



**ESCOLA SALESIANA SÃO JOSÉ**  
**CENTRO PROFISSIONAL DOM BOSCO**

Gustavo Banaco Silva  
Paulo David Archanjo  
Vitor Hugo Pires dos Santos

**CÁLCULO DE DANOS FLORESTAIS UTILIZANDO VISÃO  
COMPUTACIONAL**

CAMPINAS

2021

Gustavo Banaco Silva  
Paulo David Archanjo  
Vitor Hugo Pires dos Santos

## **CÁLCULO DE DANOS FLORESTAIS UTILIZANDO VISÃO COMPUTACIONAL**

Trabalho de Conclusão de  
Curso apresentado como requisito  
para obtenção do título de Técnico de  
Informática, concluído na Escola  
Salesiana São José, Centro  
Profissional Dom Bosco.

Orientadora: Adriana Maia da Silva Coelho

Coorientador: Daniel Rinaldi Mendonça

CAMPINAS

2021

*Dedicamos este trabalho ao Fernando Caro Menardi,  
pela solicitude e assistência.*

## **AGRADECIMENTOS**

Agradecemos, primeiramente, a Deus, que nos possibilitou a realizar este trabalho.

A todos os nossos professores do curso de Informática da Escola Salesiana São José pela excelência da qualidade técnica de cada um.

Em especial a nossa orientadora Adriana Maia da Silva Coelho por aceitar conduzir o nosso trabalho com total dedicação, interesse e apoio, deixando uma contribuição extremamente importante e positiva.

Somos gratos aos nossos pais por sempre nos incentivarem e acreditarem que seríamos capazes de superar os obstáculos que a vida nos apresentou.

Cada membro do grupo agradece ao outro pela parceria, comprometimento, compreensão e acima de tudo, a amizade.

*“Um ser humano deve transformar informação em inteligência ou conhecimento. Tendemos a esquecer que nenhum computador jamais fará uma nova pergunta.”. Grace Hopper*

## RESUMO

Atualmente na área ambiental, é comum que as instituições promovam pesquisas sobre áreas que sofreram algum dano seja por desmatamento ou incêndio. Estas pesquisas têm o intuito de levantar os dados e possuir uma visão macro daquele ambiente. Entretanto, estas instituições acabam tendo problemas com verba, visto que as pesquisas acarretam um custo elevado, e consequentemente acabam gerando problemas, os quais poderiam ser facilmente evitados através do uso de softwares inteligentes. Tendo isso em mente, foi utilizado o método de engenharia para construir uma solução viável partindo deste problema, sendo assim, o projeto contará com funções que irão facilitar o dia a dia das instituições ambientais, tornando capaz de realizar cálculos mais rápidos da área desejada utilizando imagens de satélite. Isso é possível graças ao método denominado “Binarização (Thresholding)”, onde as imagens são convertidas em uma escala de cinza, facilitando a realização do cálculo pelo código, que pós processamento terá a área devastada/incendiada em m<sup>2</sup>. Sendo assim, o objetivo geral é automatizar o trabalho das organizações responsáveis, economizando tempo, esforço intelectual e também eliminando possíveis erros matemáticos, tudo por meio de um software voltado aos cálculos sobre danos florestais.

**Palavras Chaves:** Binarização, Devastação, Software.

## **ABSTRACT**

Currently in the environmental area, it is common for institutions to promote research on areas that have suffered some damage, whether by deforestation or fire. These surveys are intended to gather data and have a macro view of that environment. However, these institutions end up having problems with funding, as research entails a high cost. Consequently, they end up generating problems, which can be easily avoided by using an intelligent software. Keeping this in mind, the project will feature functions that will facilitate the day-to-day activities of environmental institutions, making it capable of performing faster calculations of the desired area using satellite images. This is possible thanks to the method called "Binarization (Thresholding)", where the images are converted into a gray scale, facilitating the calculation by the code, which after processing will have a devastated/burned area in m<sup>2</sup>. Thus, the general objective is to automate the work of responsible organizations, saving time, intellectual effort and also eliminating possible mathematical errors, all through software aimed at calculating forest damage.

**Key words:** Binarization, Devastation, Software.

## LISTA DE FIGURAS

Figura 1: Exemplo de Clustering .....	8
Figura 2: Página inicial do site.....	10
Figura 3: Criando conexão ao Banco de Dados com JS e MySQL .....	11
Figura 4: Código em PHP para armazenamento de fotos .....	12
Figura 5: Registro da imagem no banco de dados.....	12
Figura 6: Estrutura do Banco de Dados e relacionamentos das tabelas .....	13
Figura 7: Representação do micro framework Flask .....	16
Figura 8: Exemplo de ações - Métodos HTTP.....	17
Figura 9: Exemplo de código em JSON (.json) .....	18
Figura 10: Fórmula Geral da Binarização.....	19
Figura 11: BINARY_INV – Exemplo de Uso 1 .....	19
Figura 12: BINARY_INV - Exemplo de Uso 2.....	19
Figura 13: BINARY_INV - Exemplo de Uso 3.....	20
Figura 14: Adquirindo a imagem .....	20
Figura 15: Rotacionando as imagens .....	21
Figura 16: Armazenamento das imagens em um Diretório .....	21
Figura 17: Resultado do algoritmo .....	21
Figura 18: Fluxograma .....	22
Figura 19: 2ª seção do site.....	24
Figura 20: 3ª seção do site.....	24
Figura 21: 4ª seção do site.....	25
Figura 22: Continuação - 4ª seção do site.....	25
Figura 23: 5ª seção do site.....	26
Figura 24: Página de Upload de Fotos.....	27
Figura 25: Acessando a imagem carregada a partir do link .....	27
Figura 26: Procurar no Mapa - 1ª Parte - Coordenadas .....	28
Figura 27: Procurar no Mapa - 2ª Parte - Mapa Interativo.....	28
Figura 28: Página de Contato.....	29
Figura 29: Fórmula 1 .....	30
Figura 30: Fórmula 2 .....	30
Figura 31: Fórmula da Binarização .....	31
Figura 32: Imagem Original, Imagem com escala de cinza e Imagem Binarizada 'Binary_INV' .....	32
Figura 33: Exemplo de Imagem Binarizada no site .....	32
Figura 34: Conversão para M <sup>2</sup> quando Latitude e Nível de Zoom são 0 .....	32



## LISTA DE TABELAS

Tabela 1: Materiais necessários para o projeto .....	8
--	---

## **LISTA DE ABREVIATURAS E SIGLAS**

CV - Computer Vision

OPENCV - Open-Source Computer Vision Library.

JSON – Javascript Object Notation

INPE – Instituto Nacional de Pesquisas Espaciais

HTML – HyperText Markup Language

CSS – Cascading Style Sheet

JS – JavaScript

API - Application Programming Interface

SGDB – Sistema Gerenciador de Banco de Dados

SQL - Structure Query Language

WSGI - Web Server Gateway Interface

## SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>1</b>
1.1. Justificativa.....	2
1.2. Objetivo .....	3
1.2.1. Objetivos específicos.....	4
<b>2. LEVANTAMENTO TEÓRICO .....</b>	<b>4</b>
2.1. Visão Computacional .....	4
2.1.1. Sobre a Visão Computacional .....	4
2.2. Processamento de Imagens.....	6
2.2.1. Segmentação de Imagem.....	6
2.2.1.1. K-Means Clustering.....	7
<b>3. MATERIAIS E MÉTODOS .....</b>	<b>8</b>
3.1. HTML5 & CSS3 & JS .....	9
3.2. Sobre o uso do Node JS .....	10
3.3. Sobre o uso do PHP.....	12
3.4. MySQL .....	13
3.5. Sobre o uso do Python.....	14
3.6. Sobre o uso de API .....	15
3.6.1. Flask .....	15
3.6.1.1. WSGI.....	16
3.6.1.2. Werkzeug .....	16
3.6.1.3. Jinja2.....	16
3.6.1.4. Métodos HTTP .....	17
3.6.2. JSON .....	17
3.7. A Binarização .....	18
3.7.1. Binarização Adaptativa .....	19
3.8. Algoritmo do Holograma.....	20
3.8.1. Procedimento do Algoritmo .....	20
<b>4. FLUXOGRAMA .....</b>	<b>22</b>
<b>5. RESULTADOS OBTIDOS .....</b>	<b>23</b>
5.1. Site .....	23
5.1.1. Página Inicial .....	23
5.1.2. Página de Upload de Fotos .....	26
5.1.3. Página para Procurar no Mapa.....	28
5.1.4. Página de Contato .....	29

5.2. Conversão de pixel para metro quadrado (m <sup>2</sup> ).....	30
5.3. Execução da Binarização.....	30
<b>6. CONSIDERAÇÕES FINAIS.....</b>	<b>31</b>
6.1. Objetivos Alcançados.....	33
<b>7. REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>34</b>

## 1. INTRODUÇÃO

Nos dias atuais, o tema de desmatamento se tornou um assunto muito relevante, sempre colocando em pauta a situação da Amazônia (Nova Economia, 2019), as queimadas no Pantanal (ELLER, 2020), e as queimadas na Austrália (Nova Economia, 2019). Como foi dito em uma matéria:

Graças a evolução na área das ciências e zonas tecnológicas, o ser humano avançou em busca dos próprios interesses, sem a percepção de que estava colaborando para a formação de diversos riscos ambientais. (CAMILA, 2016).

As instituições existentes que monitoram áreas verdes, acabam promovendo pesquisas e estudos voltados para o desenvolvimento científico, justamente para conhecer as causas, os prejuízos e uma possível prevenção. Porém, todos estes processos são feitos manualmente, desde o cálculo de área, até o resultado, com todos os dados correlacionados.

