



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21d/2022/07.20.15.45-NTC

TATHU - SOFTWARE PARA RASTREIO E ANÁLISE DO CICLO DE VIDA DE SISTEMAS CONVECTIVOS

Douglas Messias Uba
Renato Galante Negri
Diego Pereira Enoré
Izabelly Carvalho da Costa
Aurelienne Aparecida Souza Jorge

URL do documento original:

<http://urlib.net/8JMKD3MGP3W34T/47AF772>

INPE
São José dos Campos
2022

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE
Coordenação de Ensino, Pesquisa e Extensão (COEPE)
Divisão de Biblioteca (DIBIB)
CEP 12.227-010
São José dos Campos - SP - Brasil
Tel.:(012) 3208-6923/7348
E-mail: pubtc@inpe.br

CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELLECTUAL DO INPE - CEPPII (PORTARIA Nº 176/2018/SEI-INPE):

Presidente:

Dra. Marley Cavalcante de Lima Moscati - Coordenação-Geral de Ciências da Terra (CGCT)

Membros:

Dra. Ieda Del Arco Sanches - Conselho de Pós-Graduação (CPG)
Dr. Evandro Marconi Rocco - Coordenação-Geral de Engenharia, Tecnologia e Ciência Espaciais (CGCE)
Dr. Rafael Duarte Coelho dos Santos - Coordenação-Geral de Infraestrutura e Pesquisas Aplicadas (CGIP)
Simone Angélica Del Ducca Barbedo - Divisão de Biblioteca (DIBIB)

BIBLIOTECA DIGITAL:

Dr. Gerald Jean Francis Banon
Clayton Martins Pereira - Divisão de Biblioteca (DIBIB)

REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:

Simone Angélica Del Ducca Barbedo - Divisão de Biblioteca (DIBIB)
André Luis Dias Fernandes - Divisão de Biblioteca (DIBIB)

EDITORAÇÃO ELETRÔNICA:

Ivone Martins - Divisão de Biblioteca (DIBIB)
André Luis Dias Fernandes - Divisão de Biblioteca (DIBIB)



MINISTÉRIO DA CIÊNCIA, TECNOLOGIA E INOVAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

sid.inpe.br/mtc-m21d/2022/07.20.15.45-NTC

TATHU - SOFTWARE PARA RASTREIO E ANÁLISE DO CICLO DE VIDA DE SISTEMAS CONVECTIVOS

Douglas Messias Uba
Renato Galante Negri
Diego Pereira Enoré
Izabelly Carvalho da Costa
Aurelienne Aparecida Souza Jorge

URL do documento original:

<http://urlib.net/8JMKD3MGP3W34T/47AF772>

INPE
São José dos Campos
2022



Esta obra foi licenciada sob uma Licença Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada.

This work is licensed under a Creative Commons Attribution-NonCommercial 3.0 Unported License.

RESUMO

Sistemas convectivos são compostos por diversos tipos de nuvens e no geral estão associados com condições de tempo severas, como por exemplo granizo, descargas elétricas, chuvas intensas, ventos fortes, entre outras. Este trabalho apresenta detalhes do planejamento e implementação de um pacote de software denominado TATHU - *Tracking and Analysis of Thunderstorms*, construído para permitir o rastreamento automático e a análise do ciclo de vida desses sistemas. A arquitetura do software permite a extensão das funcionalidades de modo eficiente, a construção de técnicas para detecção, caracterização, acompanhamento e previsão desses elementos, além de possibilitar o uso de diferentes tipos de dados e variáveis ambientais. Como destaque, a solução propõe a utilização de técnicas de Geoinformática e banco de dados espaciais para auxiliar a análise computacional dos sistemas, além de um modelo conceitual para definição do problema a partir de interfaces abstratas. Por último, o trabalho apresenta alguns estudos de caso e os resultados obtidos com a aplicação dos conceitos propostos. TATHU é implementado em Python e distribuído como um pacote dessa linguagem, com código fonte aberto, multiplataforma e livre.

Palavras-chave: Sistemas Convectivos. Meteorologia. Nowcasting. Sensoriamento Remoto. Satélites. Radar. Processamento de Imagens. Geoinformática.

TATHU - SOFTWARE FOR TRACKING AND ANALYZING THE LIFE CYCLE OF CONVECTIVE SYSTEMS

ABSTRACT

Convective systems are defined as an organized ensemble of thunderstorm clusters and are associated with severe weather events and natural disasters, for example hail, lightning, precipitation extremes, high winds, among others. This work describes the key design and implementation aspects of developed software called **TATHU - Tracking and Analysis of Thunderstorms**, that can be used for automatic tracking and analyzing of the life cycle of these systems. The devised software architecture permits easy extension of its capabilities, the construction of detection, characterization, tracking and forecast techniques, and the usage of different types of remote sensing data and variables. Furthermore, the work proposes the use of Geoinformatics techniques and spatial databases in order to aid in the analysis and computational representation of the convective systems, in addition to a conceptual model for defining the problem using abstract interfaces. Finally, we show a few case studies and the results obtained with proposed concepts. TATHU is implemented in Python and distributed as a package, free, open source, and cross-platform.

Keywords: Convective Systems. Meteorology. Nowcasting. Remote Sensing. Satellite. Radar. Image Processing. Geoinformatics.

LISTA DE FIGURAS

	<u>Pág.</u>
1.1 Dados adquiridos a partir de satélites geoestacionários e radares meteorológicos. Exemplos: (a) Composição <i>true-color</i> , satélite GOES-16 e (b) canal 13 destacando nuvens com baixas temperaturas. (c) Cobertura da rede de radares meteorológicos do Brasil e (d) destaque para o radar localizado em Manaus-AM.	1
1.2 SC e o comportamento espaço-temporal: formam-se em determinada região e evoluem no tempo (posição, tamanho e composição microfísica).	3
1.3 Detecção, caracterização, acompanhamento e previsão: etapas gerais da metodologia para detecção e acompanhamento de objetos espaço-temporais em dados observados.	3
1.4 Situações consideradas pelo ForTraCC durante o acompanhamento dos SCM: (a) continuidade, (b) <i>split</i> e (c) <i>merge</i> . As linhas tracejadas representam os sistemas no instante $t-\Delta t$	6
2.1 Organização do TATHU: diferentes níveis de abstração e funcionalidades.	11
2.2 Módulo para leitura e escrita dos dados obtidos a partir do rastreamento dos SC. Diferentes possibilidades com base na implementação de duas interfaces.	13
2.3 Entidades para rastreamento e análise de SC propostas pelo pacote TATHU.	15
2.4 Modelo conceitual para definição do problema de rastreamento e análise do ciclo de vida de SC e exemplos de classes concretas - (a) <i>Detector</i> , (b) <i>Descriptor</i> , (c) <i>Tracker</i> e (d) <i>Forecaster</i>	16
2.5 Pseudocódigo da sequência de operações necessárias para detecção, caracterização, acompanhamento e previsão dos SC utilizando as interfaces abstratas propostas.	17
2.6 Transformações geométricas (rotação, translação e escala): (a) uma possibilidade é a aplicação para modificar/deformar a representação vetorial de cada SC na previsão e diferentes transformações podem ser utilizadas para cada vértice.	19
2.7 (a) Multi-limiarização: cada polígono que compõe o SC é obtido a partir de um limiar de temperatura de brilho diferente. (a) < 235 Kelvin, (b) < 220 Kelvin e (c) < 210 Kelvin.	19

2.8	Formas de representação alternativas: cada SC pode ser representado por formas geométricas diferentes da sua silhueta original (utilizando uma elipse (a) ou o envoltório convexo (b), por exemplo). Desse modo, temos a capacidade de extrair novas métricas morfológicas.	20
2.9	Fragmento do arquivo CSV gerado. Cada linha representa um SC, o qual possui um identificador único, a data e demais atributos.	23
2.10	Exemplo de visualização dos SC detectados em um instante de tempo específico.	24
2.11	Exemplo de visualização do ciclo de vida de SC a partir do suporte oferecido pelo pacote TATHU.	27
3.1	Metodologia para análise espaço-temporal de sistemas convectivos.	29
3.2	Identificação dos SC: (a) imagem infravermelho termal, (b) resultado do processo de limiarização, (c) componentes conectados e (d) vetorização. Em (e), a representação vetorial é apresentada em conjunto com a imagem de satélite.	30
3.3	(a) Imagem adquirida pelo GOES-16 em 13 de agosto de 2017 - 00:00 UTC: canal 13 ($10,3 \mu\text{m}$) e resolução espacial de 2 km, (b) SC identificados neste mesmo horário e (c) descargas atmosféricas observadas pelo sensor GLM/GOES-16 nos dias 13, 14 e 15 de agosto de 2017. Cada ocorrência está representada por 1 pixel. Em (d), todos os polígonos de cada SC detectado no período são apresentados (contornos).	31
3.4	Evolução de um determinado SC com duração de 20 horas: (a) temperatura média e tamanho e (b) temperatura média e descargas atmosféricas dentro do SC. Em (b), utilizou-se uma consulta com restrição espacial e temporal.	32
3.5	Consulta espaço-temporal para determinar as descargas atmosféricas em cada instante do ciclo de vida de um SC x	33
3.6	(a) Trajetória (polígono vermelho) do SC com os valores máximos de refletividade (linha preta) e a precipitação média estimada (linha verde) do SC para cada passo de tempo. Área sombreada representa regiões onde há informações de refletividade menores que 35 dBz (b) Trajetória dos SC representados pela união dos polígonos de cada cluster: diagnóstico (azul) e previsões de deslocamento (vermelho), considerando um intervalo de 30 minutos.	34

LISTA DE ABREVIATURAS E SIGLAS

CAPPI	–	Constant Altitude Plan Position Indicator
CPTEC	–	Centro de Previsão de Tempo e Estudos Climáticos
CLI	–	Command-line Interface
CSV	–	Comma-separated Values
dBZ	–	Valores de Refletividade do Radar
DECEA	–	Departamento de Controle do Espaço Aéreo
EPSG	–	European Petroleum Survey Group
ForTraCC	–	Forecast and Tracking the Evolution of Cloud Clusters
GLM	–	Geostationary Lightning Mapper
GOES	–	Geostationary Operational Environmental Satellite
GUI	–	Graphical User Interface
INPE	–	Instituto Nacional de Pesquisas Espaciais
JSON	–	JavaScript Object Notation
KML	–	Keyhole Markup Language
MTT	–	Multiple Target Tracking
OGC	–	Open Geospatial Consortium
PPI	–	Plan Position Indicator
SC	–	Sistema Convectivo
SCM	–	Sistema Convectivo de Mesoescala
SGBD	–	Sistema Gerenciador de Banco de Dados
SIG	–	Sistemas de Informações Geográficas
SQL	–	Standard Query Language
TATHU	–	Tracking and Analysis of Thunderstorms
UUID	–	Universally Unique Identifiers

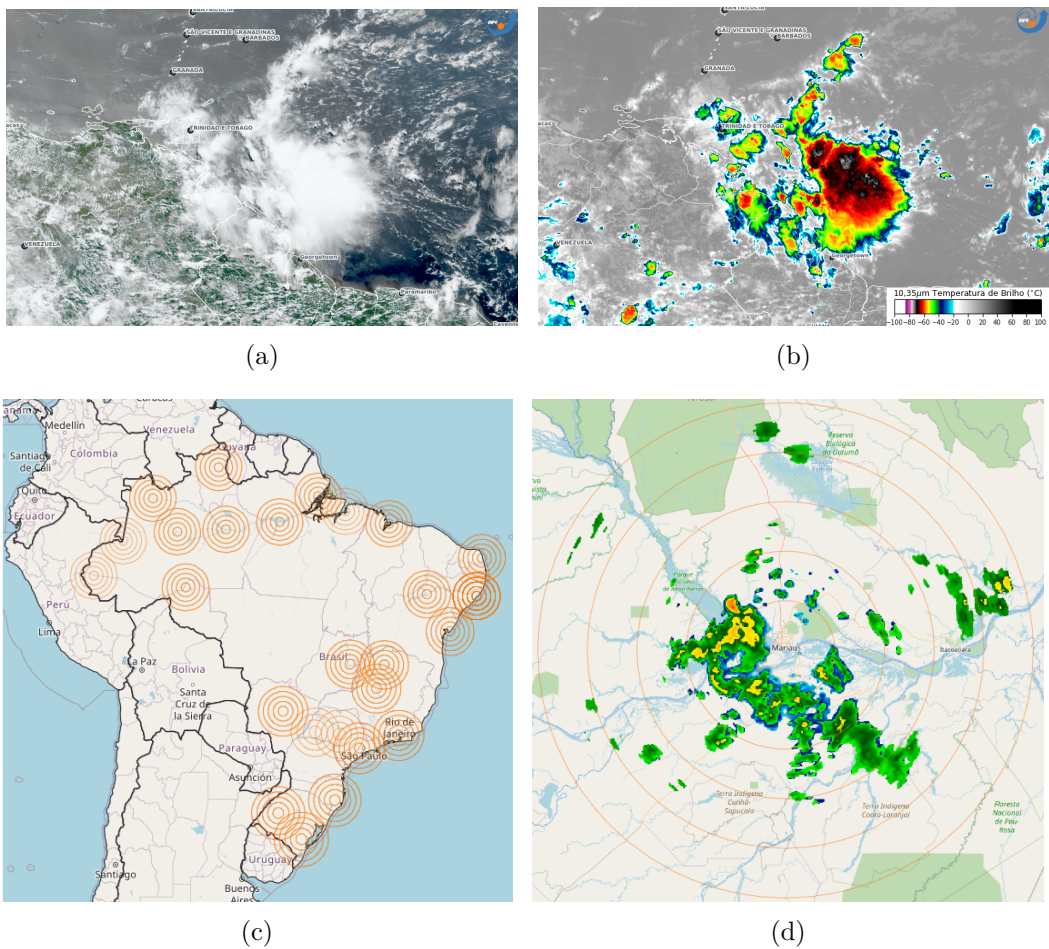
SUMÁRIO

	<u>Pág.</u>
1 INTRODUÇÃO	1
1.1 Contextualização do Problema	2
1.2 ForTraCC	5
1.3 Motivação	7
1.4 Objetivos	8
2 DESIGN AND IMPLEMENTATION	11
2.1 Entidades de Rastreo e Análise	14
2.2 A classe <code>ConvectiveSystem</code>	16
2.3 Exemplos de Utilização	21
3 EXPERIMENTOS E RESULTADOS	29
3.1 Primeiro Experimento	29
3.1.1 Dados	30
3.1.2 Análises	31
3.2 Segundo Experimento	33
3.2.1 Dados	33
3.2.2 Análises	33
3.3 Terceiro Experimento	34
4 CONCLUSÃO	37
REFERÊNCIAS BIBLIOGRÁFICAS	39

