

UNIVERSIDADE FEDERAL DA GRANDE DOURADOS

Rafael Ernesto Lopes

FERRAMENTA PARA EXTRAIR DE COORDENADAS REAIS DE  
IMAGENS DIGITAIS

Dourados

2021

Rafael Ernesto Lopes

FERRAMENTA PARA EXTRAIR DE COORDENADAS REAIS DE  
IMAGENS DIGITAIS

Trabalho de Conclusão de Curso de graduação  
apresentado para obtenção do título de Bacharel  
em Sistemas de Informação.

Faculdade de Ciências Exatas e Tecnologia  
Universidade Federal da Grande Dourados.

Orientador: Prof. Dr. Wellington Lima dos Santos

Orientador: Prof. Dr. Wellington Lima dos Santos

Dourados

2021

Rafael Ernesto Lopes

Trabalho de Conclusão de Curso de graduação  
apresentado para obtenção do título de Bacharel  
em Sistemas de Informação.

Faculdade de Ciências Exatas e Tecnologia  
Universidade Federal da Grande Dourados.

Orientador: Prof. Dr. Wellington Lima dos Santos



---

Orientador: Prof. Dr. Wellington Lima dos Santos  
FACET - UFGD



---

Professora: Profa. Ma. Claudia Regina Tinós Peviani  
FACET - UFGD



---

Professor: Prof. Me. Anderson Bessa da Costa  
FACET - UFGD

Dourados, 12 de Maio de 2021.

## **AGRADECIMENTOS**

À Faculdade de Ciências Exatas e Tecnologia(FACET-UFGD) e a todos os seus professores pela dedicação e à qualidade do ensino.

Ao professor Wellington Lima dos Santos, pela orientação e ao tempo dedicado a este trabalho.

À minha família e amigos, pelo incentivo e bons conselhos.

A Deus, pelo que foi citado acima e muitas outras coisas.

## RESUMO

Neste trabalho, objetivou-se o desenvolvimento de um *digitizer*, na forma de aplicação *web*, para extrair coordenadas de mundo a partir de imagens de gráficos do tipo XY com eixos cartesianos representados em escalas lineares. A análise de 13 ferramentas *digitizers* serviu de base para a implementação de algumas funcionalidades como a exibição dos parâmetros estatísticos associados à definição de escala, bem como melhorias na definição dos pontos de controle, o que contribui para configuração de escalas mais confiáveis e, conseqüentemente, resultados mais precisos. O modelo matemático usado para ajustar as equações de escala foi o mesmo usado para realizar transformações geométricas lineares afins no plano e, para validá-lo, procedeu-se à digitalização de uma circunferência de raio unitário contida numa imagem transformada por cisalhamento e depois por rotação. A análise estatística mostrou ótimos resultados, o que permitir concluir que o modelo matemático empregado foi usado corretamente.

Palavras-chave: *Digitizer*. Imagem digital. Digitalização de dados. Transformações lineares.

## ABSTRACT

*In this work, the objective was to develop a digitizer, in the form of web application, to extract world coordinates from images of XY-type graphs with Cartesian axes represented in linear scales. The analysis of 13 digitizing tools served as a basis for the implementation of some functionalities such as the display of statistical parameters associated with the definition of scale, as well as improvements in the definition of control points, which contributes to the configuration of more reliable scales and, consequently, more results accurate. The mathematical model used to adjust the scale equations was the same to perform linear geometric transformations affine in the plane and, to validate it, a unit radius circumference contained in an image transformed by shear and then by rotation was digitized. The statistical analysis showed excellent results, which allow us to conclude that the mathematical model was used correctly.*

**Key-words:** Digitizer. Digital image. Digitization of data. Linear transformations.

## LISTA DE ILUSTRAÇÕES

Figura 2.1 – Fluxograma de operação de um <i>digitizer</i> . . . . .	15
Figura 3.1 – Gráfico multiescalas hipotético e pontos de controle para a configuração de escalas . . . . .	25
Figura 4.1 – Imagem das curvas das funções $f(x) = \sin(x)$ e $f(x) = x \cos(x)$ . . . . .	29
Figura 4.2 – Página inicial . . . . .	30
Figura 4.3 – Imagem carregada no programa . . . . .	30
Figura 4.4 – Menu de opções . . . . .	31
Figura 4.5 – Lente fixa . . . . .	31
Figura 4.6 – Lente móvel sobre a imagem . . . . .	32
Figura 4.7 – Criação de uma nova escala . . . . .	33
Figura 4.8 – Adicionando pontos de configuração da escala . . . . .	33
Figura 4.9 – Exibição das estatísticas de erros da escala . . . . .	34
Figura 4.10–Escala configurada . . . . .	34
Figura 4.11–Selecionando pontos da imagem . . . . .	35
Figura 4.12–Coordenadas dos pontos extraídos . . . . .	35
Figura 4.13–(a) imagem de uma circunferência de raio unitário; (b) imagem transformada e pontos de controle; (c) pontos de extração de coordenadas . . . . .	36

## LISTA DE TABELAS

Tabela 2.1 – Comparação entre sistemas . . . . .	20
Tabela 4.1 – Resultados da extração das coordenadas de 12 pontos sobre uma circunferência de raio unitário . . . . .	37



## LISTA DE ABREVIATURAS E SIGLAS

CSV	<i>Comma-separated values</i>
XML	<i>Extensible Markup Language</i>
HTML	<i>HyperText Markup Language</i>
TXT	<i>Text File</i>
JSON	<i>JavaScript Object Notation</i>
CSS	<i>Cascading Style Sheets</i>
DOM	<i>Document Object Model</i>

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
1.1	DESCRIÇÃO DO PROBLEMA	12
1.2	JUSTIFICATIVA	13
1.3	OBJETIVOS	14
<b>1.3.1</b>	<b>Objetivo geral</b>	14
<b>1.3.2</b>	<b>Objetivos específicos</b>	14
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>15</b>
2.1	Ferramentas semelhantes	15
<b>2.1.1</b>	<b>WebPlotDigitizer</b>	16
<b>2.1.2</b>	<b>GraphReader</b>	16
<b>2.1.3</b>	<b>GetData Graph Digitizer</b>	17
<b>2.1.4</b>	<b>DataThief</b>	17
<b>2.1.5</b>	<b>DigitizeIt</b>	17
<b>2.1.6</b>	<b>PlotDigitizer</b>	18
<b>2.1.7</b>	<b>Engauge-digitizer</b>	18
<b>2.1.8</b>	<b>Graphixy</b>	18
<b>2.1.9</b>	<b>XYit</b>	18
<b>2.1.10</b>	<b>OriginLab</b>	18
<b>2.1.11</b>	<b>ImageJ</b>	19
<b>2.1.12</b>	<b>GraphGrabber</b>	19
<b>2.1.13</b>	<b>UN-Scan-it</b>	19
2.2	Comparativo entre as ferramentas	19
2.3	Conversão de coordenadas de imagem para coordenadas reais	20
<b>2.3.1</b>	<b>Transformações geométricas no plano</b>	21
<b>2.3.2</b>	<b>Obtenção das equações de conversão</b>	22
<b>2.3.3</b>	<b>Medida da qualidade do ajuste das equações de conversão de escalas</b>	23
<b>3</b>	<b>Metodologia</b>	<b>25</b>
3.1	Ajuste ou configuração de escalas	25
3.2	Ferramentas de desenvolvimento e tecnologias utilizadas	26
<b>3.2.1</b>	<b>Visual Studio Code</b>	26
<b>3.2.2</b>	<b>Chrome</b>	26