Atualmente, a situação de verba na área ambiental é alarmante, o INPE (Instituto Nacional de Pesquisas Espaciais), em 2021, recebeu R\$ 44,7 milhões, frente a um total previsto de R\$ 76 milhões, junto ao Ministério da Ciência e Tecnologia (DESLIGAMENTO, 2021). Mediante a este corte, para a economia de energia e consequentemente verba, corre-se o risco de haver o desligamento do Tupã, o supercomputador do INPE utilizado para fazer previsão meteorológica e climática do Brasil, e as consequências seriam desastrosas, possuindo impactos sociais, econômicos e científicos. A ausência do Tupã causaria carência de informações sobre as previsões meteorológicas de curto e longo prazo, deixando-nos com um buraco nessas informações (CÔRTES, 2021). Dessa forma, o Software irá agir como uma alternativa para as instituições, onde possam economizar verba e tempo gasto nessas pesquisas.

O Sistema de Visão Computacional se tornou com o passar do tempo uma escolha como método de aplicação em várias áreas, onde desenvolve-se teoria e tecnologia para a construção de sistemas que obtém informação de imagens ou quaisquer dados multidimensionais (Data Science Academy, 2018), isto é um

setor da inteligência artificial pelo treinamento computacional em busca da compreensão e interpretação do mundo. (AUGUSTO, 2019)

Esta tecnologia em específico, vem evoluindo desde o final da década de 70, onde estudos foram sendo realizados em abundância, devido aos computadores que já podiam processar grandes anexos de dados como imagens (Data Science Academy, 2018). Com o avanço da tecnologia, a visão computacional tornou-se possível devido às diversas pesquisas científicas como já foi dito, portanto, a partir do âmbito atual podemos enxergá-la como ferramenta para minimizar os problemas citados anteriormente.

Este trabalho está dividido em capítulos, sendo eles:

O segundo capítulo, denominado “**Visão Computacional**”, apresentando a história e a criação desta tecnologia, bem como seu uso.

O terceiro capítulo, nomeado “**Processamento das Imagens**” mostra os métodos de como o cálculo, a binarização, a conversão dos pixels e a segmentação foi utilizada.

O quarto capítulo, designado “**Materiais e Métodos**” fala sobre o método e os materiais (tecnologias) utilizados para a elaboração e produção do projeto.

O quinto capítulo, chamado “**Resultados Obtidos**” traz os resultados obtidos e a diferença entre os resultados esperados.

O sexto capítulo intitulado “**Conclusões e Considerações Finais**” apresenta os aprendizados e pensamentos gerados com o desenvolvimento do projeto.

### 1.1. Justificativa

De 1964 até hoje, a área de processamento de imagens vem crescendo vigorosamente. Além de aplicações no programa espacial, técnicas de processamento de imagens digitais são atualmente utilizadas para resolver uma variedade de problemas (GONZALEZ et al., 2008), dentre eles, questões ambientais. O processamento digital de imagens oferece uma gama de

procedimentos que visam melhorar diversos aspectos das imagens de satélite, sendo um deles a resolução espacial. Muito utilizadas em estudos do uso e ocupação do solo, as imagens de satélite servem como matéria-prima para geração de diversos produtos, possibilitando ainda a realização de vários tipos de análises (ROCHA, 2018). Baseando-se principalmente em sua importância ambiental, o tema do projeto foi escolhido com o propósito de coadjuvar, através de um software, os órgãos ambientais competentes a identificar e calcular as áreas devastadas por incêndios. O intuito de determinar o prejuízo ambiental resulta na percepção global da região em questão proporção e o tempo médio para seu reflorestamento.

Como uma justificativa adicional que ratifica a importância e necessidade deste projeto, observou-se, mediante a pesquisas na internet, que é pioneiro ao abordar a ideia do cálculo de prejuízo de áreas devastadas inserida em um único software, com tecnologias atuais, como binarização e processamento de imagens, o que aumenta sua relevância no cenário atual.

Com a criação deste software os usuários poderão ir além de apenas consultar o mapa ou binarizar imagens, conseguirão também fazer o upload de fotos de devastações ou de florestas que possuem, para serem divulgadas, juntamente da visualização de uma galeria de imagens de florestas famosas ao redor do mundo, em conjunto de sua história e características, para trazer seu conhecimento e o porquê da importância de serem preservadas e cuidadas.

## **1.2. Objetivo**

Este projeto tem como objetivo geral automatizar o trabalho das organizações responsáveis, eliminando erros matemáticos, economizando tempo e esforço intelectual, tudo por meio de um software voltado aos cálculos sobre danos florestais.

### *1.2.1. Objetivos específicos*

Através das suas funcionalidades, o software permitirá um gerenciamento mais simples e eficaz das devastações. Com essas funções será possível:

1. Ter o controle mais preciso e identificação mais rápida das devastações florestais.
2. Informar aos responsáveis da área as informações relevantes e recorrentes.
3. Evitar erros de performance.
4. Tornar automático o trabalho de calcular os danos nos respectivos sistemas.
5. Calcular o tempo para reflorestamento da área devastada.
6. Permitir uma visão em 3D através de recurso holográfico para melhor análise.

## **2. LEVANTAMENTO TEÓRICO**

### **2.1. Visão Computacional**

Esse capítulo traz informação a respeito da visão computacional, como ela foi criada, o seu propósito e quais foram as suas evoluções com o passar dos anos.

#### *2.1.1. Sobre a Visão Computacional*

A Visão Computacional, em inglês Computer Vision (CV), é uma área da ciência que se dedica ao desenvolvimento de métodos e teorias destinados a retirar informações úteis de imagens. Essas informações são transmitidas para as máquinas, mas não só para reconhecê-las. Com esta tecnologia, o aprendizado das máquinas inclui processamento, análise e identificação com processos tão complexos como os da visão humana.



Para que essa tecnologia funcione, o estudo da capacidade da visão humana é fundamental, pois é a partir desse entendimento que se faz possível o desenvolvimento de ferramentas que desempenham funções de maneira semelhante.

Assim, a CV busca replicar a visão da mesma forma que é concebida nos seres humanos. Através dela, é possível construir, reconstruir e visualizar vídeos ou imagens utilizando a análise de pixels e identificação de padrões.

As primeiras experiências em visão computacional aconteceram nos anos 1950, com o uso de algumas das primeiras redes neurais para detectar os limites de um objeto e para classificar objetos simples em categorias como círculos e quadrados. No final dos anos 1960, a visão computacional começou em universidades que eram pioneiras em inteligência artificial e foi concebida para imitar o sistema visual humano, como um trampolim para dotar robôs com comportamento inteligente. Em 1966, acreditava-se que isso poderia ser conseguido através de um projeto de verão, anexando uma câmera a um computador e fazendo com que ele “descrevesse o que via”.

O que distinguia a visão computacional do campo predominante do processamento digital de imagens na época era o desejo de extrair a estrutura tridimensional das imagens com o objetivo de alcançar uma compreensão plena da cena. Estudos na década de 1970 formaram as primeiras bases para muitos dos algoritmos de visão computacional que existem hoje, incluindo extração de bordas de imagens, rotulação de linhas, modelagem não-poliédrica e poliédrica, representação de objetos como interconexões de estruturas menores, fluxo óptico e estimativa de movimento.

A próxima década viu estudos baseados em análises matemáticas mais rigorosas e aspectos quantitativos da visão computacional. Isso inclui o conceito de escala de espaço, a inferência da forma a partir de várias sugestões, como sombreamento, textura e foco, e modelos de contorno conhecidos como cobras. Os pesquisadores também perceberam que muitos desses conceitos matemáticos poderiam ser tratados dentro da mesma estrutura de otimização dos campos aleatórios de regularização.

Na década de 1990, alguns dos tópicos de pesquisa anteriores tornaram-se mais ativos do que os outros. Pesquisas em reconstruções 3-D projetivas levaram a um melhor entendimento da calibração da câmera. Com o advento dos métodos de otimização para calibração de câmeras, percebeu-se que muitas das ideias já foram exploradas na teoria de ajuste de pacotes do campo da fotogrametria. Isso levou a métodos para reconstruções esparsas em 3-D de cenas de várias imagens.

Progresso foi feito no problema de correspondência estéreo denso e outras técnicas de estéreo multi-visão. Ao mesmo tempo, variações do corte de grafos foram usadas para resolver a segmentação de imagens. Esta década também marcou a primeira vez que técnicas de aprendizado estatístico foram usadas na prática para reconhecer rostos em imagens (Eigenface). Com o amadurecimento da internet nos anos 1990, grandes volumes de imagens foram disponibilizados online para análises, e o desenvolvimento de programas de reconhecimento de objetos começou a crescer.

Esses conjuntos de dados ajudaram a possibilitar que máquinas identifiquem pessoas específicas em fotos e vídeos. Já no final dos anos 90, uma mudança significativa ocorreu com o aumento da interação entre os campos de computação gráfica e visão computacional. Isso inclui renderização baseada em imagem, morphing de imagem, interpolação de visualização, costura de imagem panorâmica e renderização antecipada de campo de luz.

## **2.2. Processamento de Imagens**

Neste capítulo, iremos mostrar os métodos de como a segmentação, o cálculo, a conversão dos pixels e a binarização foi utilizada.

### **2.2.1. Segmentação de Imagem**

A segmentação de imagens é um passo importante no processamento de imagens, e aparece em todos os lugares se quisermos analisar o que está dentro

da imagem. Por exemplo, se procurarmos descobrir se há uma cadeira ou uma pessoa dentro de uma imagem interna, podemos precisar de segmentação de imagem para separar objetos e analisar cada objeto individualmente para verificar o que é. A segmentação de imagens geralmente serve como o pré-processamento antes do reconhecimento de padrões, extração de recursos e compressão de uma imagem.

A segmentação de imagem envolve a conversão de uma imagem em uma coleção de regiões de pixels que são representados por uma máscara ou uma imagem rotulada. Dividindo uma imagem em segmentos, você pode processar apenas os segmentos importantes da imagem em vez de processar toda a imagem.

Uma técnica comum é procurar descontinuidades abruptas nos valores dos pixels, que normalmente indicam bordas que definem uma região.

Outra abordagem comum é detectar semelhanças nas regiões de uma imagem. Algumas técnicas que seguem essa abordagem são denominadas Clustering (Agrupamento) e o Thresholding (Limiar).

#### *2.2.1.1. K-Means Clustering*

O algoritmo é usado quando você tem dados não rotulados (ou seja, dados sem categorias ou grupos definidos). O objetivo é encontrar certos grupos com base em algum tipo de semelhança nos dados com o número de grupos representados por K.