1 INTRODUÇÃO

A observação da atmosfera a partir de dados de satélites geoestacionários e radares meteorológicos (Fig. 1.1) caracteriza-se como uma importante ferramenta para a Meteorologia, em consequência, para a sociedade em geral. A alta resolução temporal dos dados adquiridos por esses instrumentos (observações variando de 5 até 30 minutos, aproximadamente) permite a visualização e o acompanhamento de fenômenos meteorológicos em tempo quase real, com destaque para aqueles associados com eventos extremos, como por exemplo chuvas intensas, granizo, ventos fortes, descargas elétricas, entre outros.

Figura 1.1 - Dados adquiridos a partir de satélites geoestacionários e radares meteorológicos. Exemplos: (a) Composição *true-color*, satélite GOES-16 e (b) canal 13 destacando nuvens com baixas temperaturas. (c) Cobertura da rede de radares meteorológicos do Brasil e (d) destaque para o radar localizado em Manaus-AM.



Fonte: DSAT/INPE e SIGMA.

Eventos de tempo severos geralmente são responsáveis por prejuízos econômicos e danos à população e, eventualmente, podem até causar vítimas fatais. Além disso, a ocorrência de eventos desse tipo tem aumentado em todo o mundo e a tendência é que esse cenário se agrave (IPCC, 2012). Nesse sentido, cada vez mais é necessário pesquisar e compreender a natureza desses fenômenos, provendo ferramentas com o objetivo de diminuir as vulnerabilidades e os impactos sobre as pessoas. Em particular, são reconhecidos os desastres produzidos por precipitações intensas, alta atividade de descargas elétricas e vendavais no Brasil. Destacam-se aqui os Sistemas Convectivos.

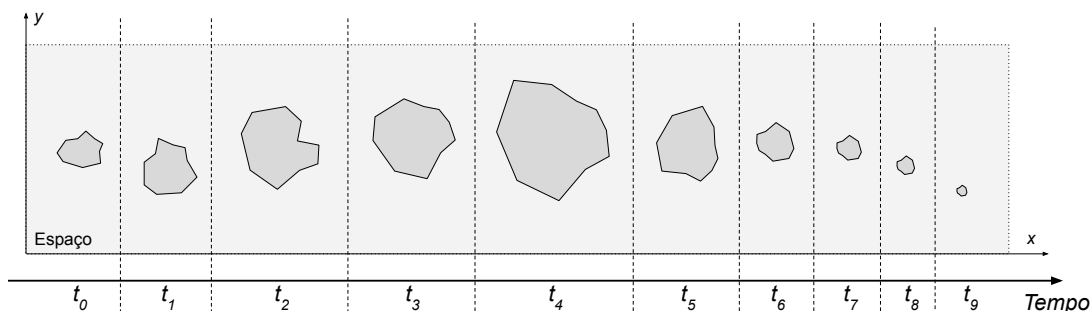
Sistemas Convectivos (SC) são compostos por diversos tipos de nuvens e geralmente estão associados com condições de tempo severas. Sistemas desse tipo são responsáveis pela maior parte das chuvas nas regiões tropicais e também por condições de tempo extremas em várias regiões do planeta Terra (VELASCO; FRITSCH, 1987). Diversos trabalhos propõem métodos automáticos de acompanhamento desses elementos com o objetivo de fornecer, para cada SC individual, características que possam descrever seu comportamento espaço-temporal, incluindo informações como tamanho, forma, temperatura mínima, altura, horário preferencial, velocidade de propagação, interações com outros SC, entre outras variáveis. (CRANE, 1979) (DIXON; WIENER, 1993) (MACHADO et al., 1994) (MACHADO; LAURENT, 2004) (VILA et al., 2008) (FIOLLEAU; ROCA, 2013) (MAKRIS; PRIEUR, 2014).

1.1 Contextualização do Problema

Sistemas de nuvem possuem um comportamento espaço-temporal: têm origem em uma determinada localização geográfica e modificam-se no tempo em relação a posição, tamanho e composição microfísica (distribuições de gotas e de cristais de gelo). A Figura 1.2 apresenta uma representação simples de um objeto espaço-temporal com alterações de localização e tamanho. As alterações dessas características nos SC ocorrem em períodos curtos de tempo, na escala de minutos ou horas, sendo desejável para o acompanhamento a utilização de observações adquiridas por instrumentos com alta resolução temporal.

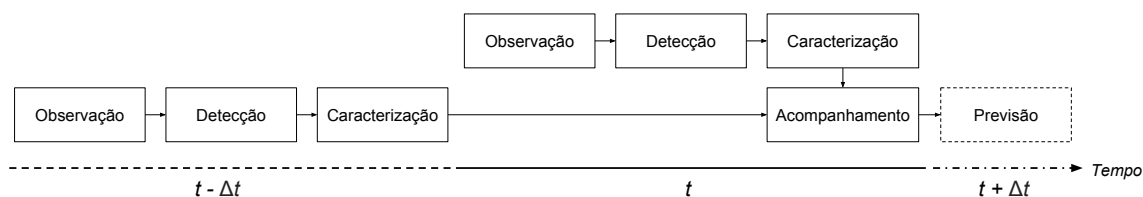
O acompanhamento automático de SC em dados de satélites e radares meteorológicos pode ser descrito como um problema MTT - *Multiple Target Tracking* (traduzido para o português como Rastreamento de Alvos Múltiplos) (MAKRIS; PRIEUR, 2014). Os passos gerais envolvidos no processo são apresentados na Figura 1.3 e detalhados posteriormente.

Figura 1.2 - SC e o comportamento espaço-temporal: formam-se em determinada região e evoluem no tempo (posição, tamanho e composição microfísica).



Fonte: Autores.

Figura 1.3 - Detecção, caracterização, acompanhamento e previsão: etapas gerais da metodologia para detecção e acompanhamento de objetos espaço-temporais em dados observados.



Fonte: Autores.

Seja t um determinado instante no tempo e Δt um intervalo de tempo qualquer (*e.g.* 15 minutos). No instante $t - \Delta t$ (passado) ocorre uma **observação**. Adquirida a observação, o próximo passo consiste na **detecção** dos objetos de interesse existentes. Uma vez detectados, a etapa seguinte envolve a **caracterização** desses objetos, ou seja, definem-se características e atributos individuais para cada um dos elementos. Os atributos extraídos são particulares para cada aplicação e dependem da natureza dos dados e dos objetivos pretendidos. Por exemplo, podem ser definidas propriedades como tamanho, temperatura, direção, forma, entre outras diversas variáveis.

No instante t (presente), uma nova observação acontece, sendo possível nesse momento a realização do **acompanhamento**. A etapa de acompanhamento é determinar como os objetos de interesse se comportaram no intervalo de tempo Δt , decorrido entre as duas observações. Nessa etapa, os processos de detecção e caracterização também são utilizados. Em síntese, o acompanhamento deve ser capaz de determinar

quais objetos da observação anterior ainda estão presentes no instante t (associação), além de também identificar novos objetos que surgiram. Após a caracterização e associação, novos atributos podem ser extraídos, relacionados com alterações observadas em cada objeto, por exemplo variação no tamanho, aumento de temperatura, mudança de direção, entre outras modificações. Por último, uma próxima etapa pode incluir a **previsão** para instantes de tempo futuros ($t + \Delta t$, por exemplo), ou seja, estimar qual será o comportamento dos objetos com o passar do tempo. Em resumo:

- **Observação:** aquisição de dados a partir de instrumentação específica, como imagens digitais obtidas a partir de satélites de órbita geostacionária ou polar, medições de radares meteorológicos, entre outras fontes;
- **Detecção:** etapa para identificar os objetos de interesse existentes nos dados observados. No caso específico de imagens digitais, pode-se considerar a utilização de diferentes técnicas de processamento, como por exemplo: limiarização, segmentação, classificação, filtros, entre outras. A metodologia empregada na etapa de detecção é dependente do tipo de aplicação e da natureza dos objetos de interesse em questão;
- **Caracterização:** extração de diferentes tipos de atributos e posterior classificação. Neste caso, pode-se considerar atributos espectrais (medidas de um sensor em diferentes canais), análises estatísticas (média, variância, etc) e de forma (tamanho, orientação, retangularidade, entre outros);
- **Acompanhamento:** inclui as etapas de detecção e caracterização seguidas de um processo de associação. O objetivo é determinar o comportamento e a evolução dos objetos de interesse, bem como o surgimento de novos objetos;
- **Previsão:** a partir de conhecimentos especializados (modelos e parametrização) e com base no histórico de cada objeto, tem o objetivo de prever qual será o comportamento para instantes futuros.

Os métodos automáticos para análise de SC, no geral, incluem todos os passos descritos acima. O processo de detecção não difere muito entre os métodos propostos. No geral, consiste em uma operação de limiarização em imagens produzidas por sensores remotos a bordo de satélites geostacionários. Tratam-se de imagens no infravermelho termal, posicionado em uma região de janela atmosférica (aproximadamente no intervalo de 10 a 12 μm). Neste caso, o limiar representa um valor de

temperatura de brilho que se aproxima da temperatura real do corpo. A escolha do valor do limiar inclui uma ampla faixa, dependendo das análises a serem realizadas (*e.g.* comumente utiliza-se um valor de temperatura de brilho frio, entre 208 a 255 K). Por outro lado, a etapa de acompanhamento é muito mais heterogênea e complexa, utilizando algoritmos e técnicas diversas, como por exemplo: mínimas distâncias, sobreposição de áreas, análises estatísticas, métodos de correlação espacial, posições relativas de centróides, entre outras (FIOLLEAU; ROCA, 2013).

1.2 ForTraCC

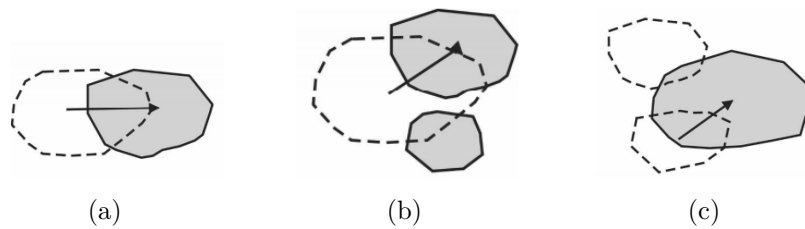
O CPTEC - Centro de Previsão de Tempo e Estudos Climáticos do INPE mantém trabalhos de pesquisa e desenvolvimento na área de rastreamento e análise do ciclo de vida de SC. Mais especificamente, o centro executa operacionalmente o rastreamento automático desses elementos e divulga os resultados obtidos, além de disponibilizar para parceiros interessados a ferramenta computacional responsável pela execução das análises, o ForTraCC. *Forecast and Tracking the Evolution of Cloud Clusters* (VILA et al., 2008) é um método automático específico para detecção e acompanhamento de sistemas convectivos de meso-escala (SCM) em imagens de satélite geoestacionários. ForTraCC é implementado em Fortran e os resultados podem ser acessados a partir do endereço www.sigma.cptec.inpe.br/fortracc.