3.2.3	<b>Javascript</b> . . . . .	27
3.2.4	<b>HTML</b> . . . . .	27
3.2.5	<b>CSS</b> . . . . .	27
3.2.6	<b>Bootstrap</b> . . . . .	28
3.2.7	<b>JQuery</b> . . . . .	28
4	<b>RESULTADOS E DISCUSSÕES</b> . . . . .	<b>29</b>
4.1	Programa desenvolvido e funcionalidades . . . . .	29
4.1.1	<b>Visão Geral</b> . . . . .	30
4.1.2	<b>Lente Fixa e Lente Móvel</b> . . . . .	31
4.1.3	<b>Escalas e Sequências de Coordenadas</b> . . . . .	32
4.1.4	<b>Seleção dos pontos de dados</b> . . . . .	34
4.2	Análise dos resultados . . . . .	35
5	<b>CONCLUSÃO</b> . . . . .	<b>38</b>
5.1	Trabalhos futuros . . . . .	39
	<b>REFERÊNCIAS</b> . . . . .	<b>40</b>

## 1 INTRODUÇÃO

O avanço da tecnologia da informação e a larga utilização de serviços digitais por vários setores da sociedade fizeram com que o mundo entrasse em uma era onde os dados são considerados uma fonte muito importante de informação e valor para as empresas, governos, instituições e indivíduos. “Para a indústria moderna, os dados gerados por máquinas e dispositivos, soluções em nuvem, gerenciamento de negócios e etc, atingiu anualmente um volume total de 1000 *Exabytes* e é esperado que aumente 20 vezes nos próximos 10 anos”. (KAYNAK; YIN, 2015)

O gigantesco volume de dados produzido diariamente encontra utilização em várias atividades tais como pesquisas acadêmicas, campanhas publicitárias, formulação de políticas públicas, estratégias de negócios, mercado financeiro, notícias e muitas outras coisas.

De uma perspectiva da ciência social, esta conexão leva diretamente para uma nova era de urbanismo inteligente, onde atores humanos, e.g., cidadãos, exploram dados processados em uma forma visual para a melhoria de sua qualidade de vida. (LYTRAS; VISVIZI, 2019)

Tamanho é a importância dos dados nos dias atuais que até mesmo novas profissões para lidar com esses dados surgiram tais como Analista de *Business Intelligence*, Analista de Dados, Cientista de Dados, Engenheiro de Dados entre outras. Estes profissionais desempenham papéis de relevância para as empresas, pois o valor dos dados tende a crescer ao longo dos anos.

O rendimento mundial do mercado de *big data* é projetado para crescer de \$35B em 2017 para \$103B em 2027 - quase triplicando seu valor, enquanto dobra seu valor na metade do intervalo, em 2022 (\$70B).(OGREAN, 2018)

Com tantos dados e informações a serem extraídas e apresentadas, uma das formas mais comuns de representar esses dados é a utilização de gráficos de linha ou curva para relacionar as variáveis necessárias e então apresentar o resultado da análise dos dados. Dentre os muitos exemplos do que pode ser feito a partir desses modelos estão a obtenção de índices de criminalidade em uma cidade, os níveis de poluição, os preços dos imóveis e as tendências de mercado.

Durante a confecção desses gráficos é bastante comum o uso de softwares tais como *MatLab*, *Power BI* e *Tableau*, que recebem os dados e, a partir de certas funções matemáticas ou modelos, relacionam esse conjunto de dados a fim de gerar uma grande variedade de tipos de gráficos.

A utilização de gráficos para representar informações extraídas de dados, técnica conhe-

cida como visualização, evoluiu muito nas últimas décadas. “Durante o último quarto do século 20, a visualização de dados floresceu em uma área mais madura, vibrante e multidisciplinar”, (FRIENDLY et al., 2008). A popularidade do uso de gráficos deve-se principalmente à dificuldade de o ser humano analisar grandes quantidades de dados condensados em tabelas ou texto, enquanto que o sistema visual humano é muito mais avançado, permitindo um entendimento melhor e mais rápido. Segundo Keahey(2013):

A visualização explora o sistema visual humano para prover uma forma intuitiva, imediata e independente de linguagem para ver e mostrar seus dados. Esta é uma ferramenta essencial para o entendimento da informação. O sistema visual humano é sem dúvida o mais rico, imediato e com a maior capacidade de inserir informação na mente humana. A quantidade da capacidade do cérebro dedicada ao processamento visual excede em muitas vezes qualquer outro sentido humano. Algumas estimativas científicas sugerem que o sistema visual humano é capaz de processar aproximadamente 9 *megabits* de informação por segundo, o que corresponde a algo próximo de 1 milhão de letras de texto por segundo. (KEAHEY, 2013).

Embora o entendimento dos dados em um gráfico seja mais fácil, pode-se deparar com a necessidade de extrair valores de alguns pontos desses gráficos, mas nem sempre a função ou modelo que gerou suas curvas está disponível, ou a quantidade de pontos a serem analisados pode ser tão grande que fazer isso de forma manual pode ser bastante improdutivo, sobretudo propenso a erros.

Para resolver este problema existem ferramentas<sup>1</sup> categorizadas como *digitizers*, que são capazes de extrair as coordenadas de um gráfico contido numa imagem digital. O modo de operar e configurar estes programas pode variar bastante, mas geralmente consiste em o usuário carregar a imagem do gráfico, configurar uma escala de conversão e selecionar os pontos desejados. Idealmente, as coordenadas destes pontos podem ser copiadas para a área de transferência, impressas ou exportadas para diferentes formatos de arquivos.

## 1.1 DESCRIÇÃO DO PROBLEMA

Uma prospecção na internet revelou que apesar de existirem várias ferramentas capazes de extrair coordenadas de imagens de gráficos, muitas delas possuem limitações, como o fato de que foram desenvolvidas para serem instaladas na máquina do usuário, tornando-as restritas a determinada(s) plataforma(s).

Verificou-se que a maioria dessas ferramentas apresenta, em ou maior ou menor grau,

<sup>1</sup> Softwares com finalidade específica ou estrita.

deficiências como interface pouco intuitiva, dificuldade de instalação e configuração, além da ausência ou deficiência de funcionalidades que permitem a obtenção de coordenadas com mais precisão e agilidade.

Uma situação comum é a apresentação de informações (curvas, barras ou pontos) em escalas diferentes num mesmo gráfico, mas são poucas as aplicações que permitem associar ou configurar mais de uma escala. Embora essa limitação possa ser contornada ajustando-se uma nova escala noutra sessão do programa, não é o desejável.

Um aspecto importante a ser considerado é a imagem digital, por ser o principal dado de entrada. Imagens digitais obtidas por meio de *scanners* geralmente apresentam um pequeno ângulo de inclinação, enquanto que imagens obtidas via câmera de dispositivos como *smartphones* tendem a apresentar ângulos de inclinação maiores, além de distorções de perspectiva. Assim, a imagem de um retângulo passa a ser um quadrilátero com os quatro lados distintos. Nem todas as aplicações apresentam de forma transparente o controle dos erros decorrentes destas situações.

Outra limitação verificada diz respeito à usabilidade, pois o processo de extração de coordenadas requer cliques precisos, os quais exigem um bom controle motor e visual do usuário, mas nem sempre estes programas incorporam funcionalidades que tornam este procedimento mais preciso e produtivo. Acrescente-se ainda a falta de suporte à língua portuguesa, o que pode ser visto como um obstáculo para muitos usuários que não têm uma proficiência mínima em inglês.

Observadas estas limitações, e considerando a percepção de que algumas melhorias importantes não são difíceis de serem implementadas, são apresentadas, a seguir, as justificativas para a realização do presente trabalho, que consiste em implementar um *digitizer* com funcionalidades simples, porém eficazes.

