadores de transistores (NVIDIA, 2021). Tarefa essa difícil, em função do excessivo consumo de energia, limitação física do silício (elemento indispensável nas placas eletrônicas) e o alto aquecimento das placas de processamento.

Diante disso, o objetivo geral deste trabalho foi analisar o emprego de diferentes estudos que utilizam GPUs (Graphic Processing Unit) para o processamento de dados espaciais em sensoriamento remoto. Para alcance deste objetivo, os seguintes objetivos específicos foram propostos: apresentar um breve overview de Sensoriamento Remoto; demonstrar a utilização das GPUs em Sensoriamento Remoto; e analisar o emprego das GPUs em diferentes contextos de aplicação

2 Desenvolvimento

2.1 Metodologia

Para a elaboração deste trabalho foi necessário realizar consultas em artigos científicos e livros, bem como pesquisa documental em jornais e revistas especializadas em tecnologia e processamento paralelo no banco de dados de pesquisa Google acadêmico. As palavras-chave utilizadas na busca foram: “GPU”, “sensoriamento remoto por processamento paralelo” e “imagens de satélite aplicada a computação”.

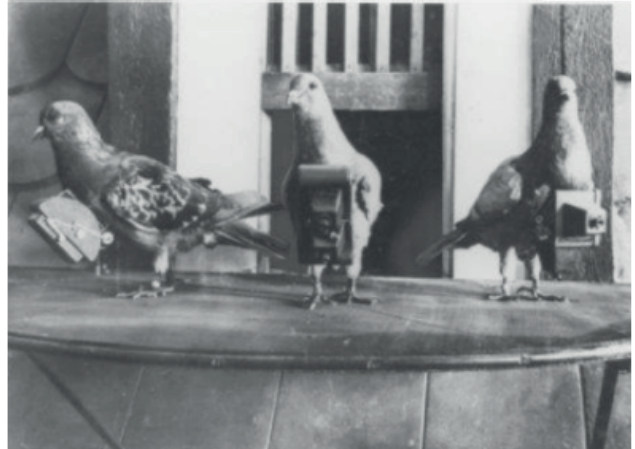
2.2 Sensoriamento Remoto

Vinculado ao uso militar, a história do sensoriamento remoto (SR) se inicia com o avanço tecnológico bélico mundial (FLORENZANO, 2011). O termo teve origem nos anos 1960, por Evelyn L. Pruit e colaboradores, que o definiram como sendo umas das mais promissoras tecnologias desenvolvidas para levantamento e monitoramento da dinâmica da superfície terrestre em escalas macros, regionais e globais (MENESES, 2012).

Tal tecnologia, que define o SR, permite aquisição de informações da superfície terrestre por meio da captação e registro da energia emitida ou refletida por essa. Essa obtenção de dados (espaciais) ocorre através da detecção e mensuração quantitativa das interações dos objetos na superfície terrestre com a radiação eletromagnética (DANELICHEN *et al.*, 2020).

Advindo da invenção da câmera fotográfica, o SR consta com diversos sensores de veículos tripulados e não tripulados. A câmera fotográfica é um bom exemplo de um sensor remoto, que surgiu com o uso de pombos correios (veículos) que eram treinados e levados a sobrevoarem a locais escolhidos determinados, tendo assim os primeiros registros de imagens aéreas (Figura 1).

Figura 1 - Pombos da Bavária (1903) com câmeras fotográficas em seu peito para sobrevoos de reconhecimento da área



Fonte: Guedes e Silva (2018).

O SR passou por grandes avanços desde o surgimento das câmeras fotográficas, porém seu auge foi na década de 1970, com o lançamento dos sistemas de satélites Landsat (Land Remote Sensing Satellite) pela NASA (National Aeronautics and Space Administration) em 23 de julho de 1972 (REUSE *et al.*, 1973). Nesse mesmo ano, a missão Apollo 17 obteve a primeira imagem colorida e em alta resolução da Terra por sensor orbital (Figura 2).

Figura 2 - The Blue Marble “A bolita azul”



Fonte: Reuse *et al.* (1973).

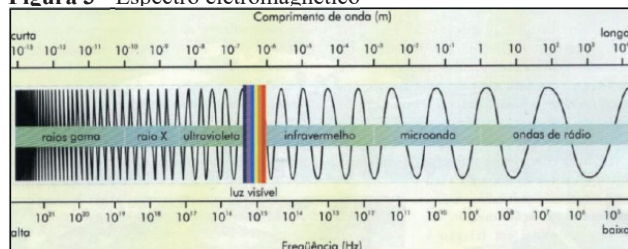
Diante disso, o avanço do SR é resultado de um enorme esforço conjunto da física, química, biologia e outras ciências como a computação. Além da multidisciplinaridade do SR, ainda é necessário um grande número de profissionais para desenvolvimento e operação contínua e observação do Planeta Terra com o objetivo de fornecer dados importantes para tomadas de decisão, pesquisas e estudos envolvendo sensores orbitais (DANELICHEN *et al.*, 2021).

O requisito chave para a interpretação dos dados coletados via SR, por diferentes sensores, é o entendimento da interação entre o objeto na superfície terrestre (solos, vegetação, água,

áreas urbanas, florestas etc.) e a radiação eletromagnética (REM). A REM tem, em sua natureza, um comportamento dual de onda e energia (ALVARENGA *et al.*, 2003). Sendo essa uma onda eletromagnética e uma partícula que se propaga pelo espaço vazio, atingindo o objeto, havendo uma interação, ou processo vice-versa, no qual o objeto emite também energia para o espaço (DANELICHEN *et al.*, 2020).

Os sensores a bordo dos satélites ou veículos terrestres de sistemas de SR recebem informações das feições da superfície e suas características por meio da REM. Nesse sentido se tem um “gráfico” para representar o comportamento da REM em diferentes comprimentos de onda (bandas em SR), que demonstra a distribuição da REM com base nas frequências e comprimento de onda denominado “espectro eletromagnético” (Figura 3).