A partir dos dados brutos em projeção satélite do instante $t - \Delta t$, o primeiro passo consiste no remapeamento para uma grade regular. Em seguida, a detecção dos diferentes SCM é realizada a partir de uma operação de limiarização dos valores de temperatura de brilho (< 235 K) em imagens do infravermelho termal que opera na janela atmosférica em torno de $11 \mu\text{m}$. Após a limiarização, os componentes conectados são rotulados e definem os sistemas convectivos. Definidas as regiões de cada SCM, a próxima etapa inclui a caracterização dos sistemas. São exemplos de atributos extraídos: temperaturas média e mínima, tamanho do SCM (número de *pixels*, área em km^2), centroide geográfico, raio efetivo, inclinação, excentricidade, eixo de inércia, etc. Os diferentes SCM são então armazenados em um formato próprio, de acordo com a hora e a data de detecção.

No instante t , uma nova imagem de satélite é recebida. Os processos de detecção e caracterização são utilizados novamente, porém, seguidos da etapa de associação. O método de associação de um mesmo SCM no tempo t com o tempo anterior é realizado a partir de um critério de sobreposição de área mínima. Para cada sistema do instante t , é necessário calcular a área de interseção com cada sistema existente no instante $t - \Delta t$. Utiliza-se uma área mínima de interseção para considerar que o SCM

é uma continuidade. Se a área de interseção for menor, o sistema não é considerado como sendo o mesmo sistema do tempo anterior (*i.e.* geração espontânea). No total, quatro tipos de situações são consideradas:

Figura 1.4 - Situações consideradas pelo ForTraCC durante o acompanhamento dos SCM: (a) continuidade, (b) *split* e (c) *merge*. As linhas tracejadas representam os sistemas no instante $t-\Delta t$.



Fonte: Adaptado de Vila et al. (2008).

- **Geração Espontânea:** o SCM é identificado em uma imagem, porém, não está presente na imagem anterior ou não cumpre com o critério de sobreposição de área mínima;
- **Continuidade:** o SCM é identificado em uma imagem e está presente na imagem anterior, *i.e.* cumpre com o critério de sobreposição de área mínima (Fig. 1.4.a);
- **Split:** no instante t , existe um SCM que cumpre com o critério de sobreposição de área mínima com dois ou mais sistemas no tempo $t-\Delta t$ (Fig. 1.4.b);
- **Merge:** no instante $t-\Delta t$, existem dois ou mais SCM que obedecem o critério de sobreposição de área mínima com somente um sistema no tempo t (Fig. 1.4.c).

Após a definição da origem do SCM no tempo t utilizando os critérios apresentados acima, sendo essa diferente de uma geração espontânea, calcula-se as alterações dos atributos no tempo decorrido Δt . Por último, o método fornece a previsão de evolução dos sistemas para os próximos 30, 60 e 90 minutos.

1.3 Motivação

Os avanços tecnológicos na aquisição de dados geoespaciais produziram um aumento considerável da quantidade de informações disponíveis, oferecendo observações cada vez melhores em termos de resolução espacial, espectral e temporal. Esse enorme conjunto de dados espaço-temporais tem motivado desafios em diversas áreas da Ciência da Computação. Para manipular esses dados devidamente, os sistemas computacionais estão evoluindo para representar, acessar, analisar e visualizar grandes bases de dados de maneira eficiente e integrada. (FERREIRA, 2012) (FERREIRA et al., 2020)

Além disso, nos últimos 20 anos, observa-se um grande crescimento na criação, utilização e adoção de novas tecnologias associadas ao desenvolvimento de soluções computacionais. Alguns tópicos incluem, por exemplo:

- Aumento da abstração em linguagens de alto nível ou de *scripting*, encapsulando processos mais complexos, permitindo que os usuários manipulem os dados de maneira mais simplificada e alheia aos detalhes relacionados ao hardware. Destacam-se neste contexto as linguagens Python e JavaScript;
- Advento da computação em nuvem, com serviços capazes de armazenar e processar grandes volumes de dados;
- Popularização do conceito de API - *Application Programming Interface* e definição de boas práticas associadas a sua construção;
- Surgimento de novas ferramentas e estruturas de desenvolvimento, produzindo uma nova geração de especialistas que entendem os princípios específicos de sua área de atuação e podem escrever programas de computador;
- Avanços em padrões de armazenamento e intercâmbio de dados mais flexíveis;
- Expansão global de dispositivos móveis, entre outras tecnologias.

Ao mesmo tempo, áreas da Ciência da Computação ganham destaque, incluindo conceitos de *Big Data*, Aprendizado de Máquina (em inglês, *Machine Learning*), Aprendizado Profundo (*Deep Learning*), Ciência de Dados (*Data Science*), Inteligência Artificial, entre outros. Em resumo, a evolução tecnológica é rápida e está acelerando.

Neste contexto, é importante uma atualização constante das tecnologias desenvolvidas pelo INPE, de modo atender eficientemente demandas atuais e futuras.

Atualmente, o ForTraCC é a solução oficial do CPTEC/INPE para prover o acompanhamento do ciclo de vida de SC. Sua concepção e desenvolvimento estão estabelecidos no final da década de 90, início da década 2000 (aproximadamente 20 anos atrás). A linguagem utilizada para a construção foi o Fortran. Embora a linguagem não seja fator impeditivo para a evolução da ferramenta, considera-se, no escopo desse trabalho, que uma linguagem mais moderna ofereça melhor grau de extensibilidade e aderência a padrões industriais e científicos já estabelecidos na área de Geoinformática.

Outro ponto importante está relacionado ao modo de armazenamento dos resultados. O ForTraCC utiliza um padrão próprio para armazenar as informações sobre os SC rastreados. Tratam-se de arquivos em formato texto, os quais oferecem baixa capacidade de exploração e análise. Nesse sentido, uma motivação é prover maior aderência aos padrões e formatos atualmente utilizados na área, incluindo banco de dados com suporte espacial, de modo fornecer maior eficiência para o intercâmbio e análise desses resultados.

Por fim, a motivação principal é prover uma **arquitetura de software extensível**, capaz de auxiliar na construção e melhoria de soluções diversas para o problema de rastreio e análise de SC em dados observados. Ou seja, não trata-se apenas da implementação de um método ou algoritmo específico para um tipo específico de dado, mas sim da disponibilização de uma base tecnológica comum que suporta diferentes abordagens e metodologias.

1.4 Objetivos

Considerando a importância do tema, as motivações apresentadas e a necessidade de acompanhar, compreender e prever o ciclo de vida de SC, este trabalho tem como objetivo principal:

- A organização, implementação e disponibilização de um pacote de software extensível e modular, livre e de código aberto, para processamento, detecção, análise e previsão do ciclo de vida de SC a partir de dados de sensoriamento remoto.

Neste contexto, considera-se como pacote de software um conjunto de classes, scripts,

algoritmos, utilitários e executáveis que permitem e auxiliam a manipulação computacional de dados ambientais no escopo do problema descrito na seção anterior. O software deve ser planejado e construído utilizando diferentes níveis de abstração, permitindo o uso por desenvolvedores e também por usuários finais.

Associado ao objetivo principal, destacam-se os seguintes tópicos:

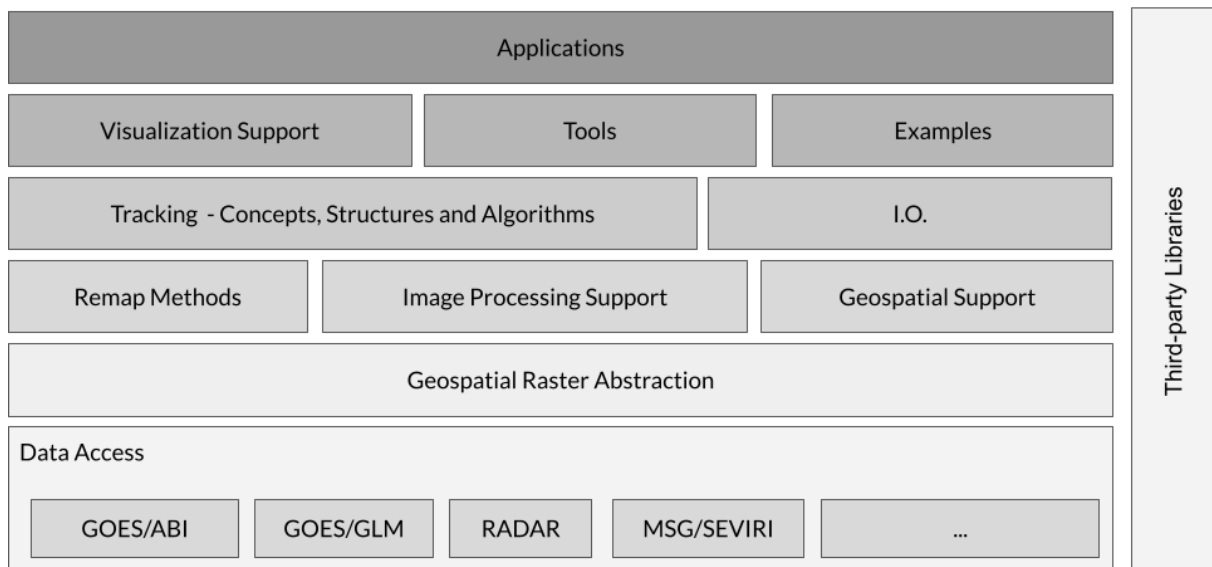
- Modernização dos métodos e ferramentas desenvolvidos e disponibilizados pelo INPE para o rastreo e a análise de objetos de interesse em imagens e dados meteorológicos, com destaque para a metodologia implementada na ferramenta ForTraCC;
- Propor e avaliar a utilização de banco de dados espaciais e técnicas de Geoinformática na análise espaço-temporal de SC, utilizando diversas fontes de dados (*i.e.* imagens obtidas por satélite ambientais, medidas de radares meteorológicos, descargas elétricas e outras variáveis atmosféricas estimadas, como por exemplo vento);
- Fomentar a pesquisa, o desenvolvimento e a aplicação de métodos computacionais para detecção automática de SC e acompanhamento do ciclo de vida incluindo técnicas de processamento de imagem, reconhecimento de padrões, visão computacional, aprendizado de máquina, dentre outras áreas associadas;
- Contribuir efetivamente com o sistema operacional de previsão em curtíssimo prazo (*nowcasting*) do CPTEC/INPE. Define-se sistema de previsão em curtíssimo prazo como a previsão de tempo considerando um período de até 6 horas. Neste contexto, o rastreo, análise e extrapolação do ciclo de vida de SC e variáveis associadas possuem importante destaque.

Este trabalho está estruturado do seguinte modo: o Capítulo 2 apresenta os detalhes do planejamento e implementação do pacote de software desenvolvido, além de exemplos de utilização. Em seguida, o Capítulo 3 descreve alguns estudos de caso, experimentos e os resultados obtidos com a aplicação dos conceitos propostos. Finalizando o texto, o Capítulo 4 expõe as conclusões e perspectivas futuras.

2 DESIGN AND IMPLEMENTATION

Este capítulo apresenta os aspectos mais relevantes do projeto de software utilizado no desenvolvimento do TATHU. O propósito é prover uma estrutura modular e extensível, melhorando em conjunto o desempenho dos algoritmos de rastreamento e análise para volumes de dados maiores e permitindo a inclusão de novas funcionalidades de modo eficiente. Além disso, o software oferece suporte para a utilização de diferentes tipos de dados ambientais e permite a utilização de banco de dados espaciais e técnicas de Geoinformática para auxiliar a análise dos elementos de interesse. Outro ponto importante é a apresentação de um modelo conceitual para definição do problema a partir de interfaces abstratas. A Figura 2.1 apresenta a estrutura do software em camadas e os diferentes níveis de abstração associados. TATHU é implementado em Python e distribuído como um pacote dessa linguagem, com código fonte aberto, multiplataforma e livre.

Figura 2.1 - Organização do TATHU: diferentes níveis de abstração e funcionalidades.



Fonte: Autores.