Figura 1: Exemplo de Clustering



<<https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html>>

### 3. MATERIAIS E MÉTODOS

Para o desenvolvimento deste projeto foi utilizado o método de engenharia.

Tabela 1: Materiais necessários para o projeto

Qtd	Recurso	Tipo	Valor
01	Mesa com cadeira para escritório	Móvel	R\$ 417,99 / Site para a consulta de preço: <a href="https://www.madeiramadeira.com.br">https://www.madeiramadeira.com.br</a> , 28 ago. 2021
01	Computador Intel Core I5 7200U, 8GB RAM, 1TB HD.	Hardware	R\$ 2.199 / Site para a consulta de preço: <a href="https://quenotebookcomprar.com.br">https://quenotebookcomprar.com.br</a> , 28 ago. 2021
01	Sistema Operacional Windows 10 x64 Home Single Language Edition.	Licença Software	Gratuito
01	Pacote Office 365	Licença Software	Gratuito

01	Microsoft Visual Studio Code 2020 x64.	Software	Gratuito
01	WampServer	Software	Gratuito
01	MySQL Workbench (Community Edition)	Software	Gratuito
01	MySQL - Front	Software	Gratuito
01	Postman	Software	Gratuito
01	PyCharm (Community Edition)	Software	Gratuito
01	Trello	Software	Gratuito
<b>VALOR TOTAL: R\$2.616,99</b>			

Vale a pena ressaltar que dos softwares utilizados na produção do projeto, apenas as licenças: Sistema Operacional Windows 10 x64 e Pacote Office 365, são pagas, porém isso não influenciou no custo uma vez que a escola ficou responsável pelos mesmos, participando com o computador com todos os recursos devidamente instalados.

### 3.1. HTML5 & CSS3 & JS

O HTML (HiperText Markup Language) é uma linguagem de marcação de hipertexto para páginas web. Ele define a estrutura da página, através de “tags” de marcação. Sua estilização, bem como de seus componentes, é realizada através da linguagem chamada CSS (Cascading Style Sheet), onde através dela, pode-se modificar esteticamente as tags presentes em um arquivo HTML, mas não é a única forma de estilização existente.

Figura 2: Página inicial do site



&lt;Arquivo Pessoal (2021)&gt;

O JS (JavaScript), por sua vez, é uma linguagem de programação, tanto *Front-End* (lado do usuário/cliente), com animações, mudanças dinâmicas e interações, quanto *Back-End* (lado servidor), com conexões a bancos de dados, validações de formulário, e, até mesmo, não estando associado a um navegador.

### 3.2. Sobre o uso do Node JS

Node.js é um ambiente de execução JavaScript server-side, ou seja, é um ambiente para execução de JavaScript do lado do servidor. Ao contrário do que se pensa, JS não é apenas utilizado para implementar animações em uma página web.

É uma plataforma de aplicação onde seu código JavaScript será compilado, otimizado e interpretado pelo V8 (máquina virtual que o navegador Chrome do Google usa para executar JavaScript no browser).

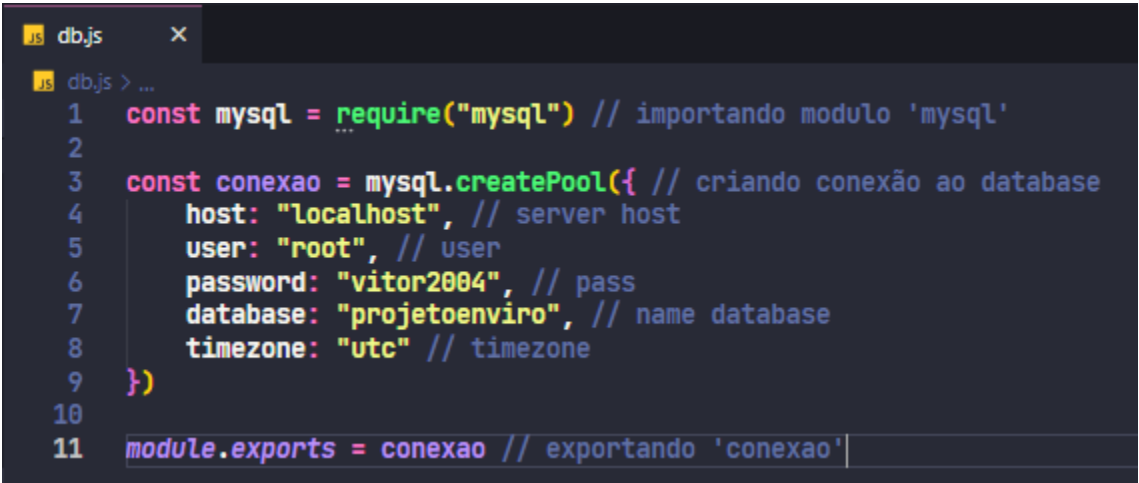
O uso do Node JS traz inúmeros benefícios para uma aplicação, dentre eles:

- **Produtividade:** Pelo fato do Node JS utilizar JavaScript em sua codificação, ele conta com uma gama de possibilidades elevada, inclusive com a reutilização de códigos, em que os usuários podem aproveitar melhor os recursos disponíveis.

- **Flexibilidade:** O Node.js é uma plataforma muito versátil e que pode ser usada em inúmeros cenários. Seu gerenciador de pacotes Node Package Manager (NPM) é classificado como o maior repositório de softwares disponível, sendo que o pacote mais famoso é um framework voltado ao desenvolvimento web, chamado *Express.js*.
- **Leveza:** Criar um ambiente Node.js e subir uma aplicação é uma tarefa que não exige muitos recursos computacionais em comparação com outras tecnologias mais tradicionais, o que o torna bastante popular pelos usuários.

Seu uso será de extrema importância para o projeto, onde no *Back-End* será feita a autenticação de usuários, com cadastro e login, controle de acesso do usuário, simulação de um servidor e procedimentos da API.

Figura 3: Criando conexão ao Banco de Dados com JS e MySQL



```
db.js
1  const mysql = require("mysql") // importando modulo 'mysql'
2
3  const conexao = mysql.createPool({ // criando conexão ao database
4    host: "localhost", // server host
5    user: "root", // user
6    password: "vitor2004", // pass
7    database: "projetoenviro", // name database
8    timezone: "utc" // timezone
9  })
10
11 module.exports = conexao // exportando 'conexao'
```

<Arquivo Pessoal (2021)>

### 3.3. Sobre o uso do PHP

A linguagem de programação PHP é uma linguagem de script open source (código aberto) de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML.

O PHP foi utilizado em nosso projeto para os códigos e scripts da função de upload de fotos do site, permitindo o armazenamento das fotos que o usuário carregar ao site em nosso banco de dados.

Figura 4: Código em PHP para armazenamento de fotos

```

1 <?php
2
3 include("../bd/bd.php");
4 $msg = false;
5
6 if(isset($_FILES['arquivo'])){
7
8     $extensao = strtolower(substr($_FILES['arquivo']['name'], -4)); // Seleciona a extensão do arquivo usando substring
9     $novo_nome = md5(time()) . $extensao; // Criptografa o nome do arquivo e concatena com sua extensão
10    $diretorio = "img/"; // Indica o diretório para onde as imagens deverão ir
11    $raiz = "uploads/";
12    move_uploaded_file($_FILES['arquivo']['tmp_name'], $diretorio.$novo_nome); // Efetua o upload
13
14
15    $sql = "INSERT INTO imagem (idImagem,imagem,data) VALUES(null,'$raiz$diretorio$novo_nome',NOW())"; // Instrução SQL para armazenamento
16    if (mysqli_query($connection, $sql)){
17        $msg = "arquivo enviado com sucesso";
18    }
19    else {
20        $msg = "falha ao enviar arquivo";
21    }
22
23    header("location: ../uploadfotos.php ");
24
25 }
26

```

<Arquivo Pessoal (2021)>

Na figura 4 é apresentado o código para o armazenamento de fotos, onde é criada uma variável chamada 'extensao' (\$extensao) para selecionar a extensão da imagem, utilizando o método de substring. Depois é inserido um novo nome na imagem utilizando criptografia e por último, armazena-se na variável diretório a pasta que se deseja guardar as imagens carregadas.

A função "move\_uploaded\_file" é a que faz o arquivo ser direcionado ao diretório especificado, o que a torna a função mais importante deste script.

E por fim, como última etapa do código, a imagem é armazenada no banco de dados, com seu nome e data de inserção, com o comando "INSERT INTO".

Figura 5: Registro da imagem no banco de dados

idImagem	imagem	data
1	61322c386dc85eecead557e3ae5de386.jpg	2021-09-12 21:39:30

<Arquivo Pessoal (2021)>



### 3.4. MySQL

O MySQL é um Sistema Gerenciador de Banco de Dados (SGBD) relacional, normalmente utilizado em aplicações gratuitas para gerir um Banco de Dados. Utiliza a linguagem SQL (Structure Query Language – Linguagem de Consulta Estruturada) para inserir, consultar, alterar e deletar registros.

Este SGDB possui várias vantagens a partir de seu uso ou pretensões do mesmo, tais como:

- **Código aberto:** A flexibilidade oferecida por seu caráter de código aberto é uma grande vantagem do MySQL, além de ser gratuito e fácil de usar.
- **Facilidade de uso:** O MySQL é simples de configurar e requer mínimos ajustes de configurações para atingir níveis de desempenho excelentes. Ferramentas de interface gráfica terceiras, como MySQL Workbench, que foi utilizado neste projeto e simbolizado na *figura 4*, e dbForge Studio, tornam o MySQL ainda mais simples para começar a utilizá-lo.
- **Compatibilidade:** O MySQL oferece compatibilidade com a maioria das principais plataformas de computador, como Linux, macOS, Microsoft Windows e Ubuntu. Além disso, proporciona alto desempenho para armazenamento de grandes volumes de dados.