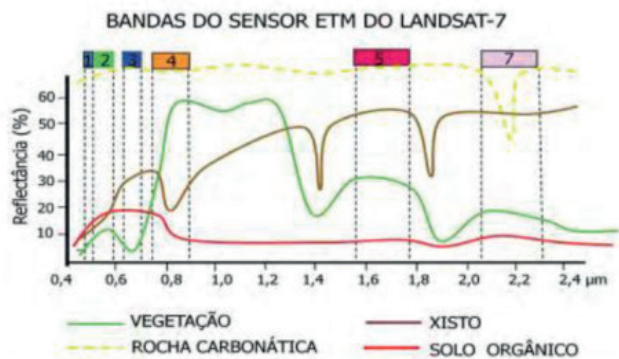
Figura 3 - Espectro eletromagnético



Fonte: Florenzano (2011).

De acordo com a Lei de Wien, toda matéria acima do zero absoluto (0 K) emite REM (CARDIM, 2019). Como cada objeto da superfície emite e absorve uma variação distinta da REM, isso possibilita o sensor remoto distinguir as feições e atributos do mesmo objeto imageado (JENSEN, 2009). Para essa característica intrínseca de cada objeto interagir com a REM se aplica o nome de “assinatura espectral”, ou seja, cada alvo absorve ou reflete, de modo diferente, cada faixa do espectro eletromagnético (Figura 4).

Figura 4 - Espectros de reflectância e as posições das bandas do sensor EMT Landsat 7 como forma para avaliar as bandas que melhor diferenciam espectralmente os quatro tipos de materiais

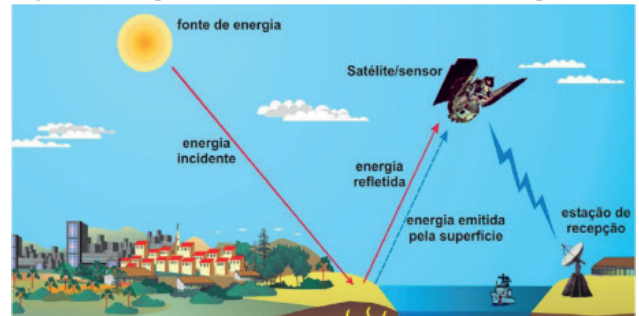


Fonte: Meneses (2012).

Para obtenção e interpretação da assinatura espectral de cada objeto da superfície terrestre é necessário o uso dos “sensores remotos”, que são instrumentos com capacidade de detectar a REM em diferentes bandas (faixas de comprimento de onda) do espectro eletromagnético (FLORENZANO,

2011). Os sensores remotos captam, convertem, armazenam e, dependendo do sensor, podem até processar produtos (matemática de bandas, como índices de vegetação) por meio da interação da REM. A Figura 5 mostra o esquema simplificado de funcionamento de um satélite passivo (necessita da luz do Sol como fonte para obter imagem).

Figura 5 - Esquema de funcionamento de um satélite passivo



Fonte: Florenzano (2011).

Fornecendo como produto uma imagem captada, como as câmeras fotográficas e os scanners, estes são classificados como sistemas imageadores, enquanto sistemas não-imageadores, como espectrorradiômetros ou radiômetros, geram como resultado produtos em formato de gráficos ou dígitos, esses sistemas são divididos em sensores passivos e ativos (GUEDES; SILVA, 2018).

Os sensores ativos são aqueles que não dependem de uma fonte de radiação externa para geração de imagens, pois produzem a sua própria radiação. Por outro lado, os sensores passivos são aqueles que necessitam de fonte externa para produção de imagens, como o Sol, por exemplo (OLIVEIRA, 2019).

Em função da capacidade de cada sensor gerar imagens detalhadas e performance de qualidade os sensores possuem resoluções espacial, espectral e radiométrica. A resolução espacial demonstra o poder de distinção de objetos na superfície, a exemplo do sensor Thematic Mapper do satélite Landsat 5 com 30 m de resolução espacial, tamanho do pixel 30x30m (Figura 6).

Figura 6 - Da esquerda para direita imagens do satélite Landsat 5 com resolução espacial de 30 m, Sport com 10 m e Ikonos com 1 m, de uma área do lago Paranoá de Brasília

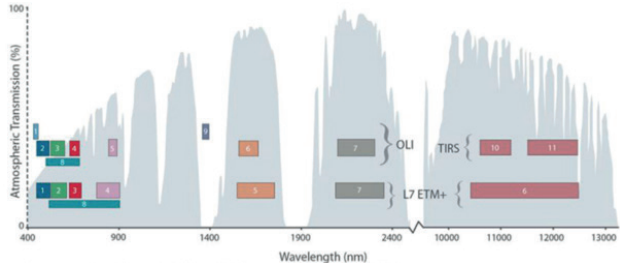


Fonte: Meneses (2012).

A resolução espectral compreende a quantidade de bandas espectrais que o sensor consegue captar, ou seja, quanto maiores forem os números de bandas que o sensor possui, melhor será sua resolução espectral da coleta (Figura 7). A

Figura 7 mostra os diferentes intervalos de comprimento de onda dos satélites Landsat 7 e 8. Dessa forma, é possível compor as composições de cores, como RGB (Red-Green-Blue) que gera a imagem colorida para cada sensor dos dois satélites.

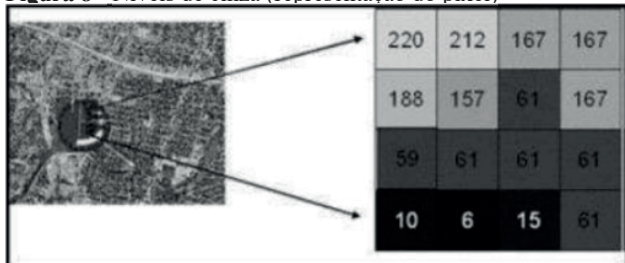
Figura 7 - Comprimentos de onda do Landsat 7 (posição inferior) e do Landsat 8 (posição superior)



Fonte: Guedes e Silva (2018).

De outro modo, se tem a resolução radiométrica que define a performance do sensor em detectar pequenos sinais e registros de diferenças na energia refletida ou emitida por cada componente na superfície terrestre (solos, rochas, água, vegetações etc.). Na prática, essa detecção é expressa em níveis de cinza (números digitais) usados para indicar valores da radiância, quantização de bits na linguagem da engenharia da computação (Figura 8).