## 1.2 JUSTIFICATIVA

Ao propor a implementação de um *digitizer* executável em navegadores *web*, o presente trabalho se justifica, na medida em que elimina os problemas de portabilidade entre plataformas, dispensando a instalação local e, principalmente, poupando o usuário de preocupações com vírus e outros *malwares*.

Também se justifica por propor a incorporação de funcionalidades ausentes ou deficientes nas aplicações avaliadas como, por exemplo, a configuração de múltiplas escalas para um mesmo gráfico e a melhoria na precisão do clique.

Por fim, uma justificativa ainda mais relevante é a proposição de um controle estatístico dos erros associados ao ajuste das equações de escala, o que deverá proporcionar maior transparência e confiabilidade na utilização da aplicação, cuja finalidade é a extração precisa de coordenadas de gráficos.

### 1.3 OBJETIVOS

#### 1.3.1 Objetivo geral

Este trabalho tem por objetivo desenvolver um *digitizer*, na forma de aplicação *web*, para extrair coordenadas de mundo a partir de imagens de gráficos do tipo XY com eixos cartesianos representados em escalas lineares.

#### 1.3.2 Objetivos específicos

Para atingir o objetivo geral, foram estabelecidos os seguintes objetivos específicos:

1. Pesquisar quais ferramentas de extração de coordenadas de imagens estão disponíveis e analisar suas características, visando estabelecer um comparativo entre elas.
2. Desenvolver a definição de múltiplas escalas, bem como facilidades para definir os pontos de controle a elas associados.
3. Elaborar um controle estatístico dos erros associados ao ajuste das equações de escala.
4. Implementar melhorias quanto à precisão na seleção dos pontos de dados.

## 2 FUNDAMENTAÇÃO TEÓRICA

As ferramentas que extraem coordenadas a partir de imagens de gráficos são conhecidas como *digitizers*. Segundo Rogers e Dawson(1979):

Um *digitizer* é uma máquina que permite que informações gráficas ou pictóricas sejam transformadas em informações numéricas. A informação numérica pode então ser manipulada estatisticamente, aritmeticamente ou logicamente, para analisar o gráfico original. (ROGERS; DAWSON, 1979)

Há uma grande variedade e disponibilidade deste tipo de software, desde os mais simples aos mais sofisticados, *desktop* ou *web*, gratuitos ou pagos. Neste capítulo são apresentadas algumas das ferramentas disponíveis com a finalidade de estabelecer um comparativo entre as principais características de cada uma delas.

### 2.1 FERRAMENTAS SEMELHANTES

Apesar de existirem diferenças entre as ferramentas com relação às plataformas de uso, aos recursos e à facilidade de utilização, de modo geral, elas compartilham características comuns ou semelhantes quanto ao modo de operação, que consiste em três etapas, quais sejam, leitura da imagem, configuração da escala e extração das coordenadas.

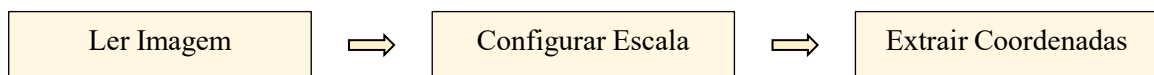


Figura 2.1 – Fluxograma de operação de um *digitizer*

Primeiro, é necessário carregar uma imagem, geralmente contida num arquivo local, mas que também pode ser colada da área de transferência, adquirida de um *scanner* ou mesmo de uma captura de tela de outro programa. Sobre a imagem, o usuário deve definir ou configurar uma escala para converter as coordenadas da imagem em coordenadas do gráfico, também denominadas coordenadas de mundo, ou coordenadas reais. A configuração de escala é feita clicando-se sobre alguns pontos da imagem, denominados pontos de controle, nos quais as coordenadas do gráfico podem ser lidas direta e inequivocamente pelo usuário, uma vez que estes valores são usados na obtenção dos parâmetros das equações de conversão entre as coordenadas da tela e as coordenadas de mundo.

Os parâmetros das equações de conversão entre as escalas são obtidos por meio de regressão linear. Com a escala definida ou configurada, o usuário passa então a selecionar ou



clicar sobre os pontos de dados, que são pontos do gráfico cujas coordenadas se deseja obter ou extrair. Algumas das ferramentas possuem modos automáticos de seleção de dados, nos quais o usuário seleciona regiões do gráfico e a ferramenta se encarrega de capturar vários pontos da região selecionada.

Para a realização da pesquisa sobre as ferramentas semelhantes alguns parâmetros foram estabelecidos, tais como: 1) a ferramenta deve estar disponível em inglês ou português; 2) quando não for gratuita, deve disponibilizar ao menos algum tempo de teste; 3) deve ser executada em computadores, o que exclui da pesquisa quaisquer ferramentas construídas especialmente para outro tipo de plataforma; e 4) não precisa ser exclusivamente um *digitizer*, ou seja, pode oferecer outras funcionalidades.

### 2.1.1 WebPlotDigitizer

O WebPlotDigitizer<sup>1</sup> é uma ferramenta de código aberto que funciona em navegador *web* ou *desktop*, está disponível em inglês, alemão, francês e chinês simplificado. Apesar de ser muito simples de usar e ter uma seção de ajuda ao usuário muito bem elaborada, observou-se que dois pontos podem ser melhorados. O primeiro é que a lente de *zoom* fica fixa na tela, quando o ideal seria ela acompanhar o cursor do mouse, evitando que o usuário divida a atenção entre o cursor na imagem e a lente no canto da tela, especialmente em telas maiores. O segundo ponto que poderia ser melhorado está relacionado ao posicionamento preciso do cursor do mouse no momento do clique, sobretudo em telas com resoluções cada vez maiores. Uma solução para isso é a possibilidade de, após o clique, o usuário mover cursor com as teclas de direção para ajustar mais precisamente a posição do clique.

### 2.1.2 GraphReader

O GraphReader<sup>2</sup> é uma ferramenta *web* gratuita, disponível apenas em inglês, possui uma tela de configuração não muito intuitiva, mas sua funcionalidade fica mais clara com o passar da utilização. A configuração da escala é diferente de todas as demais, consiste em ajustar um retângulo sobre a área da imagem e então associar coordenadas aos seus vértices, o que nem sempre é prático, especialmente se a imagem tiver alguma rotação. Além disso, a extração dos pontos é feita de forma automática, pois o sistema identifica a curva na imagem, selecionando sobre ela vários pontos cujas coordenadas serão extraídas, meio à revelia do usuário.

<sup>1</sup> <https://automeris.io/WebPlotDigitizer/>

<sup>2</sup> <http://www.graphreader.com/>

### 2.1.3 GetData Graph Digitizer

O GetData Graph Digitizer<sup>3</sup> é um software proprietário *desktop* disponível em 14 idiomas (português não incluso) e pode ser instalado apenas no sistema operacional *Windows*. Apesar de ser um programa comercial, pode-se utilizá-lo gratuitamente durante 21 dias.

A característica que mais se destaca nesta ferramenta é a capacidade de extrair coordenadas de imagens rotacionadas em relação aos eixos da tela. Isso é um diferencial relevante, pois somente imagens produzidas diretamente por softwares são perfeitamente alinhadas, enquanto aquelas obtidas por digitalização ou fotografia sempre têm algum grau de rotação, por menor que seja. Outro destaque deste programa é que ele permite exportar os dados para arquivos com extensões *Text File(TXT)*, *Comma Separated Values(CSV)* e *Extensible Markup Language(XML)*.

### 2.1.4 DataThief

O DataThief<sup>4</sup> é uma ferramenta *desktop* multiplataforma e proprietária, mas o usuário pode testar o software por alguns dias. Após esse período de testes, deve-se adquirir uma licença para continuar a usá-lo. Seu funcionamento é simples e a configuração da escala é feita com apenas três pontos de referência definidos pelo usuário. Dentre as ferramentas citadas nesta seção, a funcionalidade que mais se destaca é a possibilidade de mover o cursor do mouse com as teclas de direção, proporcionando maior precisão na seleção dos pontos.