Para utilizá-lo, é necessário instalar um servidor, como WampServer, XAMPP, entre outros, e uma aplicação cliente. O servidor é o responsável por armazenar os dados, responder às requisições, controlar a consistência dos dados, bem como a execução de transações concomitantes entre outras. O cliente se comunica com o servidor através de instruções SQL.

Figura 6: Estrutura do Banco de Dados e relacionamentos das tabelas



<Arquivo Pessoal (2021)>

O diagrama acima, mostra a estrutura e a relação entre as tabelas do banco de dados utilizado para o projeto, onde a tabela “Usuarios” é composta pelo “Nome”, “Email” e “Senha”, bem como a identificação do “Id” do usuário, e a “Links\_Imagens” composta pelo “Id” do usuário referente ao Id da tabela Usuarios e o link da imagem, que redireciona para a sua visualização no repositório de imagens na nuvem. Foi desenvolvido para armazenar as informações do usuário na hora do cadastro no site e armazenar o link das imagens.

Podemos observar que a relação das tabelas é de 1:n (1 para vários), o que significa que um usuário pode ter um ou vários links de imagens, mas estes links só podem pertencer a um usuário.

### **3.5. Sobre o uso do Python**

Python é uma linguagem de programação criada por Guido van Rossum em 1991. Os objetivos do projeto da linguagem eram: produtividade e legibilidade, em outras palavras, Python é uma linguagem que foi criada para produzir código bom e fácil de manter de maneira rápida.

Python tem uma biblioteca padrão imensa, que contém classes, métodos e funções para realizar essencialmente qualquer tarefa, desde acesso a bancos de dados a interfaces gráficas com o usuário. E, logicamente, já que esse é o objetivo deste projeto, existem muitas ferramentas para lidar com dados científicos. Essa característica da linguagem é comumente chamada “baterias inclusas”, significando que tudo que você precisa para rodar um programa está — na maior parte das vezes — presente na instalação básica. Por sua eficiência, o Python foi utilizado em todos os algoritmos de processamento do projeto.

### 3.6. Sobre o uso de API

API é um conjunto de definições e protocolos usados no desenvolvimento e na integração de software de aplicações. É um acrônimo em inglês que significa interface de programação de aplicações. (INTERFACE, 2018)

Uma API permite que sua solução ou serviço se comunique com outros produtos e serviços sem precisar saber como eles foram implementados. Isso simplifica o desenvolvimento de aplicações, gerando economia de tempo e dinheiro. Ao desenvolver novas ferramentas e soluções (ou ao gerenciar aquelas já existentes), as APIs oferecem a flexibilidade necessária para simplificar o design, a administração e o uso, além de fornecer oportunidades de inovação. (INTERFACE, 2018).

Seu uso será para realizar a conexão entre o site no *Front-End*, e o processamento de conversão de píxeis, por exemplo, no *Back-End*. Para fazê-la, será utilizado um framework chamado **Flask**.

#### 3.6.1. Flask

Flask é um micro framework que utiliza a linguagem Python para criar aplicativos Web. É ideal para quem procura:

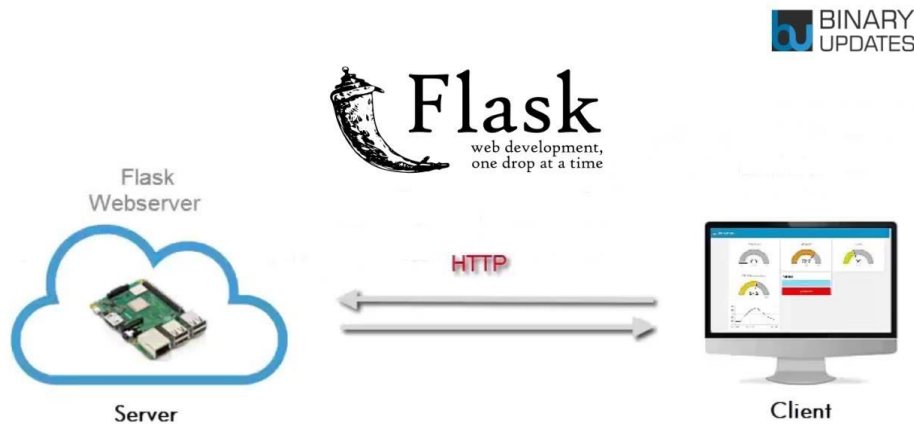
- Simplicidade;
- Rapidez;
- Soluções para projetos pequenos;
- Aplicações robustas.

Lançado em 2010 e desenvolvido por Armin Ronacher, a estrutura do Flask permite desenvolver aplicativos da web facilmente.

Ronacher ganhou fama por liderar uma comunidade de entusiastas do Python chamada Poooco. O Flask é baseado no kit de ferramentas Werkzeug WSGI e na biblioteca Jinja2. Ambos são projetos herdados da Poooco.

No projeto, o Flask é utilizado como um servidor web para a API, como é representado pela imagem abaixo:

Figura 7: Representação do micro framework Flask



<<https://www.youtube.com/watch?v=8p5B5sDw3k0>>

#### 3.6.1.1. WSGI

A Web Server Gateway Interface (WSGI) tem sido usada como um padrão para o desenvolvimento de aplicativos da Web em Python.

Em suma, WSGI é a especificação de uma interface comum entre servidores e aplicativos da web.

#### 3.6.1.2. Werkzeug

O Werkzeug é uma das bases do Flask, um kit de ferramentas WSGI que implementa funções como *requests* (requisições), por exemplo.

#### 3.6.1.3. Jinja2

O Jinja2 é um sistema de web template que combina um modelo com uma fonte de dados específica para renderizar uma página da web.

#### 3.6.1.4. Métodos HTTP

O protocolo HTTP define um conjunto de métodos de requisição responsáveis por indicar a ação a ser executada para um dado recurso. São recursos a serem estudados para o desenvolvimento de uma API, pois todos as requisições envolvem estes métodos. O método “POST” foi utilizado para este caso de uso, pois sua função é criar um recurso.

Figura 8: Exemplo de ações - Métodos HTTP

Endpoint	Método	Ação
/users	GET	Retorna a lista de usuários
/users	POST	Insere um novo usuário
/users/{id}	GET	Retorna o usuário com id = {id}
/users/{id}	PUT	Substitui os dados do usuário com id = {id}
/users/{id}	PATCH	Altera itens dos dados do usuário com id = {id}
/users/{id}	DELETE	Remove o usuário com id = {id}

<<https://blog.mbeck.com.br/api-rest-e-os-verbos-http-46e189085e21>>

#### 3.6.2. JSON

O JSON, ou JavaScript Object Notation, é um tipo de “linguagem universal” usada para a comunicação entre aplicações. É importante compreender, no entanto, que não se trata de uma linguagem de desenvolvimento, mas de um padrão de formatação leve que permite a troca de dados e informações entre diferentes linguagens e sistemas. (NOLETO, 2020)

Os arquivos JSON utilizam a extensão .json e são extremamente fáceis de ler, pois contêm apenas texto puro estruturado de forma simples e intuitiva. Além disso, a sintaxe do JSON é derivada da linguagem de programação JavaScript, o que facilita o uso em conjunto das duas ferramentas. No entanto, grande parte das linguagens modernas pode facilmente se comunicar com o JSON. (NOLETO, 2020)

Figura 9: Exemplo de código em JSON (.json)

```
1  {  
2      "nome": "Bruno",  
3      "sobrenome": null,  
4      "idade": 20,  
5      "ativo": true,  
6      "endereco": {  
7          "rua": "Rua Vinte e Um de Abril",  
8          "numero": 18  
9      },  
10     "turmas": [  
11         "Programação Orientada a Objetos",  
12         "Lógica de Programação"  
13     ]  
14 }
```

<<https://digitalinnovation.one/artigos/json-um-manual-completo-da-linguagem>>

A estrutura acima define algumas informações retiradas de um usuário. Neste exemplo, inclui: nome; sobrenome; idade; se o usuário é ativo (*true*) ou não (*false*); endereço e turma. Para o projeto, o arquivo JSON da requisição deve indicar a latitude e o nível de zoom do mapa, ambas informações utilizadas para as fórmulas de conversão de pixels para metros quadrados.

O uso do JSON será de suma importância para fazer a transferência das informações entre o Cliente (Usuário - Site) e o Servidor (Processamento), tanto para fazer os cálculos, quanto para devolver o seu resultado.

### 3.7. A Binarização

A binarização é um dos mais conhecidos algoritmos de tratamento de imagem, sua função é fazer a conversão de uma imagem com níveis de cinza para uma imagem com representação binária (dois tons), preto ou branco, e pode ser importante para diversas atividades, tais como:

- Identificar objetos e separá-los do fundo da imagem;
- Quando analisar a forma da imagem é mais importante que a intensidade dos píxeis;
- Identificar padrões.

A fórmula geral da binarização pode ser definida pela imagem abaixo:

*Figura 10: Fórmula Geral da Binarização*

The thresholded image  $g(x,y)$  is defined as  $g(x,y)$

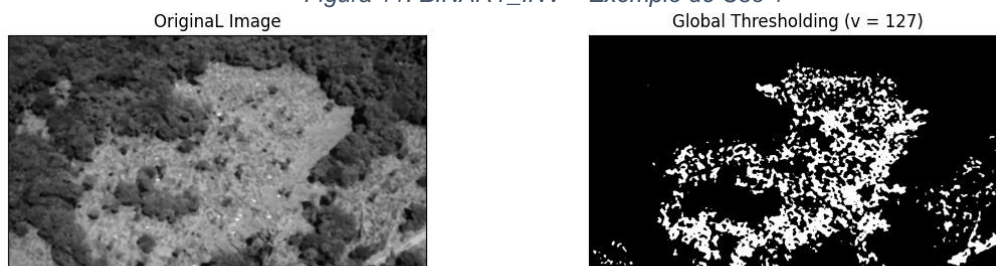
$$g(x,y) = \begin{cases} 1 & \text{if } (x,y) > T \\ 0 & \text{if } (x,y) \leq T \end{cases}.$$

<Arquivo Pessoal (2021)>

### 3.7.1. Binarização Adaptativa

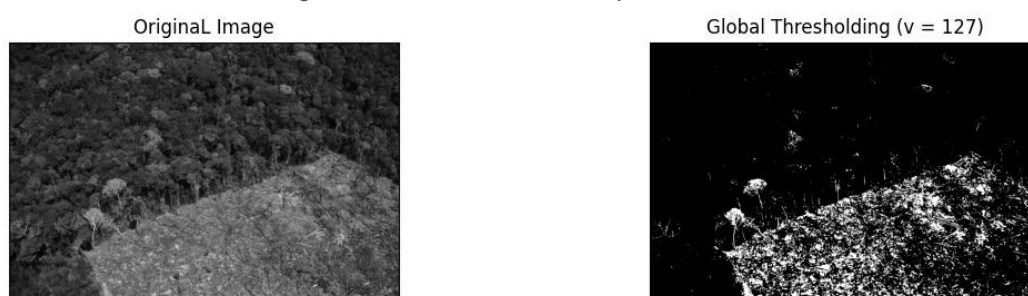
O algoritmo de binarização tem 5 métodos diferentes, cada um destinado a objetivos diferentes, o que foi usado para o projeto se chama binarização adaptativa, também conhecida como “BINARY\_INV”, essa escolha foi feita através de diversos testes, com diferentes imagens e padrões. De acordo com a nossa necessidade, esse método obteve o maior número de acertos.