Figura 8 - Níveis de cinza (representação do pixel)



Fonte: Crosta (1993).

Assim, quanto maior o número de “bits” detectados, maior será a qualidade da imagem. A exemplo do sensor TM do Landsat 5, que consegue distinguir até 256 tons diferentes de cinza, enquanto outros sensores com alta resolução, como do satélite Ikonos, possuem resoluções radiométricas de 10 a 11 bits, 1024 ou 2048 valores digitais (Figura 8).

Diante desse grande número de dados espaciais possibilitou aplicações de processamento digital de imagens (PDI). O uso de PDI permite que imagens de satélites obtidas em órbita terrestre sejam analisadas e relacionadas com pesquisas científicas, produção de alimentos, aquecimento global, previsão do tempo, defesa nacional, cartografia e outras (CHENG *et al.*, 2017).

Vale destacar que as imagens de satélite possuem alta resolução espacial e radiométrica com grandes dimensões, que impactam diretamente na capacidade de processamento requerendo do PDI inúmeras operações para alcançar resultados satisfatórios. Esses resultados são importantes para correta acurácia das previsões climáticas, produção de

alimentos, prevenção de desastres e, conseqüentemente, das tomadas de decisões dos poderes públicos relacionadas ao bem-estar da população (BÉGUÉ, 2018; OLIVEIRA, 2019).

2.3 GPU em Sensoriamento Remoto

A superfície do Planeta Terra está mudando rapidamente a uma taxa sem precedentes, a conversão dos ecossistemas em áreas antropizadas e agrícolas se expande em todo o espaço natural. O uso de imagens de satélite e o processamento de séries temporais de sensoriamento remoto para a classificação e tipo de uso e cobertura do solo tem mostrado ser indispensável na detecção de mudanças, o que tem chamado atenção do Mundo no contexto de sustentabilidade (OLIVEIRA, 2019).

Em conjunto com sensoriamento remoto, que pode ser definida como técnica que obtém informações dos objetos da superfície terrestre sem contato físico entre o sensor e o alvo (KUENZER *et al.*, 2015), o avanço tecnológico dos sistemas computacionais tem sido veloz e de grande proporção. A crescente capacidade computacional de processamento, armazenamento e transmissão de dados, tecnologias móveis, e a utilização de sistemas e padrões abertos, trazem significativas mudanças para a tecnologia da informação.

Nessa crescente corrida tecnológica, a área de computação paralela, que antes se estabelecia apenas em grandes centros de pesquisas e universidades, está disponível para usuários domésticos. Essa apresenta, em tese, soluções viáveis para o processamento de grande quantidade de imagens de sensoriamento remoto por meio de ambientes de processamento paralelo (CARDIM, 2019).

Em diversas áreas da indústria em geral e pesquisa, a necessidade de maior velocidade no processamento é exigida. Áreas como computação gráfica, computação científica, análise e processamento de imagem, modelagem de grandes quantidades de dados, previsão do tempo e muitas outras requerem resoluções de problemas relacionadas ao poder de processamento (PAGE, 2009; GULO, 2014).

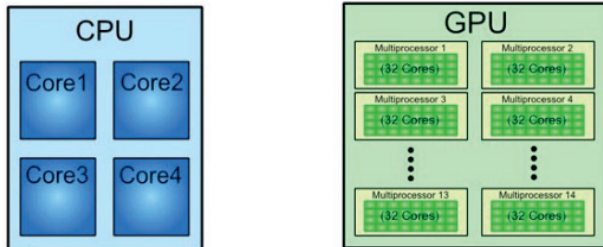
Problemas dessa magnitude podem ser encarados com a utilização da computação paralela. Utilizadas em processamento de alto desempenho, através de técnicas de paralelização, as aplicações da computação paralela são inseridas com o objetivo de estruturar decomposições de algoritmos ou dados em partes menores. Chamadas de tarefas, essas partes menores são distribuídas para serem executadas em núcleos simultaneamente ou em processadores (GEBALI, 2011).

Entre os processadores envolvidos são realizadas estas etapas de comunicação e coordenação geral das tarefas. Dois fatores são levados em consideração ao utilizar a programação paralela: o tipo de comunicação dos processadores e o tipo de arquitetura paralela.

O processamento de imagens de satélite depende, geralmente, de alto custo computacional, o que torna importante a utilização de arquiteturas paralelas para computar grande volume de dados e suas análises. Ao invés de

utilizar apenas um núcleo de processamento, as arquiteturas paralelas são majoritariamente formadas por vários núcleos de processadores. Pode-se citar duas principais abordagens paralelas (Figura 9). A multicore (CPUs), integra vários núcleos em um único microprocessador, e a abordagem manycore (GPUs), utiliza aceleradores com até milhares de núcleos (BÉGUÉ *et al.*, 2018).

Figura 9 - Arquitetura da CPU e GPU



Fonte: Aprenda Fazer (2021).

Os algoritmos de processamento digital de imagens, geralmente, necessitam de uma alta demanda computacional. Assim, com o objetivo de reduzir o tempo de processamento, pesquisadores estão se desdobrando com afinco em utilizar o máximo de recursos de estruturas de hardware disponíveis. No entanto, os processadores que existem no mercado vêm apresentando dificuldades em executar o processamento digital de imagens em tempo real, tendo em vista que a quantidade de dados e volume de informações presentes, em uma imagem de satélite, com aumento das resoluções espacial, espectral e radiométrica (ALALI *et al.*, 2013; DANELICHEN *et al.*, 2020).

Assertivamente, Danelichen *et al.* (2020) afirmam que as aplicações de sensoriamento remoto em processamento digital de imagens necessitam de uma alta demanda computacional, que inclui o processamento de hardware dedicado. À medida que aumentam as resoluções espaciais, temporais, radiométricas e espectrais na obtenção da imagem orbital em sensoriamento remoto, consequentemente, aumenta a quantidade de dados a serem computados, necessitando assim soluções de processamento, incluindo hardware.