Os itens a seguir detalham brevemente cada nível:

- **Data Access**: este módulo permite que o pacote de software acesse dados geoespaciais armazenados em diferentes fontes de dados e formatos. Basicamente, neste nível é tratado a decodificação e codificação dos arquivos

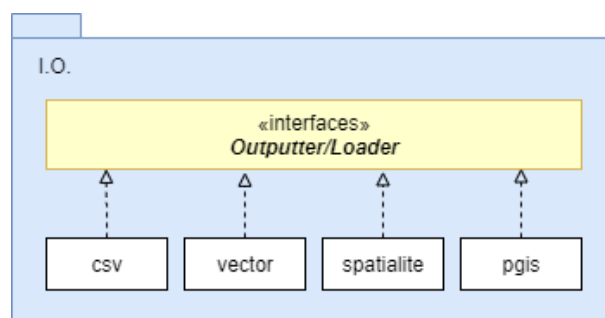
fornecidos por satélites, além de outros sensores e instrumentos;

- **Geospatial Raster Abstraction:** essa camada disponibiliza uma interface genérica para acesso aos elementos e valores dos dados geoespaciais a partir de uma representação matricial. Em conjunto com a camada **Data Access**, o objetivo é desvincular a manipulação dos dados das várias alternativas de armazenamento, incluindo diferentes formatos de arquivos, banco de dados, memória, entre outras;
- **Remap Methods:** esse módulo fornece funções para a manipulação dos dados geoespaciais considerando os diferentes sistemas de referência espacial possíveis. Neste nível, por exemplo, está disponível a conversão dos dados em projeção satélite para uma grade regular;
- **Image Processing Support:** conjunto de métodos e técnicas para processamento de imagens digitais;
- **Geospatial Support:** coleção de estruturas e métodos para manipulação, análise e processamento de dados geoespaciais em formato vetorial, incluindo indexação espacial, operadores de conjunto e topológicos, suporte a diferentes fontes de dados e ao padrão de armazenamento OGC/SFS - *Simple Feature Specification* (HERRING et al., 2011);
- **Tracking - Concepts, Structures and Algorithms:** o pacote TATHU, a partir dessa camada, propõe um modelo conceitual para definição e solução do problema de rastreamento e análise do ciclo de vida de SC. As entidades que compõem esse modelo serão descritas na seção 2.1. Basicamente, esse módulo define de modo computacional a metodologia apresentada na Figura 1.3 e implementa métodos para as etapas de detecção, caracterização, acompanhamento e previsão dos elementos de interesse;
- **Input & Output (I.O.):** módulo para leitura e escrita dos dados obtidos a partir do processo de rastreamento dos SC. Este módulo oferece opção para diversos formatos e modos de armazenamento, incluindo diferentes tipos de arquivos (por exemplo, CSV, ESRI Shapefile, KML, GeoJSON, etc.) e banco de dados com suporte espacial (*e.g.* PostGIS e SpatiaLite). As implementações são baseadas em duas interfaces: **Loader** e **Outputter** (Fig. 2.2);
- **Visualization Support:** métodos e ferramentas para visualização de dados geoespaciais no formato de mapas, incluindo exibição de dados vetori-

ais, imagens digitais, controle de estilos e suporte para animações;

- **Tools (CLI and GUI):** esta camada contém aplicativos utilitários que podem ser executados na interface linha de comando (CLI) ou a partir de ambientes gráficos (GUI);
- **Examples:** exemplos de utilização das funcionalidades disponibilizadas pelo TATHU, abrangendo diferentes usuários e modos de operação;
- **Third-party Libraries:** bibliotecas de terceiros auxiliam e fornecem o suporte necessário para a construção do pacote. Por exemplo, bibliotecas para leitura e escrita de imagens, manipulação de dados vetoriais, visualização de gráficos, representação de sistemas de referência espacial, dentre outras;
- **Applications:** última camada do software, representando o maior nível de abstração. Neste módulo estão disponíveis as aplicações que podem ser utilizadas pelos usuários finais para a detecção e análise do ciclo de vida dos SC. Nota-se que as aplicações construídas fazem uso de todo suporte oferecido pelas demais camadas que compõem o pacote.

Figura 2.2 - Módulo para leitura e escrita dos dados obtidos a partir do rastreamento dos SC. Diferentes possibilidades com base na implementação de duas interfaces.



Fonte: Autores.

Vários motivos contribuem para a escolha da linguagem Python para o desenvolvimento do software (LIN, 2012). Trata-se de uma linguagem moderna, interpretada, orientada a objetos e de código aberto, usada em diversos tipos de aplicações. Embora exista há três décadas, seu uso cresceu significativamente nas ciências atmosféricas

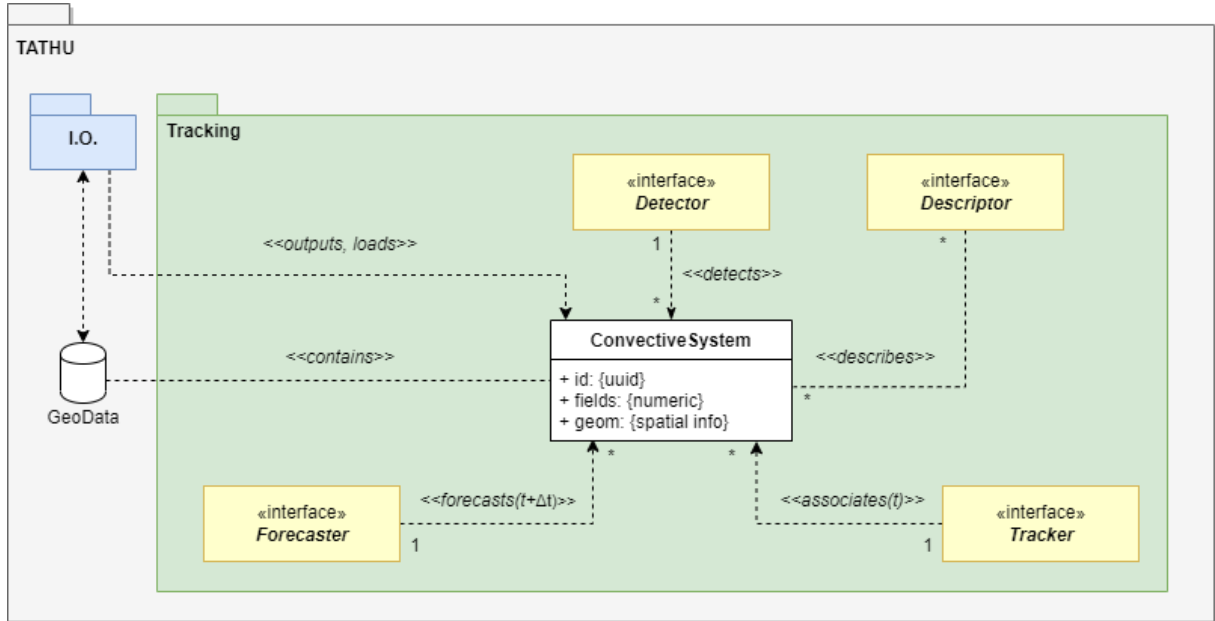
apenas alguns anos atrás, depois que a comunidade de desenvolvimento convergiu para os pacotes científicos padrão (por exemplo, pacotes para manuseio de matrizes), necessários para os processamentos nessa área. Atualmente, a linguagem pode ser vista como uma plataforma de integração robusta para todos os tipos de trabalho em ciências atmosféricas e da Terra, compreendendo desde análise de dados até computação distribuída, interfaces gráficas e SIG - Sistemas de Informações Geográficas. Além disso, a comunidade, formada por uma grande base de usuários e desenvolvedores, na indústria e na ciência, oferece diversos pacotes que podem ser integrados nas aplicações.

2.1 Entidades de Rastreo e Análise

O pacote TATHU propõe, em sua estrutura, um modelo conceitual para definição e solução do problema de rastreo e análise do ciclo de vida de SC. As entidades que compõem esse modelo são apresentadas na Figura 2.3 e melhor detalhadas nas Figura 2.4. Basicamente, uma base de dados geoespaciais contém os elementos de interesse observados, representados então pela classe `ConvectiveSystem`. Esta classe possui um identificador universal único (UUID - *Universally Unique Identifiers*), pelo menos um atributo espacial, *geom*, que indica os limites geográficos do sistema, e *n* outros atributos, *fields*. A partir disso, quatro entidades diferentes são utilizadas:

- **Detector**: interface para detecção dos SC presentes em uma imagem adquirida em determinado instante de tempo. Essa interface recebe como parâmetro a imagem e deve retornar como resultado uma lista de `ConvectiveSystem`. Para cada elemento da lista, o atributo *geom* deve estar definido. Como exemplo, a detecção pode ser realizada a partir de uma operação de limiarização, dando origem ao detector específico `ThresholdDetector`;
- **Descriptor**: interface responsável pela caracterização dos SC. Esta entidade define, para cada sistema, uma lista de atributos descritivos. Por exemplo, o cálculo de atributos estatísticos, como temperaturas média, mínima e máxima - `StatisticalDescriptor`. Ou então, um descritor capaz de verificar e computar a quantidade de células convectivas (*i.e.* regiões muito frias) um SC possui dentro de seus limites - `ConvectiveCellDescriptor`. Em resumo, o método recebe como parâmetros dados auxiliares e um lista de `ConvectiveSystem` para o qual se deseja realizar a caracterização;

Figura 2.3 - Entidades para rastreamento e análise de SC propostas pelo pacote TATHU.

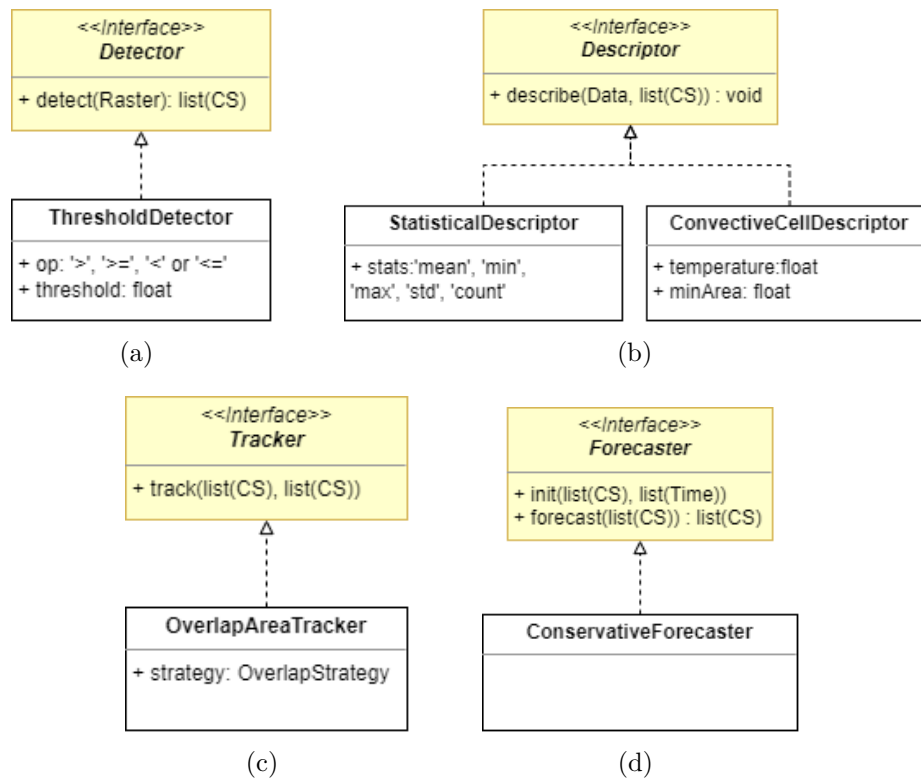


Fonte: Autores.

- **Tracker**: essa interface tem como objetivo realizar o acompanhamento dos SC (*i.e.* associar no tempo as diferentes feições identificadas em cada observação). O método abstrato recebe como parâmetros duas listas contendo objetos `ConvectiveSystem` de instantes de tempos diferentes - *previous* e *current*. Como exemplo, a associação pode ser realizada a partir da relação topológica entre os SC em conjunto com a análise das áreas de interseção. Essa abordagem é representada pela classe concreta `OverlapAreaTracker`;
- **Forecaster**: por último, essa interface é construída com o objetivo de prover previsões para os SC visando instantes de tempo futuros. Uma opção é considerar um movimento conservativo, com base apenas na velocidade atual do sistema. Essa estratégia é implementada na classe `ConservativeForecaster`.

A Figura 2.5 apresenta um pseudocódigo que representa a sequência de operações necessárias para detecção, caracterização, acompanhamento e previsão dos SC utilizando as interfaces abstratas propostas pelo pacote TATHU.

Figura 2.4 - Modelo conceitual para definição do problema de rastreamento e análise do ciclo de vida de SC e exemplos de classes concretas - (a) *Detector*, (b) *Descriptor*, (c) *Tracker* e (d) *Forecaster*.



Fonte: Autores.

2.2 A classe *ConvectiveSystem*

A classe *ConvectiveSystem* representa um SC observado e detectado em um instante de tempo específico. Esta classe possui pelo menos um atributo espacial, *geom*, que indica os limites geográficos do sistema, *n* outros atributos, *fields*, além de um identificador universal único, *uuid*. Nesta classe destaca-se a utilização da **representação vetorial** para a manipulação computacional dos sistemas.

Por exemplo, considere a detecção dos SC a partir de uma operação de limiarização dos valores de temperatura de brilho. De posse do resultado dessa operação, é necessário identificar os componentes conectados. Mais especificamente, cada componente conectado recebe um rótulo e representa um SC. Existem diversos métodos para determinação de agrupamentos de pixels conectados e o resultado é então chamado de **imagem rotulada**. Basicamente, a imagem rotulada representa os diversos SC

Figura 2.5 - Pseudocódigo da sequência de operações necessárias para detecção, caracterização, acompanhamento e previsão dos SC utilizando as interfaces abstratas propostas.

```
1 images = load()
2 previous = None
3 for each image in images:
4     systems = detector.detect(image)
5     descriptor.describe(systems)
6     tracker.track(previous, systems)
7     forecaster.forecast(previous, systems)
8     previous = systems
```

Fonte: Autores.

detectados no instante t . É possível, a partir do percorrimento dessa imagem, obter diversos atributos dos sistemas, dado que são conhecidos seus limites geográficos.