Apesar desse diferencial importante, um fato negativo a ser citado são alguns *bugs* sérios, que levam o sistema a terminar a sua execução de forma inesperada, obrigando o usuário a recomeçar o trabalho do zero.

### 2.1.5 DigitizeIt

O DigitizeIt<sup>5</sup> é uma ferramenta *desktop* disponível para *Linux*, *Windows* e *MacOSX*, que pode ser testada por 21 dias e, após esse período, deve-se comprar uma licença de uso. Seus pontos de destaque são a possibilidade de configuração de mais de uma escala e a capacidade de extrair coordenadas de gráficos com algum ângulo de inclinação em relação a tela.

<sup>3</sup> <http://getdata-graph-digitizer.com/>

<sup>4</sup> <https://datathief.org/>

<sup>5</sup> <https://www.digitizeit.de/>

### 2.1.6 PlotDigitizer

O PlotDigitizer<sup>6</sup> é uma ferramenta *desktop* multiplataforma, gratuita e com código fonte disponibilizado na internet, cujos diferenciais são a capacidade de extrair coordenadas de gráficos cujos eixos não são perfeitamente paralelos aos da tela, além de permitir que os dados sejam exportados para arquivos XML.

### 2.1.7 Engauge-digitizer

O Engauge Digitizer<sup>7</sup> é uma ferramenta *open source desktop* multiplataforma para *Windows*, *Mac OS* e *Linux* com uma interface bastante simples e tem como diferencial a possibilidade de ajustar curvas aos dados extraídos.

### 2.1.8 Graphixy

O Graphixy<sup>8</sup> é um programa *desktop* de fácil instalação, porém requer uma imagem de tamanho predefinido, o que não é prático, pois pode exigir que o usuário recorra a outro software para redimensionar a imagem antes de utilizá-lo. Apesar disso, há uma funcionalidade interessante, não disponível nas outras ferramentas, que é a possibilidade de desenhar um gráfico com o mouse e extrair as coordenadas de seus pontos. Além das coordenadas, pode-se exportar também o próprio gráfico desenhado.

### 2.1.9 XYit

O XYit<sup>9</sup> é uma ferramenta *desktop* disponível apenas para sistemas *Windows*, que pode ser utilizada gratuitamente por 10 dias e, após este período, deve-se adquirir uma licença de uso. Além da extração de coordenadas de gráficos, ela também é capaz de extrair coordenadas de mapas cartográficos. Embora seja de fácil configuração, o programa apresenta alguns erros tais como problemas ao configurar a escala e ao clicar nos botões da tela.

### 2.1.10 OriginLab

O OriginLab<sup>10</sup> é uma ferramenta proprietária disponível para os sistemas *Windows* e *Mac OS* com muitas funcionalidades, sendo que uma delas é a extração de coordenadas de

<sup>6</sup> <http://plotdigitizer.sourceforge.net/>

<sup>7</sup> <https://markummitchell.github.io/engauge-digitizer/>

<sup>8</sup> <https://sourceforge.net/projects/graphixy/>

<sup>9</sup> <http://geomatix.net/xyit/>

<sup>10</sup> <https://www.originlab.com/origin>

gráficos. Tem uma interface fácil, intuitiva e configuração que não foge ao padrão desse tipo de ferramenta. A funcionalidade que mais se destaca é a capacidade de extrair coordenadas de imagens com algum ângulo de inclinação em relação à tela.

#### 2.1.11 ImageJ

O ImageJ<sup>11</sup> é um software *open source*, disponível nas plataformas *Windows*, *Linux* e *Mac OS*, capaz de trabalhar com uma grande variedade de arquivos e de acordo com o *site* oficial possui 500 *plugins* que permitem ao usuário estender as funcionalidades do software de acordo com a sua necessidade. O ImageJ é um software de processamento de imagens que, dentre suas numerosas funcionalidades, pode operar como um *digitizer*.

#### 2.1.12 GraphGrabber

O GraphGrabber<sup>12</sup> é uma ferramenta *desktop* disponível apenas para sistemas *Windows*, cuja funcionalidade mais interessante é seleção automática de pontos, na qual o usuário seleciona uma região do gráfico que deseja extrair as coordenadas e o software irá extrair-las de vários pontos da região.

#### 2.1.13 UN-Scan-it

O UN-SCAN-IT Graph Digitizer<sup>13</sup> é uma ferramenta *desktop* disponível nas plataformas *Windows* e *Mac OS*, que permite testar de uma versão de demonstração com funcionalidade e recursos limitados. O acesso à todas as funcionalidades requer a aquisição de uma licença de uso.

Antes de extrair as coordenadas do gráfico deve-se selecionar um entre 10 modos de extração, todos automáticos, ou seja, o usuário não pode selecionar os pontos de dados manualmente.

## 2.2 COMPARATIVO ENTRE AS FERRAMENTAS

A Tabela 2.1 exibe um comparativo entre as ferramentas apresentadas, relacionando algumas características importantes que estes softwares podem possuir, ou não. Observe-se

<sup>11</sup> <https://imagej.nih.gov/ij/>

<sup>12</sup> <https://www.quintessa.org/software/downloads-and-demos/graph-grabber-2.0.2>

<sup>13</sup> <https://www.silkscientific.com/graph-digitizer.htm>

que algumas delas são exclusivamente *digitizers*, enquanto outras são softwares que contêm funcionalidades mais gerais como aqueles destinados ao processamento de imagens.

Tabela 2.1 – Comparação entre sistemas

Sistema	Em Português	Web	Software Livre	Múltiplas escalas	Selecionar com teclado	Zoom	Imagens inclinadas
WebPlotDigitizer	Não	Sim	Sim	Sim	Não	Sim	Sim
GraphReader	Não	Sim	Sim	Não	Não	Não	Não
GetData Graph Digitizer	Não	Sim	Sim	Não	Não	Sim	Sim
DataThief	Não	Não	Não	Não	Sim	Sim	Não
DigitizeIt	Não	Não	Sim	Não	Não	Não	Sim
PlotDigitizer	Não	Não	Sim	Não	Não	Não	Sim
Engauge-digitizer	Não	Não	Sim	Sim	Não	Não	Não
Graphixy	Não	Não	Sim	Não	Não	Não	Não
XYit	Não	Não	Não	Não	Não	Não	Sim
OriginLab	Não	Não	Não	Não	Não	Não	Sim
ImageJ	Não	Sim	Sim	Não	Não	Não	Não
GraphGrabber	Não	Não	Não	Não	Não	Não	Não
UN Scan-it	Não	Não	Não	Não	Não	Sim	Não

Como se pode observar na tabela, nem uma das aplicações oferece suporte à língua portuguesa. Quanto à portabilidade, vê-se que apenas quatro *digitizers* (*WebPlotDigitizer*, *GraphReader*, *GetData Graph Digitizer*, *ImageJ*) são aplicações *web*, que hoje são uma tendência, pelas vantagens que proporcionam, especialmente a independência de plataforma e a desnecessidade de instalação local.

Com relação às funcionalidades extras, pode-se observar ainda que apenas duas aplicações (*WebPlotDigitizer*, *Engauge-digitizer*) permitem configurar mais de uma escala e apenas uma delas permite o controle preciso do cursor do mouse por meio das teclas de direção.