De outra mão, a ser considerado de grande importância, é volumosa a demanda de processamento de imagens multiespectral, as quais possuem grande quantidade de bandas espectrais (resolução espectral). Nisso, com grande prioridade nas pesquisas envolvendo sensoriamento remoto tem focado no desenvolvimento de metodologias para análise de grandes dimensionalidades das imagens de satélite com maior precisão (CHENG *et al.*, 2017).

Diante disso, a inserção por meio de GPU (plataforma programável), de fácil integração com programas (software) e dedicada ao processamento paralelo, se torna indispensável. A plataforma GPU, por possuir a capacidade de execução de tarefas de modo paralelo, possui um alto poder computacional comparada ao processador de propósito geral (CARDIM *et al.*, 2018).

Em função de sua estrutura interna projetada para execução de tarefas de modo paralelo possibilita a GPU ter alto poder de processamento (Figura 10). De modo que a CPU possui apenas uma única unidade de controle complexa, uma única memória cache e pouca quantidade de processamento lógica e aritmética (ALU – Arithmetic Logic Unit), dividida em diversos blocos, a plataforma GPU possui sua unidade própria de controle simplificada e memória cache, além de múltiplas unidades ALU (CARDIM *et al.*, 2019).

Figura 10 - Exemplo físico de uma CPU e GPU



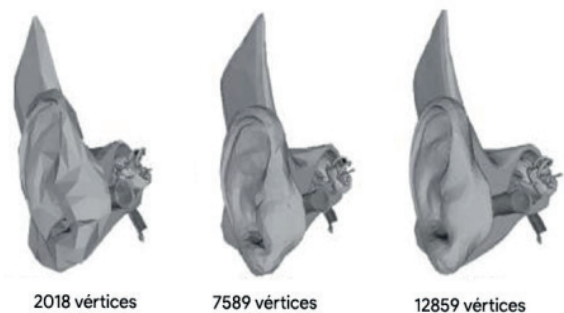
CPU

GPU

Fonte: GeekBoots (2021).

Muito utilizada em pesquisas de processamento digital de imagens, a plataforma GPU possui características que permitem que um mesmo processamento seja executado em vários pixels da imagem ao mesmo tempo, de modo paralelo (Figura 11). Assim, reduzindo o tempo de processamento das imagens em lotes de múltiplas imagens, o que não seria tão viável ou possível com a CPU.

Figura 11 - Processamento de vértices



Fonte: Aprenda Fazer (2021).

Algumas plataformas, atualmente, reúnem em um único sistema uma ou mais CPUs e uma ou mais GPUs, classificando como sistemas heterogêneos. A alta capacidade de processamento massivo de forma paralela e ótimo desempenho em cálculos para grande quantidade de dados são as principais características de GPUs (Graphics Processing Unit) que resulta em uma alta vazão (CHAPMAN *et al.*, 2010). De forma distinta do início, as GPUs hoje, Unidades de Processamento de Propósito Geral (GPGPU) atuam como coprocessadores que exploram o paralelismo massivo.

Nos primórdios dos sistemas que continham CPUs e GPUs eram utilizadas linguagens como Brook e Cg (MARK *et al.*,

2003). A fim de facilitar a interação programador e interface nas GPUs, a NVIDIA apresentou o CUDA (Compute Unified Device Architecture) em 2007 como linguagem do padrão OpenCL (Open Computing Language) e modelo comum para a programação heterogênea nas CPUs e GPUs (MUNSHI, 2009; SANDERS, 2010).

2.4 Aplicação de GPUs em Sensoriamento Remoto

Para a geração de excelentes resultados em sensoriamento remoto se deve preocupar com a geração de código fonte (PINHEIROS, 2017). O analista de dados espaciais e engenheiros de geoprocessamento devem estar atentos para a melhor maneira de o hardware executar com precisão seu código fonte, e o fator tempo de processamento é crucial neste processo.

Para diminuir o tempo de processamento, a etapa de otimização do código fonte entra em cena. Nessa etapa, os analistas buscam executar seus códigos da melhor maneira em seu hardware, em que utilizam algumas técnicas de otimização, entre essas se destaca o paralelismo. Esse processo pode ser feito também utilizando a melhoria dos trechos de códigos buscando alternativas de linguagens de alto nível como Python ou até mesmo modificando os compiladores, porém essa alternativa tem limitações e, na maioria das vezes, não alcançam resultados satisfatórios como o paralelismo.

De forma geral, um programa é escrito de forma sequencial e executado pela CPU, o que faz com que o tempo necessário para execução do código seja a somatória dos tempos gastos de cada instrução dentro do processador. Para que haja uma melhoria considerável, nesse tempo de processamento, a velocidade de processamento pode ser aumentada utilizando o conceito de processamento paralelo (paralelismo) que vem a ser a divisão de tarefas, de forma independente, a executar cada etapa em distintas unidades de processamento gráfico (GPU) ao mesmo tempo.

Passando a superar o desempenho das CPUs, as GPUs passaram a ser alternativas mais viáveis pela comunidade de sensoriamento remoto não apenas para usos gerais em processamento gráficos, mas também pela grande quantidade de dados (pixels) que uma imagem de satélite possui. Assim, a utilização no processamento de imagens de satélites começou a ser incorporada.

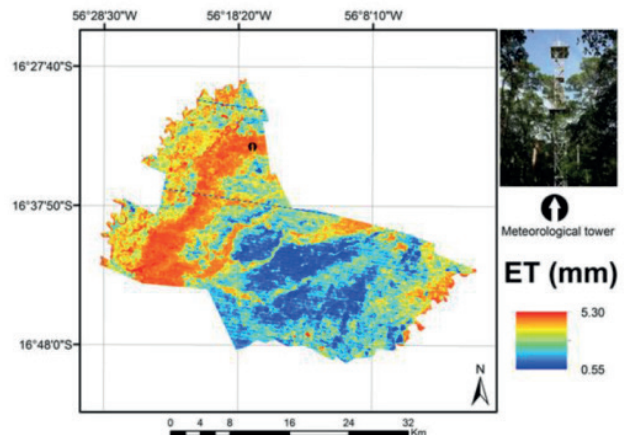
Nesse contexto, no ano de 2007 a empresa NVIDIA introduz um modelo de arquitetura em seus aceleradores gráficos denominado CUDA (Compute Unified Device Architecture) que passa a incluir componentes especificamente projetados para computação em GPUs o que facilitou em muito a programação híbrida entre GPUs e CPUs (PINHEIROS, 2017).