*Figura 11: BINARY\_INV – Exemplo de Uso 1*



<Arquivo Pessoal (2021)>

*Figura 12: BINARY\_INV - Exemplo de Uso 2*



<Arquivo Pessoal (2021)>

Figura 13: BINARY\_INV - Exemplo de Uso 3



&lt;Arquivo Pessoal (2021)&gt;

### 3.8. Algoritmo do Holograma

Conforme o andamento do projeto, pensando em uma forma de obter uma melhor visualização sobre a área calculada, foi adicionado um algoritmo para rotacionar uma imagem, de modo que seja utilizado posteriormente em um holograma.

#### 3.8.1 Procedimento do Algoritmo

Para o algoritmo funcionar, foram utilizadas algumas funções da biblioteca “PIL”, para que fosse possível a rotação das imagens.

O primeiro passo dessa estrutura é localizar a imagem que está salva em um diretório local, onde foi utilizado uma função chamada “image.open”, que recebe o valor da imagem e salva em uma variável. Isso será necessário para o andamento do código.

Figura 14: Adquirindo a imagem

```
#adquirindo a imagem no diretório
Imagem_original = Image.open("/content/fotos.bio/BINARY_INV.png")
```

&lt;Arquivo Pessoal (2021)&gt;

O segundo passo é o processo de rotacionar a imagem original em três ângulos diferentes, fazendo com que possua quatro imagens ao todo. Foram utilizadas duas funções nesse passo (rotate e a transpose), onde a rotate foi



usada para inverter a imagem em 180° graus, e a transpose para as imagens de 90° e 270° graus.

Figura 15: Rotacionando as imagens

```
#rotação das imagens
imagem_invertida = Imagem_original.rotate(180)

imagem_esquerda = Imagem_original.transpose(Image.ROTATE_90)

imagem_direita = Imagem_original.transpose(Image.ROTATE_270)
```

<Arquivo Pessoal (2021)>

O terceiro e último passo é salvar essas imagens em um outro diretório, criado para armazenar as imagens rotacionadas pelo algoritmo. O Objetivo de salvar as imagens em um diretório diferente é justamente para não ter nenhum conflito com o diretório que recebeu a imagem original.

Figura 16: Armazenamento das imagens em um Diretório

```
#salvando-as em um Diretório específico
Imagem_original.save("/content/fotos_rotacionadas/imagem_rotacionada_or.png")
imagem_invertida.save("/content/fotos_rotacionadas/imagem_rotacionada_inv.png")
imagem_esquerda.save("/content/fotos_rotacionadas/imagem_rotacionada_esq.png")
imagem_direita.save("/content/fotos_rotacionadas/imagem_rotacionada_dir.png")
```

<Arquivo Pessoal (2021)>

Após estes processos, as quatro imagens estarão prontas para serem implementadas em um holograma.