Alguns programas disponibilizam a funcionalidade de *zoom* (*WebPlotDigitizer*, *GetData Graph Digitizer*, *DataThief*, *UN Scan-it*), seja ampliando a imagem como um todo (*zoom* global), seja ampliando apenas a região próxima ao cursor do mouse, similarmente ao efeito proporcionado por uma lente de aumento (*zoom* local). Esta região ampliada é geralmente apresentada numa determinada posição da tela, por isso e no contexto deste trabalho, essa funcionalidade de *zoom* é denominada de lente fixa.

### 2.3 CONVERSÃO DE COORDENADAS DE IMAGEM PARA COORDENADAS REAIS

O sistema de coordenadas de uma imagem é um sistema cartesiano bidimensional, cujo eixo das abcissas é orientado da esquerda para a direita e o das ordenadas de cima para baixo. Esse sistema é formado por uma matriz retangular de *pixels*, no qual a abscissa e a ordenada de cada *pixel* são o índice da sua coluna e o índice da sua linha, geralmente numerados a partir de zero.

A representação de um objeto real nesse sistema requer cálculos de projeção no plano e transformações de coordenadas por meio de translação, de rotação e escala, que culminam com o arredondamento das coordenadas para números inteiros. Neste projeto, o interesse é pela transformação inversa, ou seja, dada a imagem de um objeto, objetiva-se mapear as coordenadas dessa imagem nas coordenadas do objeto, comumente denominadas coordenadas de mundo, coordenadas reais, ou ainda, coordenadas globais.

### 2.3.1 Transformações geométricas no plano

Em aplicações gráficas, as transformações lineares geométricas como translação, rotação, mudança de escala e outras são facilitadas empregando-se coordenadas homogêneas (ou projetivas), que são uma representação especial para pontos, vetores e matrizes por meio do acréscimo de uma dimensão. No plano, um ponto de coordenadas  $(x, y)$  é representado como  $(x, y, 1)$  em coordenadas homogêneas e sempre que um cálculo resultar em coordenadas  $(x, y, w)$ , estas devem ser normalizadas para  $(\frac{x}{w}, \frac{y}{w}, 1)$ .

Segundo (HETEM JUNIOR, 2006), ao se empregar coordenadas homogêneas, as equações de transformações geométricas no plano ficam reduzidas à multiplicações de matrizes  $3 \times 3$ , como ilustram as equações (2.1), (2.2) e (2.3), associadas à translação, à rotação e à escala, respectivamente.

$$P' = T(t_x, t_y) \cdot P \rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix} \quad (2.1)$$

$$P' = R(\theta) \cdot P \rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x \cos \theta - y \sin \theta \\ x \sin \theta + y \cos \theta \\ 1 \end{bmatrix} \quad (2.2)$$

$$P' = S(s_x, s_y) \cdot P \rightarrow \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} s_x x \\ s_y y \\ 1 \end{bmatrix} \quad (2.3)$$

Transformações geométricas sequenciais podem ser compostas ou agrupadas em única matriz, que é igual ao produto das matrizes associadas às transformações, na ordem inversa. As-

sim, uma transformação bidimensional genérica composta por uma combinação de translação, rotação e escala pode ser expressa por

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} rs_{xx} & rs_{xy} & trs_x \\ rs_{yx} & rs_{yy} & trs_y \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad (2.4)$$

em que os quatro elementos  $rs_{ij}$  são termos multiplicativos ligados à rotação e escala e os elementos  $trs_i$  combinam translação, rotação e escala.

Além da translação, rotação e escala, existem outras transformações como o cisalhamento em relação ao eixo  $x$  ou ao eixo  $y$  e a reflexão em relação a uma reta. Elas são denominadas transformações afins, e todas elas preservam o paralelismo entre as retas, enquanto translação e rotação também preservam a distância entre quaisquer dois pontos. Toda transformação afim pode ser expressa pelas equações:

$$x' = a_{xx}x + a_{xy}y + b_x \quad (2.5)$$

$$y' = a_{yx}x + a_{yy}y + b_y \quad (2.6)$$

### 2.3.2 Obtenção das equações de conversão

As equações (2.5) e (2.6) também podem ser utilizadas para mudança de sistemas de coordenadas, ou seja, permitem que os dados de um gráfico com eixos  $p$  e  $q$  sejam desenhados na tela, na impressora ou num arquivo de imagem. Como as transformações são lineares, pode-se fazer o inverso, ou seja, calcular as coordenadas  $(p, q)$  em função das coordenadas  $(x, y)$  da imagem, todavia com alguma magnitude de erro, já que as coordenadas da imagem sofrem arredondamentos para números inteiros. A magnitude desse erro é inversamente proporcional à resolução da imagem.

$$p = a_{xx}x + a_{xy}y + b_x \quad (2.7)$$

$$q = a_{yx}x + a_{yy}y + b_y \quad (2.8)$$

As equações (2.7) e (2.8) têm três parâmetros cada, portanto são necessários pelo menos três pontos não colineares para determiná-los. Com três pontos, a determinação dos parâmetros é puramente matemática, por isso é desejável que sejam usados pelo menos quatro pontos para

que a determinação seja feita por um método estatístico de regressão linear, que é baseado na minimização da soma dos quadrados dos resíduos.

Para determinar os parâmetros da equação (2.7), os pontos na imagem  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ , denominados pontos de controle, devem ser mapeados para os correspondentes pontos com coordenadas de mundo  $(p_1, q_1), (p_2, q_2), \dots, (p_n, q_n)$ , cujos valores são fornecidos como dados de entrada. Isso resulta no sistema de equações lineares sobredeterminado (mais equações que incógnitas):

$$\begin{cases} a_{xx}x_1 + a_{xy}y_1 + b_x = p_1 \\ a_{xx}x_2 + a_{xy}y_2 + b_x = p_2 \\ \vdots \\ a_{xx}x_n + a_{xy}y_n + b_x = p_n \end{cases} \quad \text{ou} \quad \begin{bmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ \vdots & \vdots & \vdots \\ x_n & y_n & 1 \end{bmatrix} \cdot \begin{bmatrix} a_{xx} \\ a_{xy} \\ b_x \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \\ \vdots \\ p_n \end{bmatrix} \quad (2.9)$$

A solução de um sistema sobredeterminado,  $Ax = b$ , que minimiza a soma dos quadrados dos resíduos,  $Ax - b$ , é a solução do sistema determinado  $A^T Ax = A^T b$ , ou seja:

$$\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix} \cdot \begin{bmatrix} a_{xx} \\ a_{xy} \\ b_x \end{bmatrix} = \begin{bmatrix} \sum x_i p_i \\ \sum y_i p_i \\ \sum p_i \end{bmatrix}, \quad (2.10)$$

cuja solução pode ser obtida por diversos métodos diretos, inclusive pela Regra de Cramer, cuja complexidade é  $O(N!)$ , o que o torna proibitivo (HABGOOD; AREL, 2012) para sistemas de grande porte, mas factível para sistemas de ordem  $3 \times 3$ , inclusive (VINCE, 2012) o faz para resolver sistemas de ordens  $3 \times 3$  e  $4 \times 4$ . De forma análoga, o sistema de equações que resolve a equação (2.8) é:

$$\begin{bmatrix} \sum x_i^2 & \sum x_i y_i & \sum x_i \\ \sum x_i y_i & \sum y_i^2 & \sum y_i \\ \sum x_i & \sum y_i & n \end{bmatrix} \cdot \begin{bmatrix} a_{yx} \\ a_{yy} \\ b_y \end{bmatrix} = \begin{bmatrix} \sum x_i q_i \\ \sum y_i q_i \\ \sum q_i \end{bmatrix}, \quad (2.11)$$

em que os limites de todos os somatórios nas duas últimas equações são 1 e  $n$ .