Diante disso, inúmeros estudos utilizando GPUs com dados espaciais para processamento digital de imagens em SR vêm sendo desenvolvidos, buscando a otimização do geoprocessamento com uso de imagens de satélite.

Gomes et al. (2021), com intuito de estudarem a

evapotranspiração (ET) no bioma Pantanal, em uma unidade de conservação (RPPN) via SR, utilizaram o algoritmo SEBAL (Surface Energy Balance Algorithm for Land) para quantificar índices de vegetação (NDVI, SAVI e LAI) e a ET com imagens do satélite Landsat 5, obtidas do Instituto Nacional de Pesquisas Espaciais INPE (<http://www.dgi.inpe.br>) (Figura 12).

Figura 12 - Evapotranspiração da unidade de conservação RPPN SESC Pantanal no dia 04 de setembro de 2008



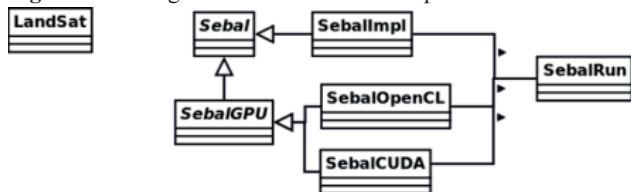
Fonte: Gomes et al. (2021).

Os autores utilizaram duas GPUs: NVIDIA 520m com 48 cores de 1.5 GHz e GPU NVIDIA GeForce GTX 560 com 384 cores de 1.5 GHz. O framework foi desenvolvido a fim de fornecer três recursos distintos para o desenvolvedor usar para o cálculo do SEBAL. Um para a linguagem Java pura, outro usando CUDA e outro para OpenCL. Tudo baseado nas imagens de satélite Landsat 5. A estrutura foi dividida em 3 partes distintas:

1. Pré-processamento: é o componente no qual as imagens Landsat são processadas para gerar a informação SEBAL inicial.
2. Cálculo dos pixels “quentes” e “frios”: esta é a parte em que você pode calcular os pixels que servirão de entrada para a determinação do balanço energético.
3. Processamento: é o componente que recebe todas as saídas das etapas 1 e 2 e realiza todos os cálculos SEBAL com 3 linguagens Java, CUDA e OpenCL.

A Figura 13 mostra o diagrama de classes que representa a estrutura. A classe LandSat, em que ocorre todo o pré-processamento das imagens com a saída de NDVI, SAVI, LAI, albedo, balanço de radiação, emissividade, temperatura de superfície e radiação de onda longa. A classe abstrata Sebal possui um método para calcular os coeficientes dos pixels “quentes” e “frios”. E as classes SebalImpl, SebalCUDA e SebalOpenCL representam o cálculo SEBAL usando Java, CUDA e OpenCL, respectivamente. Junto com essas classes, foi desenvolvida uma classe chamada SebalRun, que tem a função de executar todos os procedimentos necessários para a execução do SEBAL recebendo apenas as imagens de entrada e o tipo de execução que o desenvolvedor deseja, por exemplo, Java, CUDA ou OpenCL.

Figura 13 - Diagrama da estrutura de computo do SEBAL

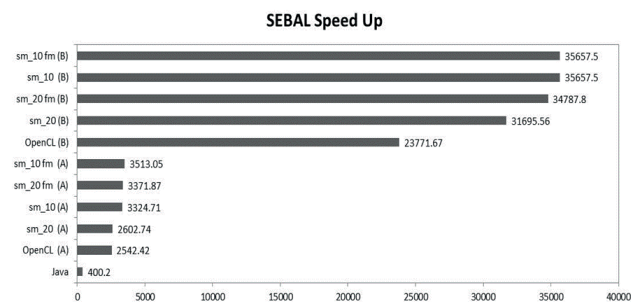


Fonte: Gomes *et al.* (2021).

A avaliação da ET estimada pelo SEBAL via função do algoritmo de processamento de imagem do software ModelMaker ERDAS 9.2 implementado na GPU JAVA foi realizada utilizando alguns indicadores: acurácia - índice de Willmott “d”; raiz do erro quadrático médio “RMSE” e o erro médio absoluto “EMA”. A precisão está relacionada ao afastamento dos valores estimados daqueles observados (software ERDAS).

O melhor desempenho para processamento do SEBAL no software ERDAS foi obtido com o CUDA com opção fastmath (fm) e 10 multiprocessadores stream (SM), apresentando um aumento de aproximadamente 36.000 vezes (Figura 14). A opção fastmath teve mais efeito a bordo com 48 núcleos do que a configuração de multiprocessador de fluxo. Além disso, a opção de multiprocessador stream teve mais efeito na placa com 384 núcleos do que a configuração fastmath. Vale ressaltar que a opção fastmath na compilação do CUDA faz com que a GPU use implementações otimizadas para algumas funções, como log e dividir (NVIDIA, 2011).

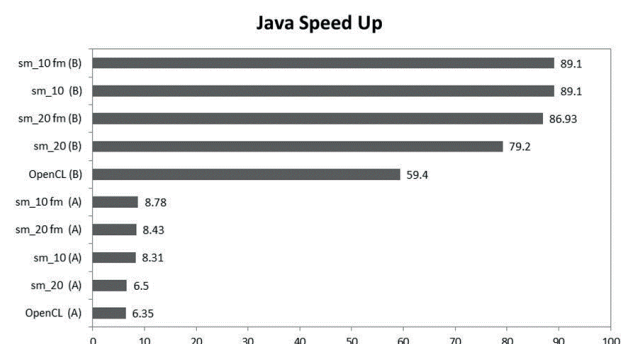
Figura 14 - Aceleração em relação ao software ERDAS



Fonte: Gomes *et al.* (2021).