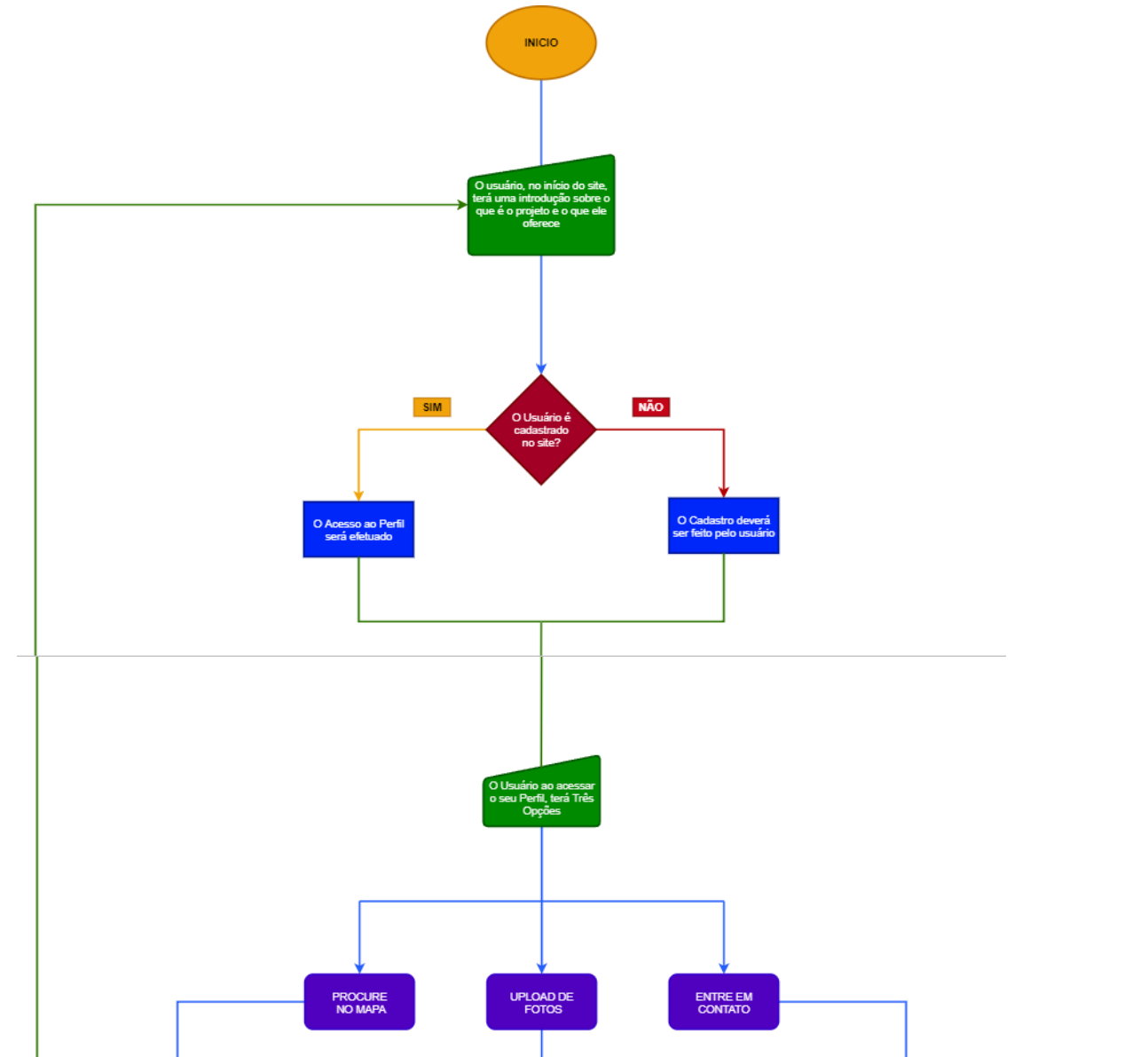
Figura 17: Resultado do algoritmo

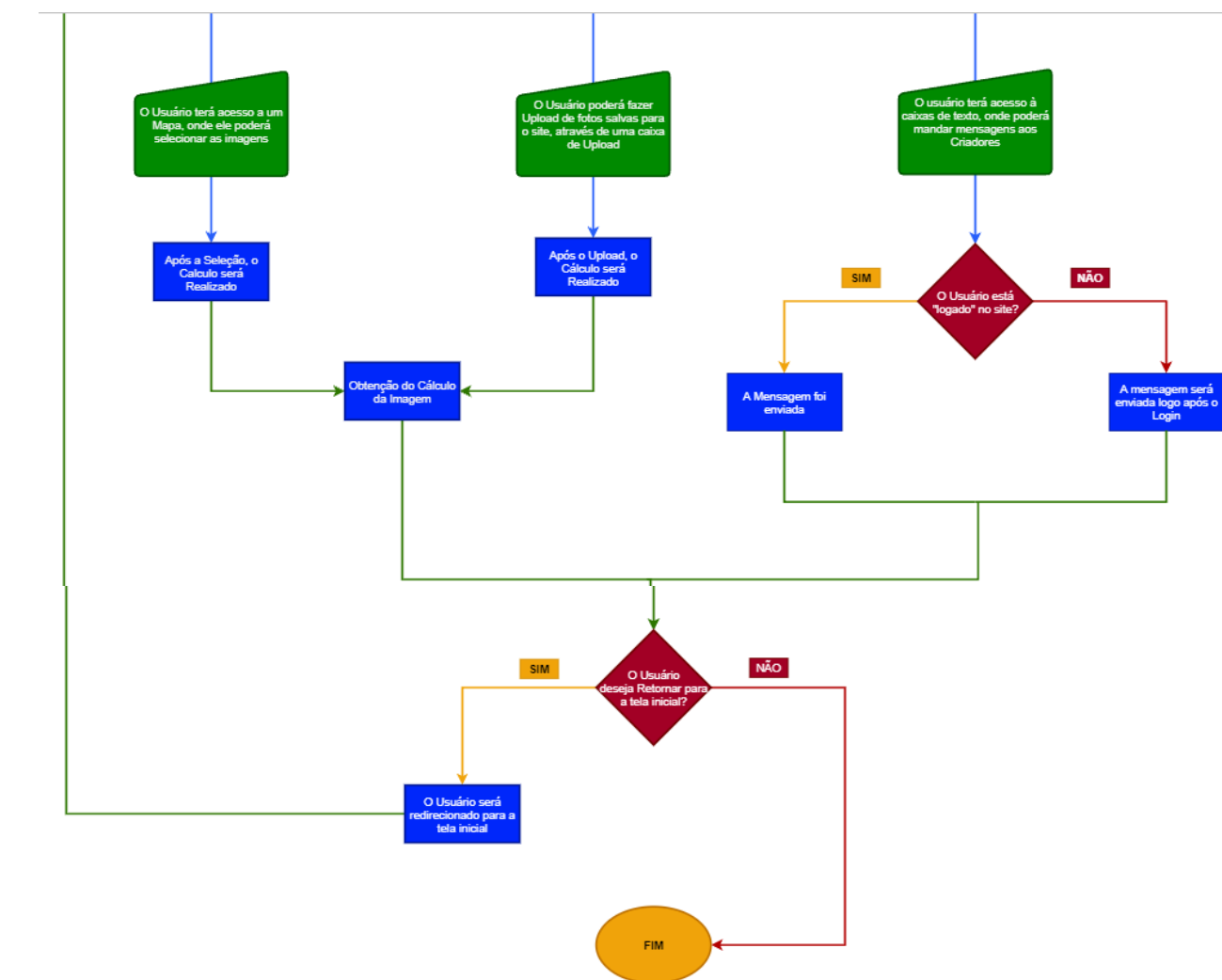


<Arquivo Pessoal (2021)>

#### 4. FLUXOGRAMA

Figura 18: Fluxograma





## 5. RESULTADOS OBTIDOS

### 5.1. Site

#### 5.1.1. Página Inicial

A primeira etapa do projeto foi o desenvolvimento do site Web, nas linguagens HTML, CSS e JS. Porém, vale ressaltar que para fins práticos, foi utilizado na posição do HTML (.html) o Handlebars (.hbs), um Template Engine (Processador de Modelo) que oferece inúmeras funcionalidades ao HTML, que por si só não possui, como estruturas de repetição, condicionais e, principalmente, exibir dados que vêm do *Back-End* da aplicação ao *Front-End*.

A parte inicial do site, como já ilustrada na *Figura 2*, é bastante simples e representa apenas a introdução do site, possuindo um cabeçalho com um menu *drop-down* (que ao passar o *mouse* as opções “caem para baixo”) das sessões do site, botão para efetuar o *logout* para caso esteja logado em uma conta e desejar sair da mesma, botão para fazer login ou criar uma conta e por último, mas não menos importante, outro botão para ativar/desativar o *darkmode* (modo escuro) do site. Como plano de fundo, uma imagem panorâmica da Floresta Amazônica e ao centro da tela, o nome do projeto

Descendo, explica-se um pouco sobre a criação do projeto, bem como seus objetivos, além de um *link* para ler a respeito das informações técnicas, como métodos e linguagens utilizados, e o nome dos autores.

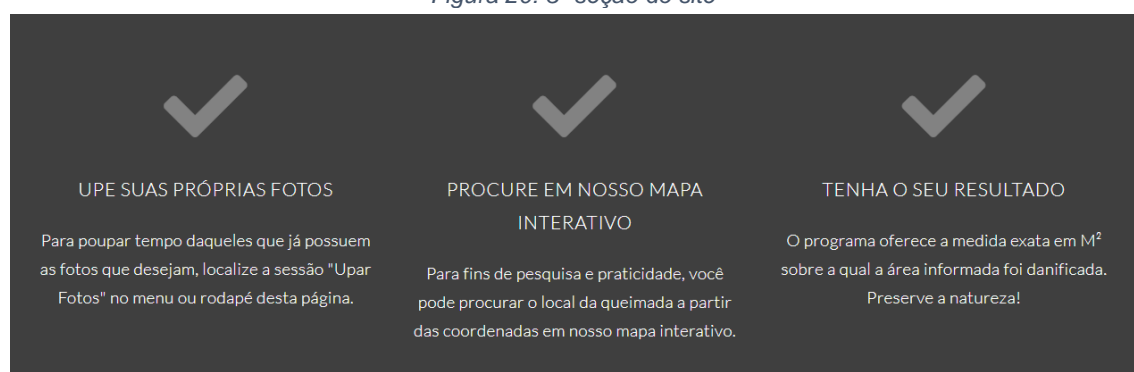
*Figura 19: 2ª seção do site*



<Arquivo Pessoal (2021)>

Na *Figura 11*, é apresentada uma seção com as funcionalidades, sendo elas o *upload* de fotos, a procura no mapa interativo e o resultado.

*Figura 20: 3ª seção do site*



<Arquivo Pessoal (2021)>

Seguidamente, as figuras 12 e 13, mostra ao usuário 4 imagens de florestas famosas no mundo, trazendo um incentivo à preservação das mesmas. Sendo elas: Floresta Amazônica, Floresta de Taiga, Reserva Florestal Nublada, e Sundarbans. Faz-se presente também uma pequena descrição de cada uma.

Figura 21: 4ª seção do site



<Arquivo Pessoal (2021)>

Figura 22: Continuação - 4ª seção do site



<Arquivo Pessoal (2021)>

Figura 23: 5ª seção do site



<Arquivo Pessoal (2021)>

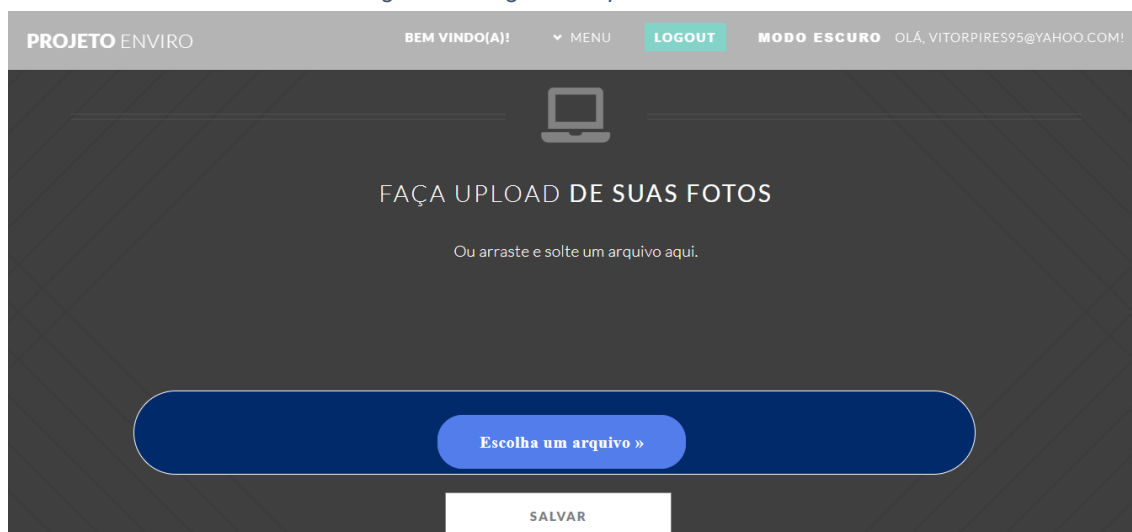
Por último, na *figura 14* é ilustrada a 5ª seção, onde é apresentado um título em inglês, com tradução significando “Pronto para fazer alguma coisa?”, onde sugere ao usuário acessar as duas ferramentas principais do site.

#### 5.1.2. Página de Upload de Fotos

Esta página é exclusiva, fazendo com que apenas os usuários que estão logados com suas credenciais consigam acessá-la, este procedimento foi feito no *Back-End* utilizando o NodeJS, como explicado no tópico 3.2.

É nela que os usuários poderão carregar suas imagens ao site, através do *Uploader*, um elemento do HTML que permite carregar arquivos de seu computador.

Figura 24: Página de Upload de Fotos



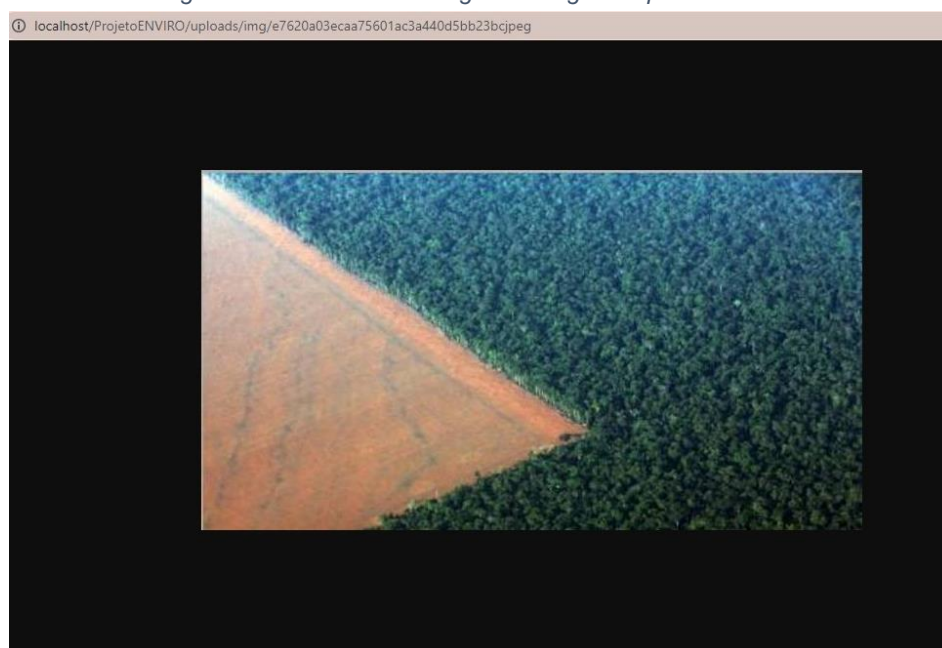
&lt;Arquivo Pessoal (2021)&gt;

Como outra funcionalidade, é apresentado ao usuário o e-mail a qual ele está logado no momento, como indicado no canto superior direito da *Figura 19*. E vale reforçar a obrigatoriedade do login para acessar esta área do site.