A definição da representação vetorial pode ser obtida a partir da imagem rotulada com uma operação chamada de **vetorização**. Essa representação apresenta algumas melhorias e vantagens quando comparada a representação matricial (*i.e. raster*):

- Possibilidade da indexação espacial dos elementos utilizando estruturas de dados do tipo árvore;
- Aplicação eficiente de operadores de conjunto e topológicos (EGENHOFER; HERRING, 1990) (CLEMENTINI; FELICE, 1995), que possuem relevante importância em aplicações SIG;
- Percorrimento de modo eficiente dos pixels pertencentes a cada SC;
- Capacidade de utilização de transformações geométricas (rotação, escala e translação).

Uma grande vantagem da representação vetorial é a possibilidade da utilização de índices espaciais a partir de estruturas de dados do tipo árvore, por exemplo R-Tree. A indexação tem como principal objetivo possibilitar e otimizar consultas espaciais, podendo ser realizada em memória ou em banco de dados. Por exemplo, com a indexação é possível responder eficientemente questões como:

- *Quais SC interceptam o sistema x?*
- *Quais SC estão dentro de um raio N de distância do sistema x?*

R-Tree (GUTTMAN, 1984) é uma estrutura computacional do tipo árvore para indexação e acesso espacial. Basicamente, a indexação e a busca são realizadas a partir do mínimo retângulo envolvente (do inglês, *minimum bounding rectangle* - MBR) dos elementos de interesse.

É importante observar que com a utilização da indexação espacial e dos operadores de conjunto e topológicos, pode-se realizar eficientemente o acompanhamento dos SC a partir do critério de sobreposição de área mínima.

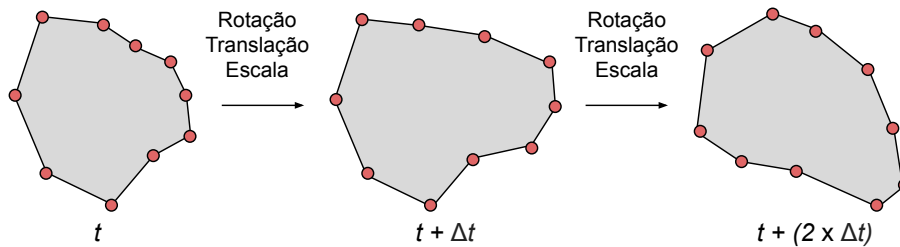
Outra vantagem da utilização da representação vetorial é a existência de padrões abertos especificados para o armazenamento e o intercâmbio desses dados. Como exemplo temos os padrões:

- SFS - *Simple Feature Access* (HERRING et al., 2011), WKT - *Well-Known Text* e WKB - *Well-Known Binary*;
- GML - *Geography Markup Language* (COX et al., 2004);
- WFS - *Web Feature Service* (VRETANOS, 2016);
- Entre outros variados tipos de arquivos, como por exemplo GeoJSON e ESRI Shapefile.

Uma possível aplicação das transformações geométricas é na modificação e deformação da representação vetorial de cada SC. Por exemplo, podem ser utilizadas na etapa de previsão e evolução dos sistemas convectivos, sendo suportável a aplicação de diferentes transformações para cada vértice da representação (Fig. 2.6). Especificamente no caso de SC, a utilização da representação vetorial do tipo POLYGON é a mais adequada.

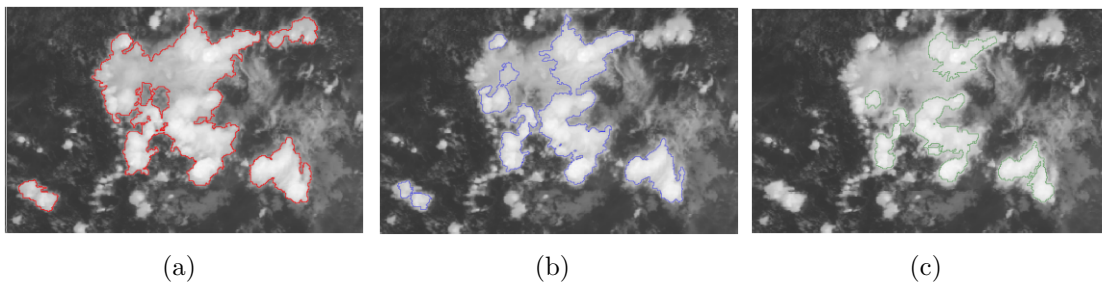
Uma alternativa é a utilização do tipo MULTIPOLYGON. Neste caso, cada polígono que compõe o SC pode ter sido obtido a partir de diferentes limiares de temperatura de brilho. Desse modo, cada sistema teria uma estrutura mais detalhada representando suas variabilidades e respectivos limites espaciais. A Figura 2.7 apresenta o resultado desse processo, denominado de multi-limiarização.

Figura 2.6 - Transformações geométricas (rotação, translação e escala): (a) uma possibilidade é a aplicação para modificar/deformar a representação vetorial de cada SC na previsão e diferentes transformações podem ser utilizadas para cada vértice.



Fonte: Autores.

Figura 2.7 - (a) Multi-limiarização: cada polígono que compõe o SC é obtido a partir de um limiar de temperatura de brilho diferente. (a) < 235 Kelvin, (b) < 220 Kelvin e (c) < 210 Kelvin.

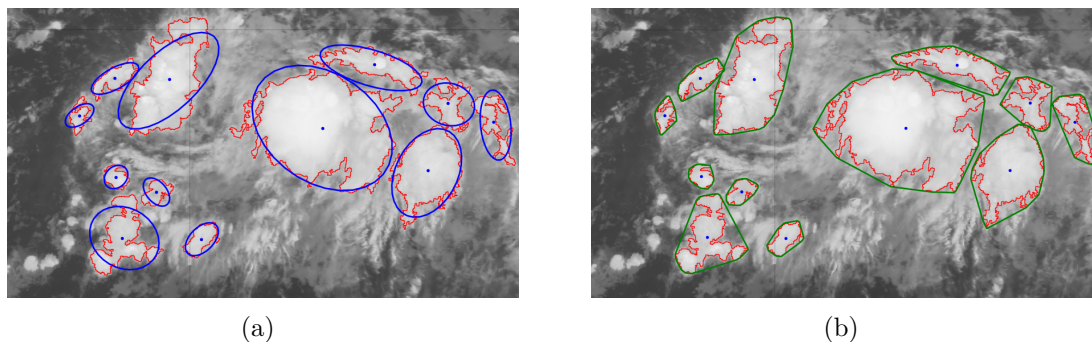


Fonte: Autores.

Outra opção é a utilização de formas de representação diferentes da silhueta original do SC. Por exemplo, cada SC pode ser representado por uma elipse ou pelo seu envoltório convexo (do inglês, *Convex Hull*). A elipse pode ser obtida com a utilização de métodos de ajuste, a partir da representação do tipo POLYGON, considerando as coordenadas de cada vértice. Essas formas de representação alternativas podem ser utilizadas para extrair métricas morfológicas, por exemplo, retangularidade, circularidade, dentre outras.

Por último, na etapa de armazenamento dos SC, é possível a utilização de banco de dados com suporte espacial. Esses bancos possuem a capacidade de armazenar atri-

Figura 2.8 - Formas de representação alternativas: cada SC pode ser representado por formas geométricas diferentes da sua silhueta original (utilizando uma elipse (a) ou o envoltório convexo (b), por exemplo). Desse modo, temos a capacidade de extrair novas métricas morfológicas.



Fonte: Autores.

butos tabulares em conjunto com a representação espacial dos objetos, permitindo a construção de diferentes funcionalidades e modos de consulta. A partir do armazenamento dos resultados nesse tipo de estrutura, consegue-se realizar consultas espaço-temporais de modo bastante eficiente. Como exemplo, considere os seguintes cenários hipotéticos, comuns em estudos e aplicações de rastreo e análise de SC. Todos eles podem ser eficientemente respondidos com esta abordagem.

- *Recuperar os sistemas convectivos detectados nas regiões Sul/Sudeste do Brasil no período de 01/01/2020 até 08/04/2021;*
- *Recuperar os sistemas convectivos detectados na região Norte do Brasil cuja área é maior que $N \text{ km}^2$;*
- *Quantos SC ocorreram em um determinado intervalo de tempo e em determinada região geográfica?*
- *Quantos SC duraram mais de x horas?*
- *Qual o tempo de vida médio dos SC?*
- *Qual SC teve a menor temperatura média e a maior expansão de área?*
- *Qual SC mais se distanciou do local de onde surgiu? Como evoluiu em termos de forma?*

- Qual a taxa de descargas elétricas observadas em um determinado SC, para cada instante de tempo?
- Qual o comportamento do vento nas regiões próximas de um determinado SC, para cada instante de tempo?

2.3 Exemplos de Utilização

Esta seção apresenta trechos de código Python que utilizam e ilustram os conceitos propostos pelo pacote TATHU. Destaca-se que esses exemplos possuem apenas fragmentos mais relevantes. Maiores detalhes sobre as classes e métodos podem ser obtidos na documentação oficial do pacote.

O primeiro código exemplifica a leitura e utilização de um arquivo netCDF contendo os valores medidos pelo sensor ABI a bordo do satélite GOES-16, canal 13, no dia 15 de junho de 2021 - 00:00 UTC. Define-se uma região geográfica de interesse (**extent**) e uma resolução espacial. De posse dessas informações, o remapeamento é realizado da projeção original satélite para uma grade regular, com sistema de coordenadas LatLon/WGS84 (EPSG:4326).

```

1 from tathu.constants import LAT_LON_WGS84
2 from tathu.satellite import goes16
3
4 # Path to netCDF GOES-16 file (IR-window) - ("past")
5 path = './data/goes16/ch13/2021/06/ch13_202106150000.nc'
6
7 # Geographic area of regular grid
8 extent = [-100.0, -56.0, -20.0, 15.0]
9
10 # Grid resolution (kilometers)
11 resolution = 2.0
12
13 # Remap
14 grid = goes16.sat2grid(path, extent, resolution, LAT_LON_WGS84)

```

Exemplo 2.1 - Leitura e remapeamento de dado GOES-16 da projeção original satélite para uma grade regular com resolução espacial de 2 km e sistema de coordenadas LatLon/WGS84.

O próximo trecho apresenta a utilização básica do pacote para detecção dos SC nesse instante de tempo específico, seguido da definição dos atributos estatísticos. Destaca-se o uso dos conceitos **Detector** e **Descriptor**, acessíveis a partir dos módulos importados nas linhas 1 e 2, respectivamente.

```

1 from tathu.tracking import detectors
2 from tathu.tracking import descriptors
3
4 # Threshold value
5 threshold = 230 # Kelvin
6
7 # Define minimum area
8 minarea = 3000 # km^2
9
10 # Create detector
11 detector = detectors.LessThan(threshold, minarea)
12
13 # Detect systems
14 systems = detector.detect(grid)
15
16 # Create statistical descriptor
17 attrs = ['min', 'mean', 'std', 'count']
18 descriptor = descriptors.StatisticalDescriptor(stats=attrs)
19
20 # Describe systems (stats)
21 descriptor.describe(grid, systems)

```

Exemplo 2.2 - Utilização Básica - Detecção em um instante de tempo específico e definição dos atributos estatísticos.

Em seguida, o resultado (*i.e.* `systems`) é exportado para um arquivo de texto CSV. A Figura 2.9 mostra um fragmento do arquivo gerado, em que cada linha representa um SC, o qual possui um identificador único, a data e demais atributos. Neste caso, a representação geométrica pode ser armazenada a partir de uma string WKT. Por exemplo: “POLYGON((-87.7066264194337 14.5002961500494,-87.6886589047003 14.5002961500494, -87.6886589047003 14.4463968410661, ...))”.

```

1 from tathu.io import icsv
2
3 # Create outputter
4 outputter = icsv.Outputter('systems.csv', writeHeader=True)
5
6 # Save to file
7 outputter.output(systems)

```

Exemplo 2.3 - Utilização Básica - Exportação do resultado da detecção e caracterização para um arquivo CSV.