O tempo de resposta foi reduzido de aproximadamente 7 para 90 vezes usando o código Java (Figura 15). O melhor desempenho de processamento SEBAL em relação ao implementado em Java foi obtido pelo CUDA com a opção fastmath (fm) e 10 stream multiprocessadores (SM). A opção fastmath teve mais efeito na placa com 48 núcleos do que a configuração do multiprocessador stream. Além disso, a opção de multiprocessador stream teve mais efeito na placa de 384 núcleos do que a configuração fastmath.

Figura 15 - Aceleração em relação ao código Java



Fonte: Gomes *et al.* (2021).

Os autores criaram três implementações do algoritmo SEBAL usando Java, CUDA e OpenCL. Essas implementações foram testadas usando um conjunto de dados consistentes e os códigos foram cronometrados. Na melhor das hipóteses, foi possível atingir uma velocidade de cerca de 36.000 (90) vezes (passando de 8 horas de trabalho para 2 minutos, incluindo o tempo necessário para ler os dados de entrada do disco rígido e armazená-los de volta). Essa aceleração foi alcançada usando apenas uma placa habilitada para CUDA. Para acelerar ainda mais o programa se pode usar várias placas NVIDIA e em outras plataformas usando OpenCL. Os valores diários de ET estimados pelas linguagens Java, CUDA e OpenCL foram semelhantes aos da ET estimada pelo algoritmo SEBAL no software ERDAS.

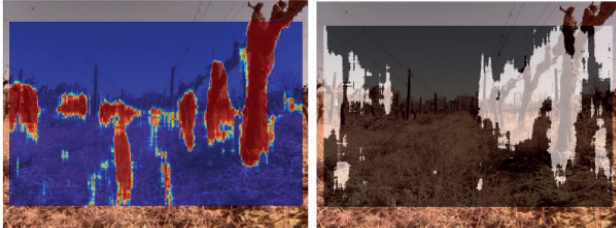
Outro estudo realizado por Azevedo (2018), visando automatizar a agricultura de vinha de encosta em uma plantação de videira em Portugal, em função da irregularidades do terreno e erros na agricultura de precisão com o uso do GNSS (Global Navigation Satellite System) implementou o algoritmo ViTruDe (Vine Trunks and Masts Detector) em GPUs para detecção de elementos naturais nas vinhas, nomeadamente os mastros e troncos, para auxílio à localização do AgRob V16, uma plataforma robótica que está sendo desenvolvida de forma modular para a execução de tarefas relacionadas com agricultura de encosta, como monitorização, pulverização, poda e colheita.

O algoritmo ViTruDe foi analisado utilizando um sistema operacional Ubuntu 16.04 LTS (64-bit) e processador (CPU) Intel Core TM i7-4700HQ, para duas diferentes GPUS: NVIDIA GeForce GT 740M e NVIDIA GeForce GTX1050 TI e separado em segmentos menores, que pudessem ser distribuídos por diferentes núcleos de uma GPU. Isto permitiu melhorar a velocidade da computação e assim contribuir para um melhor desempenho em tempo real da plataforma robótica. Os resultados obtidos por Azevedo (2018), relativos ao tempo de processamento, demonstraram que esta implementação em GPU permitiu alcançar uma solução cerca de 80 vezes mais eficiente do que a abordagem original em CPU.

Na figura 16 é feita a comparação de resultados da implementação do ViTruDe em CPU e para GPU. A imagem

utilizada para estas duas execuções são captadas nas videiras da Universidade de Trás-os-Montes e Alto Douro na cidade de Vila Real, em Portugal, com uma câmara sem infravermelhos nem filtro azul. Observa-se que em ambas as implementações, para a configuração do ViTruDe, em que é utilizado um descritor do tipo plus colour e SVM de núcleo linear com duas classes. Os pixels em vermelho, na ilustração da implementação em CPU, estão os pixels em que é provável a existência de troncos, enquanto, na ilustração em GPU, esses pixels estão a branco.

Figura 16 - Comparação dos resultados do ViTruDe em CPU (esquerda) e GPU (direita) para a mesma imagem de entrada



Fonte: Azevedo (2018).

Os resultados de Azevedo (2018) para o ViTruDe são demonstrados na figura 17, esses resultados foram obtidos fazendo a média aritmética dos tempos para 20 execuções do mesmo frame de entrada. O processamento da imagem salta de três em três pixels nas secções do ViTruDe de cálculo, normalização e classificação dos descritores, ou seja, processa apenas um terço da imagem total nessas fases.

A Figura 17 mostra que a abordagem em GPU é mais eficiente em unidades de tempo de processamento. Nos testes efetuados, com as especificações de hardware mencionadas, a implementação em GPU do ViTruDe no GPU1 é cerca de 14 vezes mais rápida quando comparada com o mesmo procedimento em CPU, enquanto a mesma abordagem no GPU2 é mais de 80 vezes mais rápida do que o ViTruDe em CPU e cerca de seis vezes mais rápida do que o GPU1.

Figura 17 - Comparação entre os tempos de processamento do ViTruDe em CPU e GPU

Funções	Tempo de Processamento (segundos)					
	CPU	Todos os Pixels			Saltando Três Pixels	
		GPU1	GPU2	GPU1	GPU2	
Extração LBP	0,017969	0,000452	0,000193	0,000491	0,000194	
Extração HSV	0,008083	-	-	-	-	
Cálculo dos Descritores		7,383115	0,976206	3,633317	0,471755	
Normalização dos Descritores	≈ 600	0,696364	0,174869	0,090228	0,018046	
Classificação dos Descritores		34,709829	6,238318	5,998721	0,890845	
Tempo Total	≈ 10 minutos	42,80 segundos	7,40 segundos	9,73 segundos	1,39 segundos	

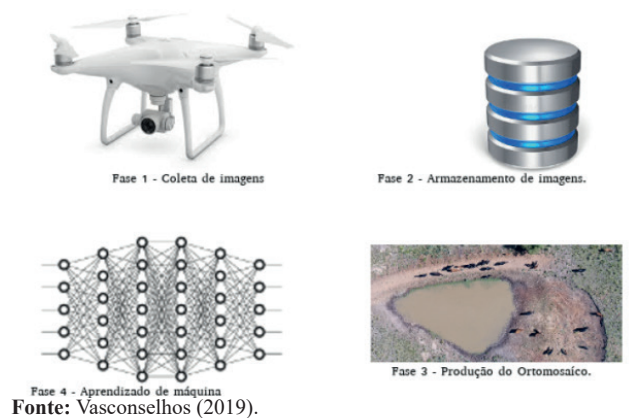
Fonte: Azevedo (2018).