Ao clicar no botão “Escolha um Arquivo”, será aberto o explorador de arquivos do Windows, onde o usuário selecionará sua imagem, podendo-a ser com extensões *.jpg*, *.jpeg* e *.png*., como configurado.

Após tê-la carregado e clicado no botão “SALVAR”, o link para o acesso à figura será salvo no banco de dados, através do script explicado no tópico 3.3.

Figura 25: Acessando a imagem carregada a partir do link



&lt;Arquivo Pessoal (2021)&gt;



### 5.1.3. Página para Procurar no Mapa

Como a página de Upload de Fotos, esta também é exclusiva para os usuários que estejam logados no site.

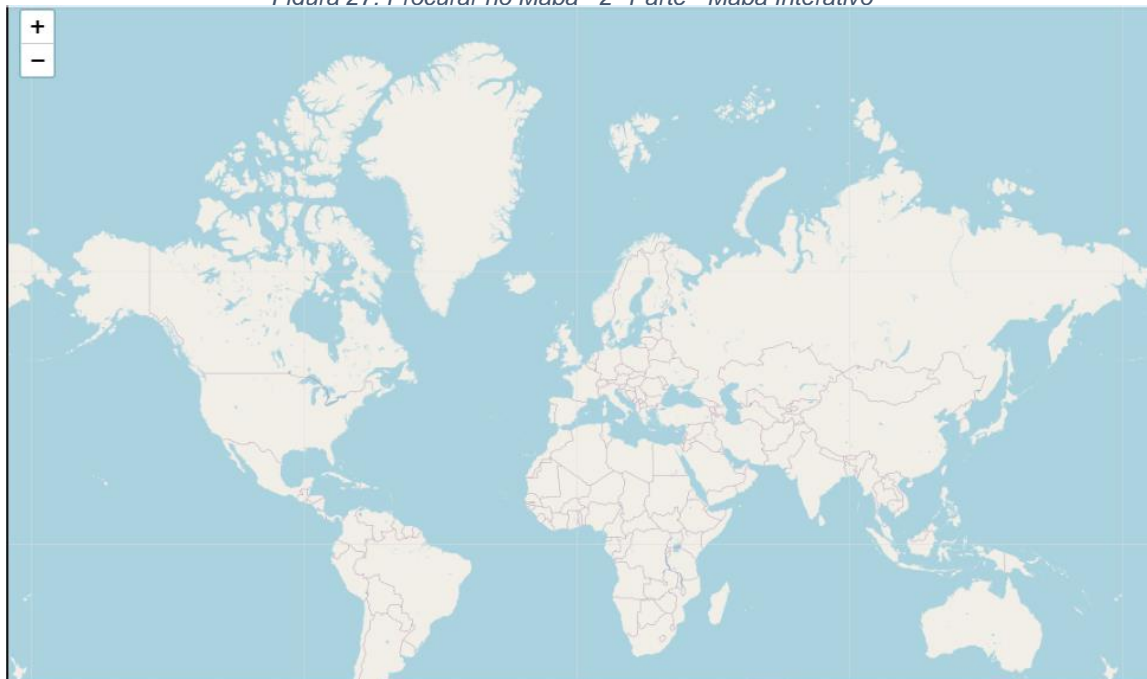
É nesta parte do site que o usuário poderá interagir com o Mapa Interativo, ou colocar as coordenadas, podendo ir para qualquer lugar do mundo.

Figura 26: Procurar no Mapa - 1ª Parte - Coordenadas

<Arquivo Pessoal (2021)>

Após serem inseridas as coordenadas, estes valores serão enviados via JSON (explicado no item 3.5.2.) para a API, que encaminhará esta informação ao *endpoint* correspondente, para então, calcular a área e fazer a conversão.

Figura 27: Procurar no Mapa - 2ª Parte - Mapa Interativo



<Arquivo Pessoal (2021)>



Juntamente com a página Upload de Fotos, é a página principal do site, o que faz o Projeto ENVIRO ser o Projeto ENVIRO, tornando-o dinâmico com o usuário e fazendo o que se propõe em objetivos.

#### 5.1.4. Página de Contato

Nesta página, o usuário não precisa estar logado no site para acessar.

Nela, ele poderá entrar em contato por e-mail diretamente com os criadores do site, inserindo seu nome, e-mail, assunto e a própria mensagem.

Figura 28: Página de Contato



## 5.2. Conversão de pixel para metro quadrado (m²)

Como parte crucial do cálculo, foram utilizadas duas fórmulas disponibilizadas pela documentação oficial da “OpenStreetMap”, empresa da qual utilizamos seus recursos de mapeamento para o site do projeto, e por conta disso, seria necessário seguir suas fórmulas para evitar erros de escala.

Contextualizando, as duas equações representadas pelas imagens abaixo são usadas em conjunto, dado que o resultado da primeira fórmula se utiliza na segunda, como podemos perceber pela variável “*Stile*”.

Relembrando o que foi dito anteriormente quando falamos sobre JSON, essas informações variáveis (*latitude* e *zoomlevel*) são essenciais para tal conversão, e por isso, a cada nova requisição de um usuário, será enviado um arquivo do formato JSON para o corpo do processamento.

Figura 29: Fórmula 1

The horizontal distance represented by each square tile, measured along the parallel at a given latitude, is given by:

$$S_{tile} = C \cdot \cos(latitude) / 2^{zoomlevel}$$

where C means the equatorial circumference of the Earth (40 075 016.686 m  $\approx 2\pi \cdot 6\,378\,137.000$  m for the reference geoid used by OpenStreetMap).

<openstreetmap.org>

Figura 30: Fórmula 2

As tiles are 256-pixels wide, the horizontal distance represented by one pixel is:

$$S_{pixel} = S_{tile} / 256 = C \cdot \cos(latitude) / 2^{(zoomlevel + 8)}$$

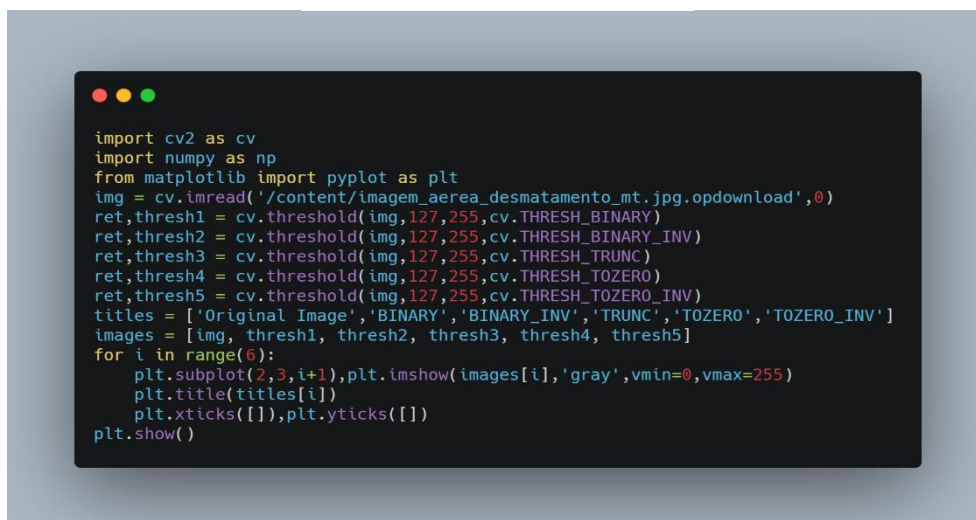
For example on the equator and at zoom level 0, we get 40 075 016.686 / 256  $\approx 156\,543.03$  (in meters per pixel).

<openstreetmap.org>

## 5.3. Execução da Binarização

Um dos resultados obtidos pelo projeto foi a realização da binarização adaptativa, e por ser o único método de tratamento de imagem, foi uma conquista orgulhosa para o bom funcionamento do programa.

Figura 31: Fórmula da Binarização



&lt;Arquivo Pessoal (2021)&gt;

## 6. CONSIDERAÇÕES FINAIS

O Projeto ENVIRO possui objetivos ousados e complexos em desenvolvimento, mas com propostas bastante pertinentes na área ambiental, onde, se finalizados, podem auxiliar e muito as instituições ambientais a diminuir seus custos. Seu objetivo geral é automatizar o trabalho das organizações responsáveis, eliminando erros matemáticos, economizando tempo e esforço intelectual dos profissionais.

Dos algoritmos, já é possível realizar a conversão dos pixels da imagem em metros quadrados ( $M^2$ ), mas por enquanto, não está integrado ao site. Também é possível fazer a binarização da imagem.

Na *figura 27*, é apresentado a imagem normal, a que foi enviada ao site, posteriormente convertida em escala de cinza e por último, mas não menos importante, a binarização, ressaltando que no projeto foi utilizada a do tipo 'BINARY\_INV'.