### 2.3.3 Medida da qualidade do ajuste das equações de conversão de escalas

Considerando que podem haver pequenas distorções na imagem, especialmente se esta for obtida via *scanner* ou câmera fotográfica, e que o usuário pode informar coordenadas de mundo incorretas, é desejável que se tenha uma noção sobre a qualidade das equações ajustadas



pelas equações (2.10) e (2.11). Os erros absolutos, ou resíduos, que são a diferença entre as coordenadas de mundo calculadas pelas equações (2.7) e (2.8) e as fornecidas  $(p_i, q_i)$  são dadas pelas equações:

$$e_{p_i} = p(x_i, y_i) - p_i \quad (2.12)$$

$$e_{q_i} = q(x_i, y_i) - q_i \quad (2.13)$$

Um medida da qualidade do ajuste em regressões lineares bastante usada é o coeficiente de determinação ( $R^2$ ), sendo que  $0 \leq R^2 \leq 1$ . Quanto mais próximo da unidade for o  $R^2$  melhor é o ajuste, ou seja, mais fielmente a equação ajustada expressa a variação dos dados. Como mostram as equações abaixo, se todos os erros fossem nulos, o numerador da fração seria nulo e o ajuste seria perfeito ( $R^2 = 1$ ).

$$R_p^2 = 1 - \frac{\sum_{i=1}^n e_{p_i}^2}{\left(\sum_{i=1}^n p_i^2\right) - \frac{1}{n} \left(\sum_{i=1}^n p_i\right)^2} \quad (2.14)$$

$$R_q^2 = 1 - \frac{\sum_{i=1}^n e_{q_i}^2}{\left(\sum_{i=1}^n q_i^2\right) - \frac{1}{n} \left(\sum_{i=1}^n q_i\right)^2} \quad (2.15)$$

### 3 METODOLOGIA

A configuração de escala é um etapa sensível na utilização de um *digitizer*, pois, na etapa seguinte, as coordenadas dos pontos seleccionados são totalmente dependentes das equações de escala ajustadas. Porém, o levantamento das ferramentas revelou que raramente são divulgados os métodos usados na configuração de escala, assim como os erros inerentes a esse procedimento. Por esta razão, o próximo item visa ampliar a compreensão do item (2.3.2), no qual a abordagem é mais matemático-estatística.

#### 3.1 AJUSTE OU CONFIGURAÇÃO DE ESCALAS

Considere-se a Figura 3.1, na qual é representada uma imagem de um gráfico com uma escala horizontal ( $p$ ), duas escalas verticais ( $q$  e  $r$ ) e os pontos de controle  $C_1$ ,  $C_2$ ,  $C_3$ ,  $C_4$  e  $C_5$ , usados para ajustar as escalas. Esses pontos de controle devem ser definidos pelo usuário com um clique, em locais da imagem onde as coordenadas reais ( $p, q$ ) e ( $p, r$ ) de cada ponto  $C_i$  podem ser lidas direta e inequivocamente, tal como nas marcas dos eixos coordenados e nos cruzamentos das linhas de grade, pois elas devem ser fornecidas como dados de entrada, após cada clique. As coordenadas de imagem ( $x_i, y_i$ ) desses pontos  $C_i$  são lidas diretamente no momento do clique.

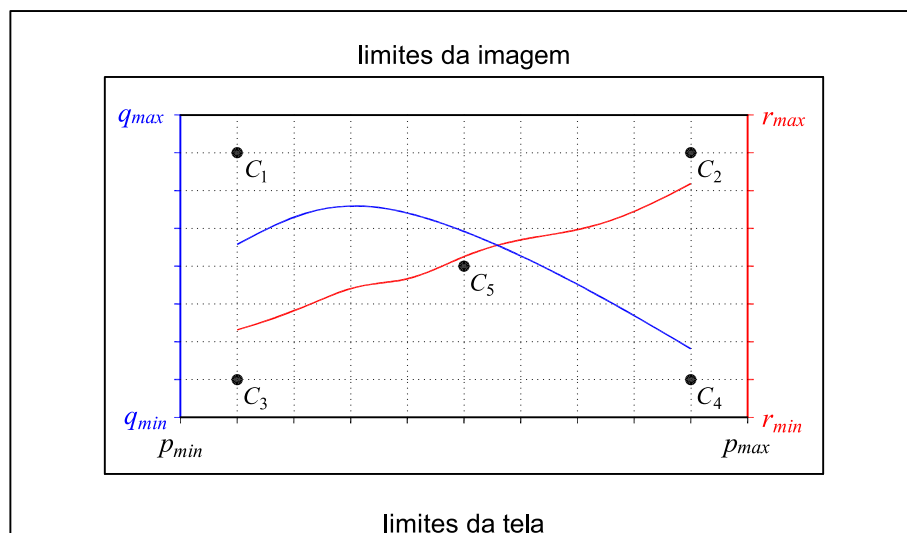


Figura 3.1 – Gráfico multiescalas hipotético e pontos de controle para a configuração de escalas

Como apresentado no item (2.3.2), os parâmetros do par de equações  $p(x, y)$  e  $q(x, y)$  e os parâmetros do par de equações  $p(x, y)$  e  $r(x, y)$  podem ser determinados resolvendo-se os sistemas de equações lineares (2.10) e (2.11). Neste trabalho, utilizou-se a Regra de Cramer para resolver os referidos sistemas. Os erros eventualmente cometidos na digitação das coordenadas

de mundo dos pontos de controle implicarão na redução dos coeficientes de determinação ( $R^2$ ), calculados pelas equações (2.14) e (2.15). A visualização dos erros absolutos, equações (2.13) e (2.12), também permite identificar quais coordenadas apresentam maiores desvios, os quais podem ter sido causados por equívocos na entrada de dados. O *digitizer* desenvolvido neste projeto permite ajustar qualquer número de escalas, que tanto podem estar associadas a um mesmo gráfico como também a gráficos diferentes em uma mesma imagem.

## 3.2 FERRAMENTAS DE DESENVOLVIMENTO E TECNOLOGIAS UTILIZADAS

Para implementar o *digitizer* proposto neste projeto foram utilizadas as ferramentas de desenvolvimento e as tecnologias descritas a seguir, de forma resumida.

### 3.2.1 Visual Studio Code

O *Visual Studio Code*<sup>1</sup> é um editor de código, gratuito, desenvolvido e mantido pela *Microsoft Corporation* e um dos editores mais populares atualmente, o que se deve principalmente à grande quantidade de extensões que podem ser adicionadas a ele, ampliando sua funcionalidade e praticidade. Para Fan et al. (2019):

(...) é notável que o *Visual Studio Code* foi classificado como o ambiente de desenvolvimento mais popular na pesquisa realizada em 2019 pelo *Stack Overflow*, apenas 4 anos após seu lançamento em 2015. IDEs leves são altamente customizáveis, e podem ser significativamente extensíveis com um grande número de *plugins* disponíveis para apoiar várias tarefas de desenvolvimento e linguagens de programação.(FAN et al., 2019)

### 3.2.2 Chrome

O *digitizer* proposto neste trabalho pode ser executado em qualquer navegador *web* moderno, porém, para o seu desenvolvimento utilizou-se o *Chrome*<sup>2</sup>, por duas razões principais: 1) trata-se de um dos navegadores mais utilizados do mundo com uma parcela de 65% do mercado de navegadores (W3COUNTER, 2021); 2) ele disponibiliza uma ferramenta de desenvolvimento chamada de *Chrome Dev Tools* que proporciona várias vantagens como, por exemplo, a depuração de código com bastante facilidade.