Além de estudos com monitoramento climático de biomas, agricultura de precisão se tem também aplicações da utilização do processamento paralelo na pecuária, em que o manejo extensivo em áreas de milhares de hectares é oneroso, este é o caso do estudo realizado por Vasconcellos (2019).

Vasconcellos (2019) desenvolveu um método autônomo para identificar de forma remota animais em campo com aplicação de pecuária de precisão. Com uso de SR, o autor fez uso de imagens por aerolevantamento com câmara RGB

embarcada em veículo aéreo não tripulado em uma área de 20 ha, disponível na Embrapa Pecuária Sul, situada no município de Bagé, Rio Grande do Sul (Figura 18).

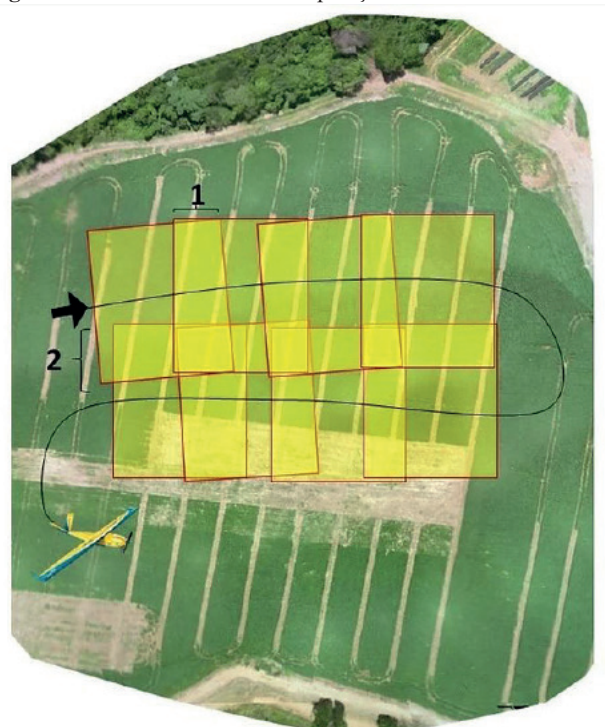
Figura 18 - Metodologia proposta por Vasconselhos (2019)



Fonte: Vasconselhos (2019).

Depois de coletada, as imagens por meio do VANT (Figura 19), Vasconselhos (2019) armazenou essas imagens em um banco de dados (geodatabase), a fim de manter a temporalidade para análises futuras e extração de padrões. Realizada essa etapa, o autor gerou polígonos dos animais identificados em campo (*Regions of Interest – ROI*), em que foi possível classificar as imagens por meio do método na detecção dos animais, e a *Intersection over Union (IoU)*, a qual avalia a porcentagem da sobreposição da predição feita corretamente sobre o objeto real na imagem, obtendo como resultado a contagem dos animais de forma automatizada, associada à identificação da posição geográfica aproximada de cada animal.

Figura 19 - Linhas de voo e sobreposição



Fonte: Vasconselhos (2019).

Para processamento paralelo das imagens geradas pelo VANT, Vasconcelhos (2019) utilizou a biblioteca de computação numérica chamada TensorFlow, a qual possui suporte a processamento paralelo em CPU com múltiplas cores e GPU. Com suporte da Google fez uso da máquina virtual (VM) com Linux e ambiente de desenvolvimento Python 2 e 3 já preparados para pesquisas voltadas para inteligência artificial com CPU e GPU de forma gratuita, totalmente em cloud, chamado Google Colaboratory (Colab).

A Colab foi útil na aceleração de processamentos computacionais de projetos de aprendizado de máquina nos quais o fator “tempo” é determinante. Possui como base o ambiente web Jupyter Notebook, com Python, podendo ser mais rápidos que 20 cores físicos em aplicação de aprendizado profundo, eliminando a etapa de configuração de ambientes de execução para treinamentos naquela aplicação.

A partir dos 2 voos realizados, Vasconcelhos (2019) gerou ortomosaicos com o software OpenDroneMap 0.3 (Figura 20). O ambiente disponível para processamento foi um computador com processador Core i7, 4 cores, memória RAM disponível 8GB, sem GPU, disco rígido de 1TB SATA 7.200 RPM e sistema operacional Linux 4.13.0-38 64 bits. O processo demorou cerca de 2 h em cada voo. De acordo com os resultados de Vasconcelhos (2019), o método foi capaz de detectar animais com uma acurácia de 92%, com 1000 épocas e IoU de 60%.

Figura 20 - Metodologia geral de Vasconcelhos (2019)



Fonte: Vasconcelhos (2019).

3 Conclusão

A utilidade das GPUs nas principais áreas de sensoriamento remoto com uso de imagens de satélite (agricultura, biomas e mudança do uso e ocupação do solo) foi bem mais compreendida neste trabalho com a apresentação dos conhecimentos sobre sensoriamento remoto, geotecnologias e unidades de processamento de um hardware computacional.

Estes conhecimentos auxiliaram no entendimento da resposta espectral de cada alvo na superfície e velocidade de processamento das máquinas computacionais, possibilitando assim uma melhor aplicação de placas gráficas de vídeo em contraste com o uso comum das unidades centrais de processamento.

Este estudo demonstrou também que a utilização de GPUs surge como alternativa promissora na presente era tecnológica e avanço de futuros trabalhos que destinam entendimento das estruturas de núcleos gráficos de processamento com imagens de satélites, e que representam grande quantidade de dados “Big Data” em menor tempo.