Figura 32: Imagem Original, Imagem com escala de cinza e Imagem Binarizada 'Binary\_INV'

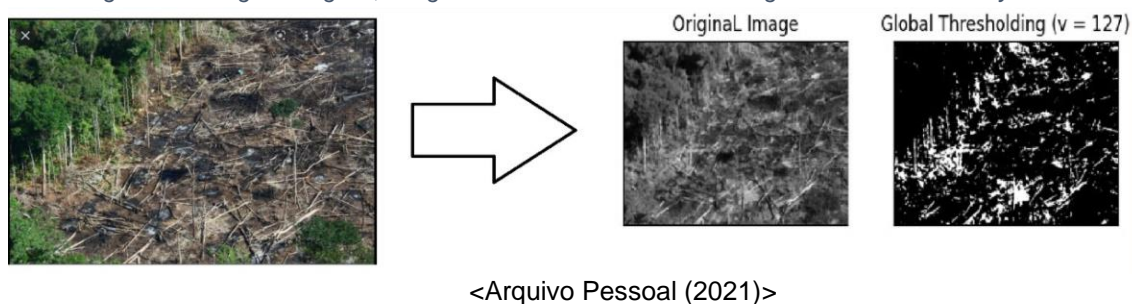
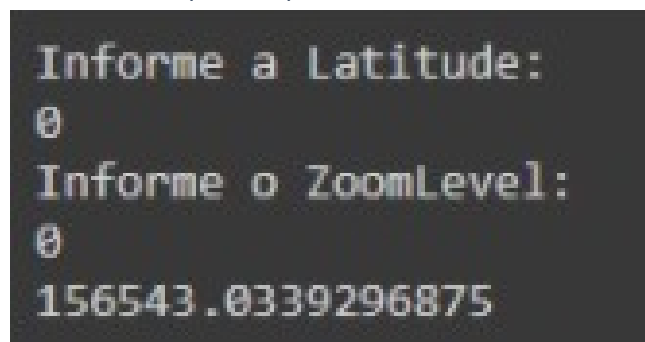


Figura 33: Exemplo de Imagem Binarizada no site



Na imagem 29, é apresentado o resultado em metros quadrados quando a latitude e o nível de zoom são iguais a 0, esses são valores teste que a fórmula apresenta, pois 0 em ambos os valores é a linha do equador.

Figura 34: Conversão para  $M^2$  quando Latitude e Nível de Zoom são 0



## **6.1. Objetivos Alcançados**

Os objetivos alcançados a partir da finalização parcial do projeto foram:

- Diminuição dos riscos de erros matemáticos, bem como de performance, graças a conversão de pixels em metro quadrado.
- Automatizar os processos de cálculos de danos.
- Ter o controle mais preciso e identificação mais rápida das devastações florestais, graças a binarização e ao tipo escolhido da mesma: (BINARY\_INV).

## 7. REFERÊNCIAS BIBLIOGRÁFICAS

ANTONELLO, Ricardo. **Introdução a Visão Computacional com Python e OpenCV**. Diamantina: [s. n.], 06/02/2017. 46 p.

BASTOS, Camila. **Dano Ambiental. Poluição e proteção ao meio ambiente**. Disponível em:  
<<https://camilabastos2.jusbrasil.com.br/artigos/308526908/dano-ambiental>>.  
Acesso em: 18 abr. 2021

BECKER, Bertha Koiffmann. **Geopolítica da Amazônia**. DOSSIÊ AMAZÔNIA BRASILEIRA I. Disponível em: <[https://www.scielo.br/scielo.php?pid=S0103-40142005000100005&script=sci\\_arttext](https://www.scielo.br/scielo.php?pid=S0103-40142005000100005&script=sci_arttext), v. 19, n. 53, 15 jan. 2005.Z> Acesso em: 21 abr. 2021

BINARY UPDATES. **Setup Flask Webserver on Raspberry Pi3**. Youtube, 3 fev. 2019. Disponível em: <<https://www.youtube.com/watch?v=8p5B5sDw3k0>>.  
Acesso em: 20 jun. 2021

CAMPOMORI, Cleber. **Indo Além com API'S Rest**. Disponível em:  
<<https://www.treinaweb.com.br/blog/rest-nao-e-simplesmente-retornar-json-indo-alem-com-apis-rest>>. Acesso em: 16 jul. 2021

CASTIGLIONI, Matheus. **[BR][NodeJS] Back-end - Implementando autenticação de usuários**. Youtube, 08 ago. 2019. Disponível em:  
<<https://www.youtube.com/watch?v=rFnWwl-zE5M>>. Acesso em: 26 jul. 2021

DANTAS, Carolina. **Amazônia tem 1 ° semestre de 2021 com maior área sob alerta de desmate em 6 anos**. Disponível em:  
<<https://g1.globo.com/natureza/amazonia/noticia/2021/07/02/primeiro-semester-de-2021-temo-maior-numero-de-alertas-de-desmatamento-na-amazonia-em-6-anos.ghtml>>. Acesso em: 20 jun.2021

Data Science Academy. **O que é Visão Computacional ?**. Disponível em:  
<<https://blog.dsacademy.com.br/o-que-e-visao-computacional/>>. Acesso em: 15 mar. 2021.

Debug Everything. **Programação em Python – Criando uma API com Flask**. Disponível em: <<https://blog.debugeverything.com/pt/programacao-em-python-aplicativo-flask/>>. Acesso em: 07 jul. 2021

DESLIGAMENTO de supercomputador do Inpe pode trazer consequências para economia e saúde. **Jornal da USP**, São Paulo, 11 jun. 2021. Atualidades. Disponível em: <<https://jornal.usp.br/atualidades/desligamento-de-supercomputador-do-inpe-pode-trazer-consequencias-para-economia-e-para-saude/>>. Acesso em: 27 ago. 2021

DevMedia. **Imagem dentro de DIV javascript**. Disponível em: <<https://www.devmedia.com.br/forum/resolvido-imagem-dentro-de-div-javascript/567195>>. Acesso em 10 ago. 2021

Geeks for Geeks. **How to rotate an image using Python ?**. Disponível em: <<https://www.geeksforgeeks.org/how-to-rotate-an-image-using-python/>>. Acesso em: 14 out. 2021

HERBERT, Anthony. **Como processar dados de solicitação de entrada no Flask**. Disponível em: <<https://www.digitalocean.com/community/tutorials/processing-incoming-request-data-in-flask-pt>>. Acesso em: 04 jul. 2021

INTERFACE de programação de aplicações. **Red Hat**, 18 ago. 2018. Disponível em: <<https://www.redhat.com/pt-br/topics/api/what-are-application-programming-interfaces>>. Acesso em: 22 ago. 2021

KDnuggets. **Introduction to Image Segmentation with K-Means clustering**. Disponível em: <<https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html>>. Acesso em: 14 jul. 2021

LIMA, Victor. **Curso de Node.js - Cadastrando usuários no banco de dados #56**. Youtube, 5 dez. 2018. Disponível em: <<https://www.youtube.com/watch?v=YPkv701xk78>>. Acesso em: 30 jul. 2021

LIMA, Victor. **Curso de Node.js - Registro de usuários #54**. Youtube, 3 dez. 2018. Disponível em: <<https://www.youtube.com/watch?v=acCEOBM2eC0>>. Acesso em: 30 jul. 2021

LUIZ, Thiago. **Processamento Digital de Imagens utilizando Python**. [S. l.], 6 ago. 2019. Disponível em: <<https://medium.com/@thiagoluiz.nunes/relat%C3%B3rio-sobre-o->

processamento-digital-de-imagens-utilizando-python-c7936143d940>. Acesso em: 9 mar. 2021

MACHADO, Pedro. **Register and Login Tutorial | ReactJS, NodeJS, MySQL - Cookies, Sessions, Hashing**. Youtube, 17 set. 2020. Disponível em: <<https://www.youtube.com/watch?v=sTHWNPVNvm8>>. Acesso em: 08 ago. 2021

MDN Web Docs. **Trabalhando com JSON**. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Learn/JavaScript/Objects/JSON>>. Acesso em: 04 jul. 2021

MÉTODOS de requisição HTTP. **MDN Web Docs**, 2021. Disponível em: <<https://developer.mozilla.org/pt-BR/docs/Web/HTTP/Methods>>. Acesso em: 28 ago. 2021

NOGUEIRA, Salvador. **Agência Espacial Brasileira zera orçamento do Inpe para pesquisa em 2021**. Disponível em: <<https://valor.globo.com/brasil/noticia/2020/08/17/agenciaespacial-brasileira-zera-orcamento-do-inpe-para-pesquisa-em-2021.ghtml>>. Acesso em: 17 mar. 2021

NOLETO, Cairo. JSON: o que é, qual sua importância e por que usar? **Betrybe**, 2020. Disponível em: <<https://blog.betrybe.com/tecnologia/json/>>. Acesso em: 28 ago. 2021

OLIVEIRA, João. **Upload de arquivos e imagens com PHP e MySQL [TUTORIAL]**. Youtube, 4 ago. 2015. Disponível em: <<https://www.youtube.com/watch?v=iNm86iordCQ>>. Acesso em: 12 set. 2021

PINHEIRO, Fagner. **Consumindo APIs com Python – Parte 1: O que são APIs?** Disponível em: <<https://www.treinaweb.com.br/blog/consumindo-apis-com-python-parte-1/>> Acesso em: 14 mar. 2021

Portal INPE. **Geração de Imagens**. Disponível em: <<http://www.dgi.inpe.br/>>. Acesso em: 10 mar. 2021

Portal INPE. **INPE - Instituto Nacional de Pesquisas Espaciais**. Disponível em: <<http://www.inpe.br/>>. Acesso em: 10 mar. 2021



Portal OPENCADD. **Discover Page Computer Vision Conf – OPENCADD**. Disponível em: <<https://bityli.com/hfygi>>. Acesso em: 10 mar. 2021

QASStack. **How can I save an image with PIL?**. Disponível em: <<https://qastack.com.br/programming/14452824/>>. Acesso em: 14 ago. 2021

RICCIERI, Thiago. **Sistema de Upload de Arquivos com PHP | Como fazer Passo a Passo**. Youtube, 28 jun. 2020. Disponível em: <<https://www.youtube.com/watch?v=edif-iOEans>>. Acesso em: 12 set. 2021

SANTANA, Felipe. DATA ANALYSIS. In: **Processamento de Imagens com Python - Tutorial Completo**. [S. l.], 30 jan. 2020. Disponível em: <<https://minerandodados.com.br/processamento-de-imagens-com-python/>>. Acesso em: 16 mar. 2021

WIEDERER, Christine. **Processamento de imagens em JavaScript**. [S. l.]: IMasters, 7 mar. 2017. Disponível em: <<https://imasters.com.br/front-end/processamento-de-imagens-em-javascript>>. Acesso em: 17 ago. 2021

WILIAN, João. Flask: o que é e como codar com esse micro framework Python. **GeekHunter**, 2020. Disponível em: <[https://blog.geekhunter.com.br/flask-framework-python/#O\\_que\\_e\\_o\\_Flask](https://blog.geekhunter.com.br/flask-framework-python/#O_que_e_o_Flask)>. Acesso em: 28 ago. 2021