---

<sup>1</sup> <https://code.visualstudio.com>

<sup>2</sup> <https://www.google.com/chrome/>

### 3.2.3 Javascript

A Linguagem *Javascript*<sup>3</sup> pode ser interpretada ou compilada em tempo de execução e está disponível em todos os navegadores de internet e também vem sendo utilizada em aplicações do lado do servidor com o uso de ferramentas como o *NodeJs* e aplicativos *mobile* com *frameworks* como o *ReactNative*<sup>4</sup>. É uma das linguagens de programação mais utilizadas do mundo de acordo com o *Índice Tiobe*<sup>5</sup> e possui uma grande comunidade de desenvolvedores e entusiastas ao redor do mundo. Segundo (Ohuru, 2020):

*Javascript* se estabilizou muito bem e agora é a linguagem de programação mais importante que um desenvolvedor deve saber. A versão atual é ECMAScript 2019 ou ECMAScript10. *Javascript* também encontrou um caminho nos sistemas embarcados, dispositivos de IoT, aplicativos móveis, aplicativos para *desktop* e a lista continua. (OHURU, 2020)

### 3.2.4 HTML

O *HyperText Markup Language*(HTML)<sup>6</sup> ou Linguagem de Marcação de Hipertexto é a linguagem utilizada para dar estrutura para as páginas *web*. O hipertexto refere-se aos *links* para outras páginas, os quais possibilitam a navegabilidade entre elas e a marcação refere-se às *tags* utilizadas no HTML para estruturar as páginas *web*, tanto de forma organizacional como de forma semântica.

### 3.2.5 CSS

De acordo com Ndia, Muketha e Omieno (2019):

*Cascading Style Sheets*(CSS)<sup>a</sup> é uma linguagem padrão usada por desenvolvedores *front web* para definir a aparência de documentos estruturados escritos em HTML e XML. CSS é composto por uma sequência de regras de estilo, onde cada regra possui um seletor que seleciona os elementos necessários para estilizar o documento HTML ou XML. (NDIA; MUKETHA; OMIENO, 2019)

<sup>a</sup> <https://www.w3.org/Style/CSS/Overview.en.html>

Dentre as várias capacidades do CSS estão a definição de cor, opacidade, espaçamento entre os elementos que compõem a página, animações e responsividade, que é algo bastante

<sup>3</sup> <https://developer.mozilla.org/en-US/docs/Web/JavaScript>

<sup>4</sup> <https://reactnative.dev/>

<sup>5</sup> <https://www.tiobe.com/tiobe-index/>

<sup>6</sup> <https://developer.mozilla.org/en-US/docs/Web/HTML>

discutido nos dias atuais devido ao uso de dispositivos com diversos tamanhos de tela.

### 3.2.6 Bootstrap

O *Bootstrap*<sup>7</sup> é uma ferramenta desenvolvida pela empresa *Twitter Inc* e é descrito no seu site oficial como um *toolkit* para criar páginas *web* responsivas de forma rápida e ágil, com componentes prontos que podem ser reutilizados para criar as páginas necessárias. Embora seja uma biblioteca bastante robusta, neste trabalho, foram usados apenas os arquivos de CSS do *Bootstrap*. A popularidade do *Bootstrap* deve-se à sua facilidade em criar páginas *web*. Segundo Shah:

Além disso, o *Bootstrap* é fácil de instalar e implementar em um *design web*. *Bootstrap* também fornece vários modelos para um rápido lançamento de um *website*, enquanto a implementação dos métodos dos componentes e os códigos em *JQuery* podem ser copiados da página oficial da documentação. (SHAH, 2015)

### 3.2.7 JQuery

O *JQuery*<sup>8</sup> segundo seu próprio *website* é uma biblioteca *Javascript* que é rápida, pequena e rica em funcionalidades que lidam com a manipulação do *Document Object Model* (DOM), tratamento de eventos, animações e requisições em *Ajax*. Desde seu lançamento foi adotada em vários projetos e por isso é utilizada atualmente em 75.49% dos *websites*, só no Brasil está presente em aproximadamente 700.494 deles (BUILTWITH, 2021).

---

<sup>7</sup> <https://getbootstrap.com/>

<sup>8</sup> <https://jquery.com>

## 4 RESULTADOS E DISCUSSÕES

Primeiro, será apresentada uma descrição geral do *digitizer* implementado e, em seguida, será mostrado um exemplo de utilização do mesmo, juntamente com a análise dos resultados.

### 4.1 PROGRAMA DESENVOLVIDO E FUNCIONALIDADES

O programa desenvolvido pode ser executado em navegadores *web* atualizados e em todos os sistemas operacionais suportados por esses navegadores, que, inclusive, podem estar desconectados da *internet* ou em modo *offline*. Neste modo, as bibliotecas *JQuery* e *Bootstrap* devem estar presentes no dispositivo do usuário, todavia isso não chega a ser um empecilho, já que as duas bibliotecas, na forma minificada, e o programa requerem menos de 500 KB de armazenamento.

Foram implementadas todas as características e funcionalidades usadas na elaboração do quadro comparativo da Tabela 2.1, na qual é possível constatar que nem uma das aplicações atende a todas elas, as quais foram utilizadas por representarem um indicativo objetivo para a avaliação da facilidade de uso dos programas com esta finalidade.

Para a demonstração das funcionalidades do *digitizer* desenvolvido neste trabalho, utilizou-se a Figura 4.1 como exemplo.

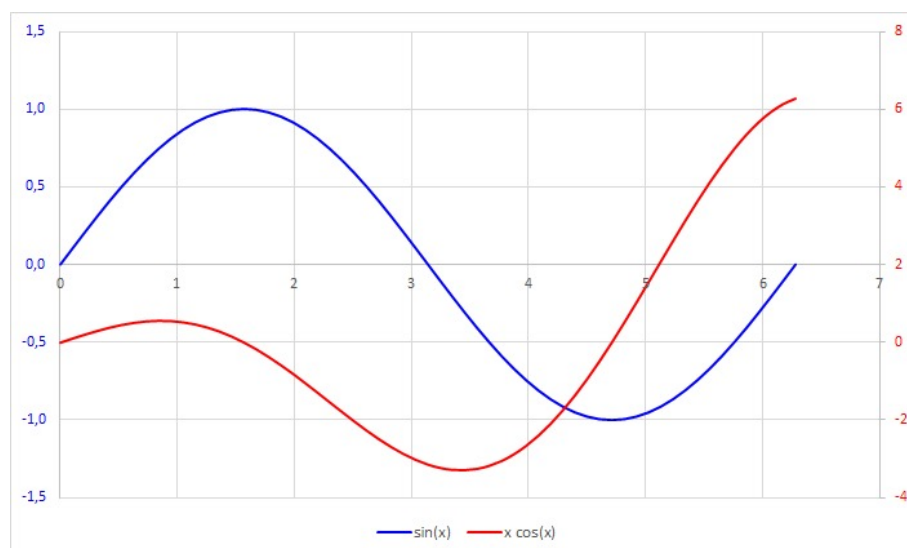


Figura 4.1 – Imagem das curvas das funções  $f(x) = \sin(x)$  e  $f(x) = x \cos(x)$

### 4.1.1 Visão Geral

Na Figura 4.2, vê-se que o programa tem um *layout* simples e limpo, onde a parte superior da tela contém uma barra conhecida como *header*, na qual são exibidas as dimensões da imagem selecionada, as coordenadas de imagem e as respectivas coordenadas reais do cursor do mouse, além de um botão para acesso às coordenadas extraídas e outro botão para acessar o menu de opções do programa.