A visualização dos resultados pode ser realizada com base no seguinte trecho de

Figura 2.9 - Fragmento do arquivo CSV gerado. Cada linha representa um SC, o qual possui um identificador único, a data e demais atributos.

```
1 name,timestamp,event,min,mean,count,std
2 c55f99b4-84a4-4b4b-8393-25aaaf85fb75,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,201.8952178955
3 dc616f08-e0cd-4a15-87ed-7becc5ab253a,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,201.8952178955
4 ed97d8cc-d4e7-4a52-b686-1763bd0281f1,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,196.6716918945
5 e57dccdf-cf36-4f41-9160-840f29a9111e,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,218.9177856445
6 37f1de1a-871b-4b5d-b971-48ddb84cd1ed,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,203.6773681640
7 32a42b28-74a1-4221-912e-c401d9051c88,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,191.3252563476
8 5eaca958-6816-4311-b85c-5c595465b416,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,214.4317016601
9 53085a82-6c74-40f6-b415-8aaaf8c7a45c,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,204.6606140136
10 154db4b4-202b-43b9-afb1-3c3957c2d1a6,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,194.2135620117
11 44cf29aa-d685-484d-b841-ecbc6cd5349f,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,200.4203491210
12 7bc01815-b359-4734-b2d5-393247c193e7,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,197.7778472900
13 1f958162-c676-47e4-bb86-439f6ca6553b,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,196.8560485839
14 9a69280d-d2c8-4bca-ab6f-6112edc45f1b,2022-06-15 00:00:20.400000,SPONTANEOUS_GENERATION,210.1299743652
```

Fonte: Autores.

código e o resultado é apresentado na Figura 2.10.

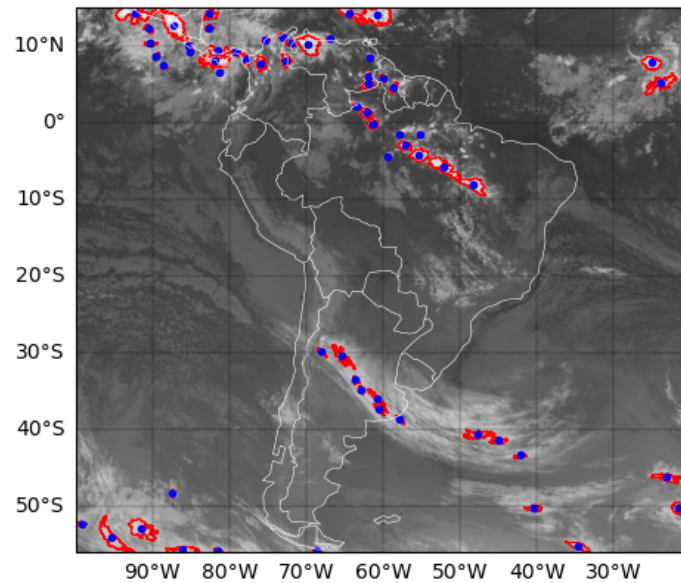
```
1 from tathu.tracking.visualizer import MapView
2
3 # Create MapView component
4 m = MapView(extent)
5
6 # Plot grid
7 m.plotImage(grid, cmap='Greys', vmin=180.0, vmax=320.0)
8
9 # Plot systems
10 m.plotSystems(systems, edgecolor='red', centroids=True)
11
12 # Show GUI result
13 m.show()
```

Exemplo 2.4 - Visualização dos SC detectados.

Para exemplificar a capacidade modular do pacote, o trecho a seguir mostra como é possível exportar o mesmo resultado para uma uma instância de banco de dados com suporte espacial, no caso o SpatiaLite. Essa tarefa pode ser realizada utilizando-se uma outra implementação concreta do módulo Input & Output.

```
1 from tathu.io import spatialite
2
3 # New outputter
4 database = spatialite.Outputter('systems.sqlite', 'systems')
5
```

Figura 2.10 - Exemplo de visualização dos SC detectados em um instante de tempo específico.



Fonte: Autores.

```
6 # Now, save to database
7 database.output(systems)
```

Exemplo 2.5 - Exportação do resultado para uma instância de banco de dados SpatiaLite.

Uma vez detectados os SC presentes na imagem do dia 15 de junho de 2021 - 00:00 UTC, é possível agora a realização da etapa de acompanhamento. Isto é feito a partir de uma nova imagem, do mesmo dia, porém no horário 00:10 UTC. Destaca-se o uso do conceito Tracker, acessível a partir do módulo importado na linha 13.

```
1 # Path to new netCDF GOES-16 file - ("present")
2 path = './data/goes16/ch13/2021/06/ch13_202106150010.nc'
3
4 # Remap
5 grid = goes16.sat2grid(path, extent, resolution, LAT_LON_WGS84)
6
7 # Tracking
8 previous = systems
9
10 # Detect new systems
11 systems = detector.detect(grid)
12
```

```

13 from tathu.tracking import trackers
14
15 # Define overlap area criterion
16 overlapAreaCriterion = 0.1 # 10%
17
18 # Create overlap area strategy
19 strategy = trackers.RelativeOverlapAreaStrategy(
20     overlapAreaCriterion)
21
22 # Create tracker entity
23 t = trackers.OverlapAreaTracker(previous, strategy=strategy)
24 t.track(current)
25
26 # Save to database
27 database.output(systems)

```

Exemplo 2.6 - Acompanhamento dos SC a partir de uma nova imagem.

Por último, fechando a etapa de acompanhamento e análise, a previsão dos SC para instantes futuros é realizada com base no seguinte fragmento de código:

```

1 from tathu.tracking import forecasters
2
3 # Define times
4 times = [15, 30, 45, 60, 90, 120] # minutes
5
6 # Forecaster entity
7 f = forecasters.Conservative(previous, intervals=times)
8
9 # Forecast result for each time
10 forecasts = f.forecast(current)

```

Exemplo 2.7 - Previsão dos SC para instantes futuros. Por exemplo

Considerando que os diferentes SC foram detectados e armazenados ao longo do tempo, a recuperação desses objetos de interesse pode ser realizada com base no seguinte trecho de código:

```

1 from tathu.io import spatialite
2
3 # Setup informations to load systems from database
4 dbname = 'systems.sqlite'
5 table = 'systems'
6
7 # Connect to database
8 db = spatialite.Loader(dbname, table)

```

```

9
10 # Get all-systems name
11 names = db.loadNames()
12
13 # Load first system, geometry and attributes
14 family = db.load(names[0], ['min', 'mean', 'std', 'count'])

```

Exemplo 2.8 - Recuperação do SC armazenados.

Outros métodos podem ser utilizados para recuperação dos SC de modo mais eficiente, por exemplos: a partir do tempo de duração, considerando um dia ou um intervalo de datas, com base em uma restrição espacial, entre outros. Para casos mais específicos, é possível ainda realizar uma consulta diretamente a fonte de armazenamento utilizando linguagem SQL.

```

1 # Load SC with life-cycle time-duration >= 10 hours
2 systems = db.loadByDuration(10, operator='>=')
3
4 # Load SC with life-cycle time-duration < 1 hours
5 systems = db.loadByDuration(1, operator='<')
6
7 # Load SC from day 26/06/2021
8 systems = db.loadByDay('20210626')
9
10 # Load SC using SQL query
11 systems = db.query('generic query example')

```

Exemplo 2.9 - Exemplos de recuperação de SC a partir de consultas a fonte de armazenamento.

O ciclo de vida do SC pode ser visualizado a partir do suporte oferecido pelo pacote, seguindo o exemplo do trecho de código apresentado abaixo. O resultado é ilustrado pela Figura 2.11, em que cada gráfico representa um instante de tempo no ciclo de vida dos sistemas.

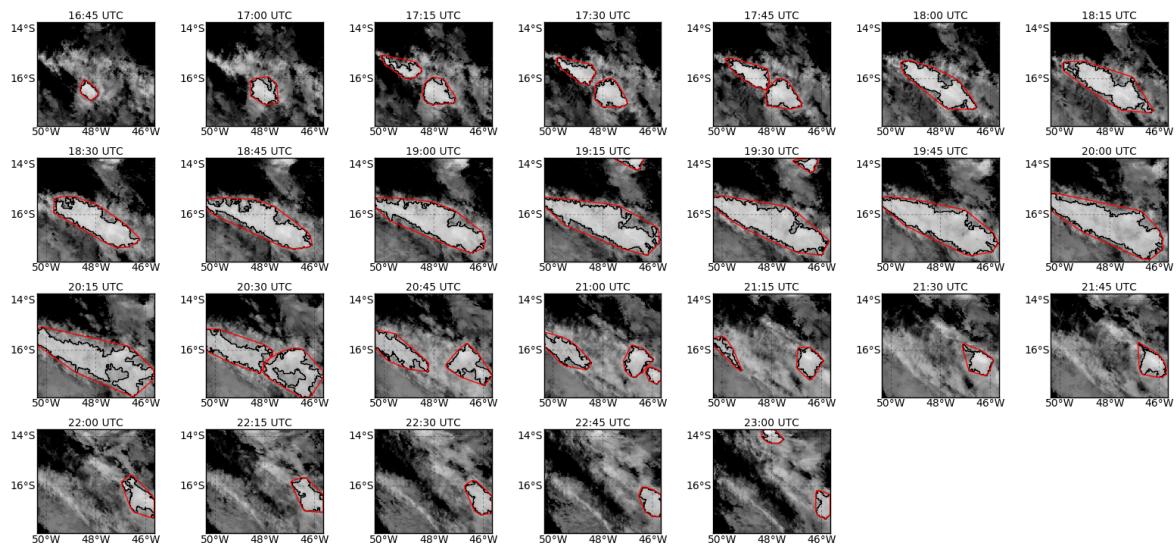
```

1 from tathu.tracking import visualizer
2
3 # Visualize life-cycle
4 view = visualizer.SystemHistoryView(family)
5 view.show()

```

Exemplo 2.10 - Visualização do ciclo de vida de SC.

Figura 2.11 - Exemplo de visualização do ciclo de vida de SC a partir do suporte oferecido pelo pacote TATHU.



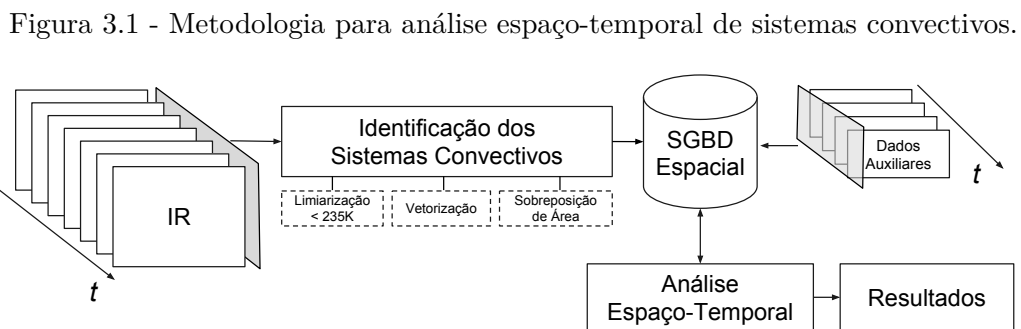
Fonte: Autores.

3 EXPERIMENTOS E RESULTADOS

Este capítulo apresenta alguns estudos de caso, experimentos e os resultados obtidos com a aplicação dos conceitos propostos e a utilização do software TATHU.

3.1 Primeiro Experimento

A metodologia do primeiro experimento é apresentada na Figura 3.1. A partir de uma série temporal de imagens de satélite, o primeiro passo consiste na identificação dos SC. A identificação é realizada utilizando-se uma operação de limiarização dos valores de temperatura de brilho. Após a limiarização, os componentes conectados são rotulados, passam por uma avaliação de tamanho mínimo e definem os SC. Um processo de vetorização é aplicado em cada sistema (Fig. 3.2) e o critério de sobreposição de área mínima empregado de modo a associar os elementos no tempo.

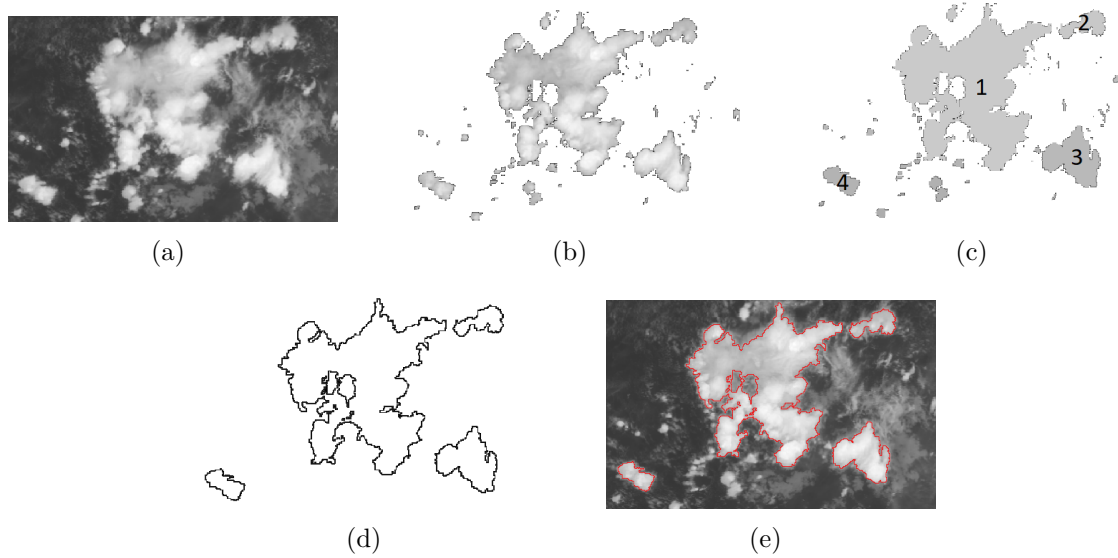


Fonte: Autores.