Referências

- ALALI, M.I. *et al.* Implementing Image Processing Algorithms in FPGA Hardware. IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies. *Anais*. Amman: 2013. doi: 10.1109/AEECT.2013.6716446.
- ALVARENGA, B.S. *et al.* O Ensino de Conceitos E Práticas de Espectrorradiometria Laboratorial: Estudo de Caso Com Solos do Estado de São Paulo. *Anais XI SBSR*, Belo Horizonte, Brasil, INPE, p. 739-747, 2003.
- APRENDA FAZER. GPU ou Unidade de processamento gráfico: O que é, para que serve e como funciona? Disponível em: <<https://aprendafazer.net/gpu-ou-unidade-de-processamento-grafico-o-que-e-para-que-serve-e-como-funciona/>>. Acesso em 19 out. 2021.
- AZEVEDO, F.A.S. *Paralelização de Algoritmos de Processamento de Imagem em Contexto de Vinha*. Mestrado Integrado em Engenharia Eletrotécnica e de Computadores da Faculdade de engenharia da Universidade do Porto, 2018.
- BÉGUÉ, A. *et al.* Remote sensing and cropping practices: a review. *Remote Sensing*, v.10, n.1, p. 99, 2018. doi: 10.3390/rs10010099.
- CARDIM, G. P. *et al.* Statistical Evaluation and Analysis of Road Extraction Methodologies Using a Unique Dataset from Remote Sensing. *Remote Sensing*, v. 10, p. 620, 2018. doi: 10.3390/rs10040620.
- CARDIM, G. P. *Proposição de plataforma Co-Design para processamento de Imagens de sensoriamento remoto*. 49 f. Tese (Doutorado em Ciências Cartográficas) – Universidade Estadual Paulista, Presidente Prudente – SP, 2019.
- B. CHAPMAN. *et al.* Parallel Computing: From Multicores and GPUs to Petascale. *G. R. Joubert*, Ed. IOS Press BV, 2010.
- CHENG, G. *et al.* Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, v.105, n.10, p.1865–1883, 2017. doi: 10.1109/JPROC.2017.2675998.
- CRÓSTA, A. P. Processamento digital de imagens de

- sensoriamento remoto. *Edição revisada*. Campinas. IG/ UNICAMP, 1993.
- DANELICHEN, V. H. M. *et al.* Automation of choosing hot and cold pixels process in the estimate of the sensitive heat flow and evapotranspiration. *Rev Iberoam. Ciênc. Amb.*, v.11, p.638-651, 2020. doi: 10.6008/CBPC2179-6858.2020.005.0057.
- DANELICHEN, V. H.M. *et al.* Assessment of spectral indexes for estimating soil water content in the Brazilian Pantanal. *Ciênc. Natura*, v.43, p.1-17, 2021. doi: 10.5902/2179460X42724
- FLORENZANO, T. G. Iniciação em Sensoriamento Remoto São Paulo: *Oficina de Textos*, 128 p, 2011.
- GEBALI, F. Algorithms and Parallel Computing. *John Wiley & Sons*, 2011.
- GEEKBOOTS. Difference Between CPU, GPU and TPU. Disponível em: <<https://www.geekboots.com/story/cpu-vs-gpu-vs-tpu>>. Acesso em 19 out. 2022.
- GUEDES, J.C.F.; SILVA, S.M.P. Remote sensing in vegetation study: physical principles, sensors and methods. *ACTA Geográfica*, v.12, n.29, p.127-144, 2018. doi: 10.5654/acta.v12i29.4001.
- GOMES, R. S. R. *et al.* The Surface Energy Balance Algorithm for Land (SEBAL) framework in Graphics Processing Units (GPU) using Cuda and OpenCL. *Rev Bras. Geog. Fis.*, v.14, n.3, p.1805-1814, 2021. doi: 10.26848/rbgf.v14.3.p1805-1814.
- GULO, C. A. S. J. *Técnicas de computação de alto desempenho para o processamento e análise de imagens complexas da cavidade pélvica feminina*. porto: universidade do porto, 2014.
- JENSEN, J.R. Sensoriamento remoto do ambiente: uma perspectiva em recursos terrestres. Rio de Janeiro: *Parêntese*, 2009.
- KUENZER, C. *et al.* Remote sensing time series revealing land surface dynamics: Status quo and the pathway ahead. *Remote Sensing Time Series*. Springer, Cham, p. 1-24, 2015. doi: 10.1007/978-3-319-15967-6_1.
- MARK, W.R. *et al.* Cg: A system for programming graphics hardware in a c-like language. *ACM Transactions Graphics*, v.22, n.3, p.896-907, 2003. doi: 10.1145/882262.882362.
- MENESES, P.R; ALMEIDA, T. Introdução ao processamento de imagens de sensoriamento remoto. Brasília: *UNB*, 2012.
- MUNSHI, A. The OpenCL Specification. In: IEEE Hot Chips 21 Symposium (HCS), p. 1–314. *IEEE*, 2009.
- NVIDIA. NVIDIA CUDA C Programming Guide. 2011. Disponível em <http://developer.download.nvidia.com/compute/DevZone/docs/html/C/doc/CUDA_C_Programming_Guide.pdf>. Acesso em: 10 out. 2022.
- OLIVEIRA, S.S.T. *Explorando Paralelismo em Big Data no Processamento de Séries Temporais de Imagens de Sensoriamento Remoto*. Goiânia: Universidade Federal de Goiás, 2019.
- PAGE, D. A Practical Introduction to Computer Architecture. [s.l.]: *Springer*, 2009.
- PINHEIROS, A.I.S. *Modelo Computacional Paralelo Baseado em GPU para Cálculo do Campo de Vento de um Sistema de Dispersão Atmosférica de Radionuclídeos*. Rio de Janeiro: UFRJ, 2017.
- ROUSE, J.W. *et al.* Deering. Monitoring vegetation systems in the Great Plains with ERTS, *Third ERTS Symposium*, NASA SP-351 I, p.309-317, 1973.
- SANDERS, J.; KANDROT, E. CUDA by example: an introduction to general purpose GPU programming. *Addison-Wesley Professional*, 2010.
- SOARES, J. *et al.* Estudo da dinâmica espaço-temporal do NDVI no Município de Sorriso-MT. *Rev Bras. Geog. Fis.*, v.13, p.834, 2020. doi: 10.26848/rbgf.v13.2.p834-841
- VASCONSELHOS, B.C. *Método aplicado ao monitoramento remoto de animais baseado em aerolevantamento com VANT e aprendizagem profunda*. Bagé: Universidade Federal do Pampa, 2019.