A etapa seguinte consiste no armazenamento das informações produzidas em um SGBD com suporte espacial. Dados auxiliares no tempo também devem ser armazenados, de modo a prover informações complementares durante as análises espaço-temporais. São exemplos de dados auxiliares: ocorrência de descargas atmosféricas, informações de vento estimadas em imagens de satélites, dados de pluviômetros, etc. O último passo consiste na análise dos sistemas a partir de operadores espaciais e da linguagem de consulta SQL fornecidos pelo SGBD.

Para realização dos experimentos, foram utilizados dados adquiridos pelo satélite geoestacionário GOES-16. O satélite foi lançado em 19 de novembro de 2016 e está posicionado atualmente sobre a Linha do Equador, na longitude -75° . A área de estudo compreende o Brasil, demais países da América do Sul e América Central.

Figura 3.2 - Identificação dos SC: (a) imagem infravermelho termal, (b) resultado do processo de limiarização, (c) componentes conectados e (d) vetorização. Em (e), a representação vetorial é apresentada em conjunto com a imagem de satélite.



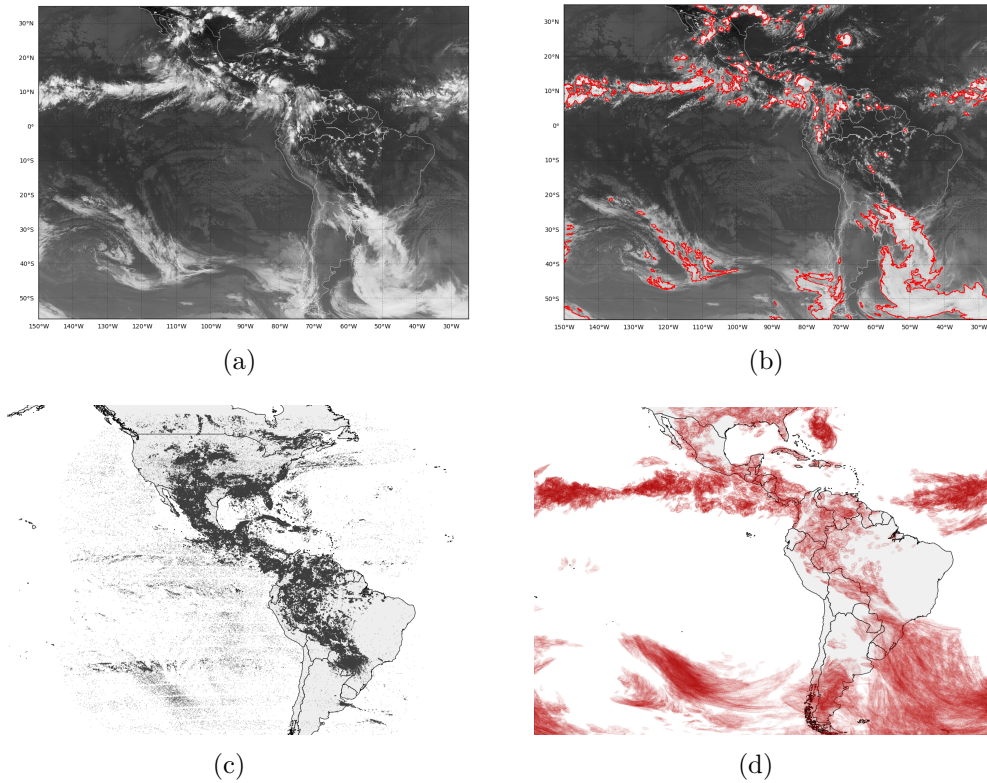
Fonte: Autores.

3.1.1 Dados

Na identificação dos SC, utilizou-se imagens do canal 13 ($10,3 \mu m$), com resolução espacial de 2 km e temporal de 15 minutos. As imagens foram adquiridas nos dias 13, 14 e 15 de agosto de 2017, totalizando 288 imagens. Todas as imagens foram mapeadas para uma grade regular, no sistema de referência espacial EPSG:4326, tendo como limites geográficos as coordenadas: ($150^{\circ}O$, $56^{\circ}S$) e ($25^{\circ}L$, $35^{\circ}N$). Como exemplo, a Figura 3.3.a apresenta a imagem adquirida em 13/08/2017 - 00:00 UTC. Os SC foram definidos a partir do limiar 235K e área mínima de 3000 km^2 . Para acompanhamento, utilizou-se o valor de 1200 km^2 como critério de mínima sobreposição. As representações vetoriais e matriciais de cada SC identificado foram armazenadas em um banco de dados PostgreSQL com a extensão espacial PostGIS habilitada. A Figura 3.3.b mostra os SC definidos para o dia 13/08/2017 - 00:00 UTC.

Como dados auxiliares, utilizou-se informações obtidas no mesmo período pelo sensor GLM - *Geostationary Lightning Mapper* do GOES-16. O GLM é o primeiro sensor para observação de descargas atmosféricas a bordo de um satélite geostacionário. No total, foram processados 2.204 arquivos, contendo para os 3 dias 3.307.460 ocorrências de flashes (aprox. $13 \text{ flashes s}^{-1}$) (Fig. 3.3.c). As localizações de cada ocorrência (ponto) foram armazenadas no banco de dados espacial.

Figura 3.3 - (a) Imagem adquirida pelo GOES-16 em 13 de agosto de 2017 - 00:00 UTC: canal 13 ($10,3 \mu\text{m}$) e resolução espacial de 2 km, (b) SC identificados neste mesmo horário e (c) descargas atmosféricas observadas pelo sensor GLM/GOES-16 nos dias 13, 14 e 15 de agosto de 2017. Cada ocorrência está representada por 1 pixel. Em (d), todos os polígonos de cada SC detectado no período são apresentados (contornos).



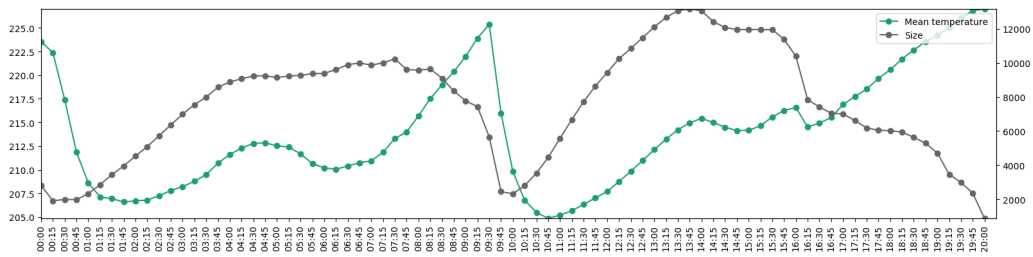
Fonte: Autores.

3.1.2 Análises

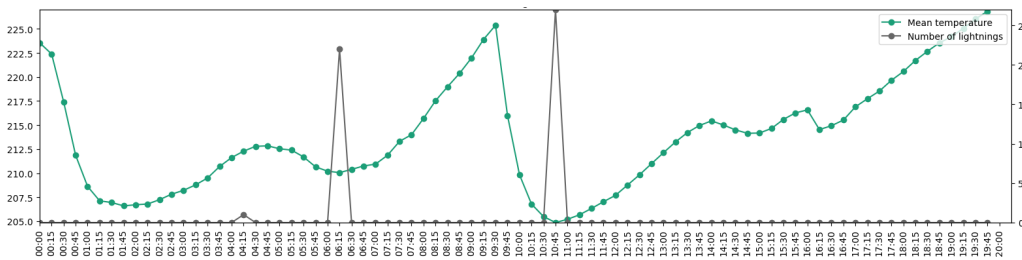
As análises foram realizadas a partir dos operadores espaciais e da linguagem SQL, ambos fornecidos pelo SGBD. No total, foram detectados 5.797 sistemas, com um tempo de vida médio de 01:22:31 horas. Foram gerados 36.508 polígonos (Fig. 3.3.d). 704 SC apresentaram duração maior que 3 horas. Apenas 83 SC tiveram duração maior que 12 horas. O SC com maior tempo de vida foi de 2 dias 23:45:00 horas. 2.218 SC (38%) foram detectados em apenas uma única imagem, ou seja, sofreram Dissipação Natural, por *Split* ou *Merge* entre duas observações seguidas. A Figura 3.4 apresenta gráficos da evolução de um determinado SC com duração de 20 horas em relação a dois atributos: (a) temperatura média e tamanho e (b) temperatura média e o número de descargas atmosféricas observadas no SC. Nesta análise, foi realizada

uma consulta com restrição espaço-temporal entre as representações geométricas do SC (polígono) e das observações de descargas atmosféricas (pontos) (Fig. 3.5). É possível observar um pico de descargas acompanhado por uma diminuição da temperatura média do SC (10:45 UTC). Também é possível observar que o sistema passou por dois ciclos de tamanho: o primeiro no intervalo entre 00:00 e 09:45 UTC e o segundo entre 10:00 e 20:00 UTC.

Figura 3.4 - Evolução de um determinado SC com duração de 20 horas: (a) temperatura média e tamanho e (b) temperatura média e descargas atmosféricas dentro do SC. Em (b), utilizou-se uma consulta com restrição espacial e temporal.



(a)



(b)

Fonte: Autores.

Em resumo, os sistemas foram detectados automaticamente em cada imagem a partir de um limiar de temperatura frio (235K) e associados no tempo utilizando um critério espacial de sobreposição de área mínima. Neste caso, a representação vetorial de cada SC foi utilizada, combinando um operador de conjunto (**Intersection**) e métodos de indexação e recuperação espacial (**R-Tree**). Como exemplo, o suporte fornecido pelo SGBD foi utilizado para analisar as descargas atmosféricas em cada instante de tempo no ciclo de vida do SC. É possível notar as vantagens da utilização de ferramentas geo-computacionais nesse tipo de aplicação.

Figura 3.5 - Consulta espaço-temporal para determinar as descargas atmosféricas em cada instante do ciclo de vida de um SC x .

```
SELECT lightnings.id, ST_AsBinary(lightnings.geom)
FROM lightnings, convective_systems
WHERE ST_Intersects(lightnings.geom, convective_systems.geom)
AND convective_systems.name = 'x'
AND (lightnings.start_date_time - convective_systems.date_time) < '00:07:30'
AND (lightnings.start_date_time - convective_systems.date_time) > '-00:07:30'
ORDER BY lightnings.start_date_time;
```

Fonte: Autores.

3.2 Segundo Experimento

O segundo experimento utiliza a mesma metodologia apresentada na Figura 3.1. Entretanto, a limiarização é aplicada sobre valores de refletividade obtidos a partir de dados dos radares meteorológicos.

3.2.1 Dados

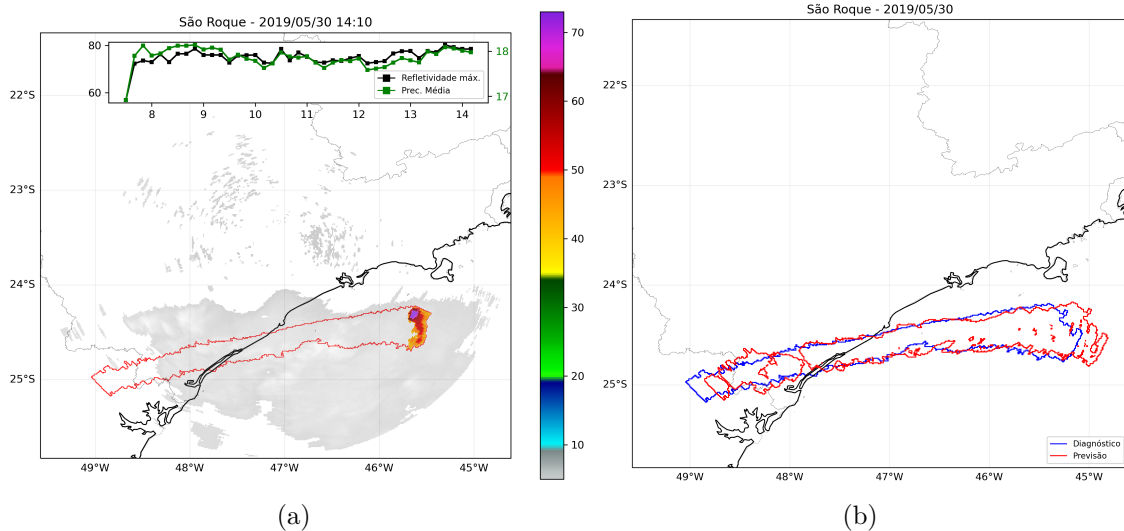
Para identificação dos SC, utilizou-se informações de refletividade do produto CAPPI - *Constant Altitude Plan Position Indicator* na altura de 3 km, com resolução espacial de 1 km e temporal de 10 minutos do radar localizado em São Roque - SP (-21, 34°, -49, 58°), pertencente ao DECEA - Departamento de Controle do Espaço Aéreo. Os dados estão em grade regular com 500 x 500 pontos latitudinais e longitudinais. Os SC foram definidos a partir do limiar 35 dBZ e área mínima de 9 km². Para acompanhamento, utilizou-se o valor de 10 km² como critério de mínima sobreposição.

3.2.2 Análises

As análises foram realizadas a partir dos operadores espaciais e da linguagem SQL, ambos fornecidos pelo SGBD. A Figura 3.6.a ilustra a trajetória (polígono vermelho) de uma família de SC que foi identificada utilizando-se o limiar de 35 dBZ para o radar de São Roque no dia 30 de maio de 2019. O primeiro SC da família foi identificado no horário das 07:30 UTC e o último registro foi às 16:10 UTC, quando atingiu o limite da área de cobertura do radar, totalizando 59 imagens (9 horas e 40 minutos). A área sombreada mostra as regiões onde há informações de refletividade menores que 35 dBz. Na parte superior da Figura 3.6.a, tem-se um gráfico da evolução do SC com os valores máximos de refletividade (linha preta) e a precipitação

média estimada (linha verde) do SC para cada passo de tempo. É possível observar que o SC apresentou valores altos de refletividade na maior parte da trajetória desde sua identificação até o final do rastreamento, quando o sistema atingiu o limite do radar. Valores altos de refletividade, especialmente na região com valores superiores a 65 dBZ, poderiam sugerir precipitação de granizo. A Figura 3.6.b apresenta a trajetória dos SC representados pela união dos polígonos de cada cluster, tanto para os clusters diagnósticos (azul), para cada horário de observação do radar, como para as previsões de deslocamento considerando um intervalo de 30 minutos (vermelho). Nota-se que, ao menos para casos de continuidade do SC, considerar movimento conservativos apresenta bons resultados.

Figura 3.6 - (a) Trajetória (polígono vermelho) do SC com os valores máximos de refletividade (linha preta) e a precipitação média estimada (linha verde) do SC para cada passo de tempo. Área sombreada representa regiões onde há informações de refletividade menores que 35 dBZ (b) Trajetória dos SC representados pela união dos polígonos de cada cluster: diagnóstico (azul) e previsões de deslocamento (vermelho), considerando um intervalo de 30 minutos.



Fonte: Autores.

3.3 Terceiro Experimento

Este experimento tratou da construção de redes geográficas com base em eventos de precipitação usando dados de radar meteorológico. Os eventos considerados ocorreram de janeiro a março de 2019, com foco na Área Metropolitana de São Paulo (MASP). Com base neste conjunto de eventos, foram analisadas as relações entre as

propriedades meteorológicas e as métricas topológicas das redes correspondentes. Os resultados mostram correlações significativas e algumas particularidades ao analisar grupos de eventos específicos. Para identificação dos eventos, o pacote de software TATHU foi utilizado. Mais detalhes sobre esse experimento podem ser verificados em (JORGE et al., 2022).

4 CONCLUSÃO

Este documento apresentou um pacote de software denominado TATHU - *Tracking and Analysis of Thunderstorms*, planejado e implementado para permitir o rastreamento automático e a análise do ciclo de vida de SC em sequências de imagens de satélites e campos meteorológicos como, por exemplo, ecos de radares ou variáveis de modelos de previsão numérica de tempo. Destacou-se a apresentação do problema do ponto de vista computacional, em conjunto com o apontamento de técnicas e métodos que podem auxiliar na sua resolução. A motivação principal é contribuir para o aprimoramento dos processos e ferramentas já existentes no INPE nesse contexto.

Como destaque, a solução propõe a utilização de técnicas de Geoinformática e banco de dados espaciais para auxiliar a análise computacional dos sistemas meteorológicos, além de uma arquitetura de software extensível que permite a adição de novas funcionalidades e melhorias na ferramenta de modo eficiente.

Outro ponto importante é a apresentação de um modelo conceitual para definição e solução do problema a partir da utilização de quatro interfaces abstratas: *Detector*, *Descriptor*, *Tracker* e *Forecaster*, e suas respectivas implementações concretas.

TATHU é implementado em Python e distribuído como um pacote dessa linguagem, com código fonte aberto, multiplataforma e livre. O versionamento e acesso é realizado a partir de um repositório Git, disponível no endereço <https://github.com/uba/tathu>. Algumas características incluem:

- Público-alvo diversificado, incluindo desenvolvedores de software e usuários finais;
- Entrega uma base tecnológica comum, facilitando o uso por bolsistas, alunos, pesquisadores e instituições parceiras; *i.e.* os esforços computacionais e novos desenvolvimentos, específicos de cada contexto, podem ser avaliados e partes comuns serem direcionadas para aumentar a cartela de funcionalidades do pacote. Em resumo, esforços de desenvolvimento podem ser melhor gerenciados e aproveitados;
- Construção colaborativa, integrando pesquisa e desenvolvimento tecnológico;
- Extensível e inclusivo, permitindo a utilização de bibliotecas de terceiros para contribuir na construção das funcionalidades.

Vários testes com o pacote estão em andamento. Ainda há a necessidade de determinar as limitações do software em termos de volume de dados que ele pode processar de modo eficiente, além da capacidade de extensibilidade para diversas aplicações e cenários. No entanto, os resultados preliminares obtidos a partir dos experimentos são bastante promissores. Novos estudos devem ser realizados para que a metodologia e o software evoluam interdisciplinarmente, conectando Ciência da Computação, Meteorologia e demais áreas correlatas.

Por último, espera-se com a publicação do código-fonte, que o pacote seja utilizado por diferentes instituições parceiras, nacionais e internacionais, interessadas em avaliar e contribuir com o desenvolvimento de novas funcionalidades.

Este trabalho foi conduzido pela Divisão de Satélites e Sensores Meteorológicos - DISSM e Divisão de Previsão de Tempo e Clima - DIPTC, da Coordenação-Geral de Ciências da Terra - CGCT/INPE.

REFERÊNCIAS BIBLIOGRÁFICAS

- CLEMENTINI, E.; FELICE, P. D. A comparison of methods for representing topological relationships. **Information sciences-applications**, Elsevier, v. 3, n. 3, p. 149–178, 1995. 17
- COX, S.; DAISEY, P.; LAKE, R.; PORTELE, C.; WHITESIDE, A. Geography markup language (gml) encoding specification v3.1.1. 02 2004. 18
- CRANE, R. K. Automatic cell detection and tracking. **IEEE Transactions on Geoscience Electronics**, v. 17, n. 4, p. 250–262, 1979. 2
- DIXON, M.; WIENER, G. Titan: Thunderstorm identification, tracking, analysis, and nowcasting a radar-based methodology. **Journal of Atmospheric and Oceanic Technology**, v. 10, p. 785–797, 1993. 2
- EGENHOFER, M. J.; HERRING, J. Categorizing binary topological relations between regions, lines, and points in geographic databases. **Technical Report, Department of Surveying Engineering, University of Maine**, v. 9, n. 94-1, p. 76, 1990. 17
- FERREIRA, K. R. **An algebra for spatiotemporal data: from observations to events**. 120 p. Tese (Doutorado) — Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 2012-11-28 2012. Disponível em: <<http://urlib.net/sid.inpe.br/mtc-m19/2012/12.11.16.43>>. Acesso em: 22 ago. 2017. 7
- FERREIRA, K. R.; QUEIROZ, G. R.; VINHAS, L.; MARUJO, R. F. B.; SIMOES, R. E. O.; PICOLI, M. C. A.; CAMARA, G.; CARTAXO, R.; GOMES, V. C. F.; SANTOS, L. A.; SANCHEZ, A. H.; ARCANJO, J. S.; FRONZA, J. G.; NORONHA, C. A.; COSTA, R. W.; ZAGLIA, M. C.; ZIOTI, F.; KORTING, T. S.; SOARES, A. R.; CHAVES, M. E. D.; FONSECA, L. M. G. Earth observation data cubes for brazil: Requirements, methodology and products. **Remote Sensing**, v. 12, n. 24, 2020. ISSN 2072-4292. Disponível em: <<https://www.mdpi.com/2072-4292/12/24/4033>>. 7
- FIOLLEAU, T.; ROCA, R. An algorithm for the detection and tracking of tropical mesoscale convective systems using infrared images from geostationary satellite. **IEEE transactions on Geoscience and Remote Sensing**, IEEE, v. 51, n. 7, p. 4302–4315, 2013. 2, 5

GUTTMAN, A. R-trees: A dynamic index structure for spatial searching. **SIGMOD Rec.**, Association for Computing Machinery, New York, NY, USA, v. 14, n. 2, p. 4757, jun 1984. ISSN 0163-5808. Disponível em: <<https://doi.org/10.1145/971697.602266>>. 18

HERRING, J. et al. Opengis® implementation standard for geographic information - simple feature access - part 1: Common architecture [corrigendum]. Open Geospatial Consortium, 2011. 12, 18

IPCC. **Managing the Risks of Extreme Events and Disasters to Advance Climate Change Adaption**. [S.l.]: Cambridge University Press, 2012. 512 p. 2

JORGE, A. A. S.; UBA, D.; FERNANDES, A. A.; COSTA, I. C.; SANTOS, L. B. L. **Precipitation event-based networks: an analysis of the relations between network metrics and meteorological properties**. arXiv, 2022. Disponível em: <<https://arxiv.org/abs/2206.11339>>. 35

LIN, J. W.-B. Why python is the next wave in earth sciences computing. **Bulletin of the American Meteorological Society**, American Meteorological Society, Boston MA, USA, v. 93, n. 12, p. 1823 – 1824, 2012. Disponível em: <<https://journals.ametsoc.org/view/journals/bams/93/12/bams-d-12-00148.1.xml>>. 13

MACHADO, L.; GUEDES, R.; SILVEIRA, J.; WALTZ, R.; ALVES, M. Ciclo de vida de sistemas convectivos. In: **VIII Congresso Brasileiro de Meteorologia, SBMET, Anais**. [S.l.: s.n.], 1994. v. 2, p. 323–326. 2

MACHADO, L. A. T.; LAURENT, H. The convective system area expansion over amazonia and its relationships with convective system life duration and high-level wind divergence. **Monthly Weather Review**, v. 132, n. 3, p. 714–725, 2004. 2

MAKRIS, A.; PRIEUR, C. Bayesian multiple-hypothesis tracking of merging and splitting targets. **IEEE Transactions on Geoscience and Remote Sensing**, IEEE, v. 52, n. 12, p. 7684–7694, 2014. 2

VELASCO, I.; FRITSCH, J. M. Mesoscale convective complexes in the americas. **Journal of Geophysical Research: Atmospheres**, Wiley Online Library, v. 92, n. D8, p. 9591–9613, 1987. 2

VILA, D. A.; MACHADO, L. A. T.; LAURENT, H.; VELASCO, I. Forecast and tracking the evolution of cloud clusters (fortracc) using satellite infrared imagery:

Methodology and validation. **Weather and Forecasting**, v. 23, n. 2, p. 233–245, 2008. 2, 5, 6

VRETANOS, P. A. Web feature service implementation specification with corrigendum. In: . [S.l.]: Open Geospatial Consortium, 2016. 18

PUBLICAÇÕES TÉCNICO-CIENTÍFICAS EDITADAS PELO INPE

Teses e Dissertações (TDI)

Teses e Dissertações apresentadas nos Cursos de Pós-Graduação do INPE.

Manuais Técnicos (MAN)

São publicações de caráter técnico que incluem normas, procedimentos, instruções e orientações.

Notas Técnico-Científicas (NTC)

Incluem resultados preliminares de pesquisa, descrição de equipamentos, descrição e ou documentação de programas de computador, descrição de sistemas e experimentos, apresentação de testes, dados, atlas, e documentação de projetos de engenharia.

Relatórios de Pesquisa (RPQ)

Reportam resultados ou progressos de pesquisas tanto de natureza técnica quanto científica, cujo nível seja compatível com o de uma publicação em periódico nacional ou internacional.

Propostas e Relatórios de Projetos (PRP)

São propostas de projetos técnico-científicos e relatórios de acompanhamento de projetos, atividades e convênios.

Publicações Didáticas (PUD)

Incluem apostilas, notas de aula e manuais didáticos.

Publicações Seriadas

São os seriados técnico-científicos: boletins, periódicos, anuários e anais de eventos (simpósios e congressos). Contam destas publicações o Internacional Standard Serial Number (ISSN), que é um código único e definitivo para identificação de títulos de seriados.

Programas de Computador (PDC)

São a seqüência de instruções ou códigos, expressos em uma linguagem de programação compilada ou interpretada, a ser executada por um computador para alcançar um determinado objetivo. Aceitam-se tanto programas fonte quanto os executáveis.

Pré-publicações (PRE)

Todos os artigos publicados em periódicos, anais e como capítulos de livros.