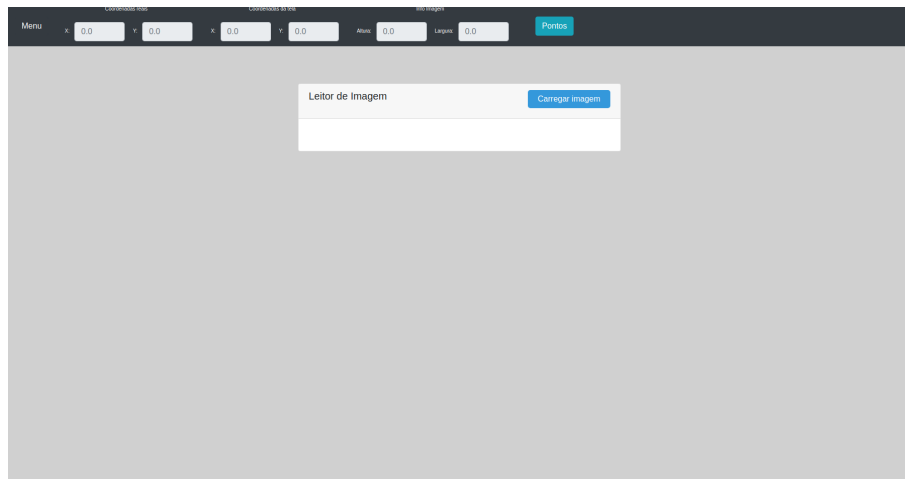


Figura 4.2 – Página inicial

No centro da tela, o botão “Carregar Imagem” permite carregar a imagem de onde as coordenadas serão extraídas. Ao carregar uma imagem, suas dimensões passam a ser exibidas na seção “Info Imagem” do *header* e, ao movimentar o cursor do mouse sobre a imagem, suas coordenadas são exibidas, como mostrado na Figura 4.3.

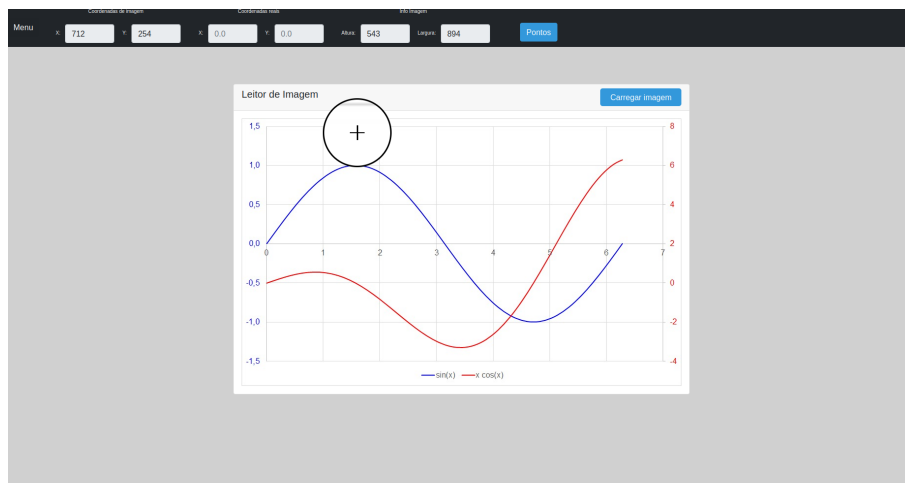


Figura 4.3 – Imagem carregada no programa

Com a imagem carregada, o usuário deve clicar no botão “Menu”, para exibir um painel na lateral esquerda com as seguintes funcionalidades: Lente Lateral, Lente Móvel, Escalas e

Sequência de Coordenadas, como exibe a Figura 4.4.

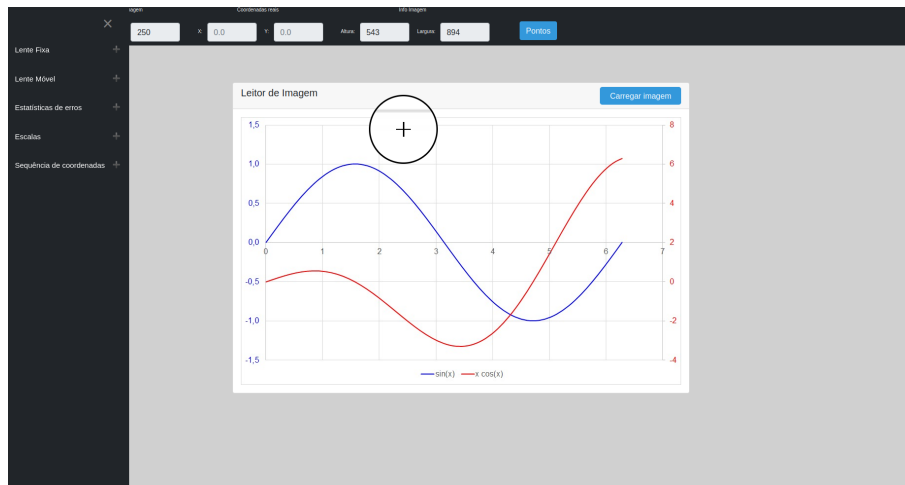


Figura 4.4 – Menu de opções

#### 4.1.2 Lente Fixa e Lente Móvel

Os programas analisados no capítulo 2 que disponibilizam a funcionalidade de *zoom* o fazem ampliando a região do cursor em um algum local fixo da tela, ou ampliando a imagem como um todo, sendo que a segunda forma se mostra menos prática que a primeira.

No programa desenvolvido, a funcionalidade *zoom* foi implementada de duas formas, que foram denominadas lente fixa e lente móvel. A funcionalidade lente fixa consiste em ampliar a região do cursor do *mouse* em uma janela localizada inicialmente no canto superior direito, mas que, diferentemente dos outros programas, pode ser reposicionada na tela. A lente fixa pode ser exibida clicando-se na opção “Lente Fixa”, no painel à esquerda, tal como aparece no lado superior direito da Figura 4.5.

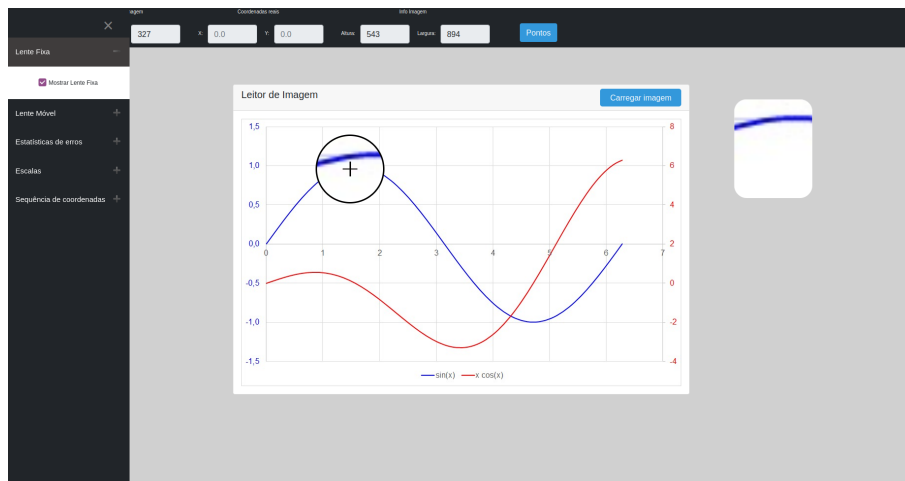


Figura 4.5 – Lente fixa



Além da lente fixa, que é a implementação usual nos programas que disponibilizam a funcionalidade de *zoom* local, também foi implementada a funcionalidade lente móvel, que é um *zoom* local semelhante a uma lupa real, ou seja, a ampliação acompanha o cursor do *mouse*. Este *zoom* local móvel é bem mais confortável para o usuário, pois, em vez de olhar para o cursor do mouse e para a lente fixa, ele foca seu olhar para ambos num mesmo ponto da tela. A lente móvel, cujo tamanho pode ser alterado, é exibida assim que uma imagem for carregada e o cursor for movimentado sobre mesma, Figura 4.6.

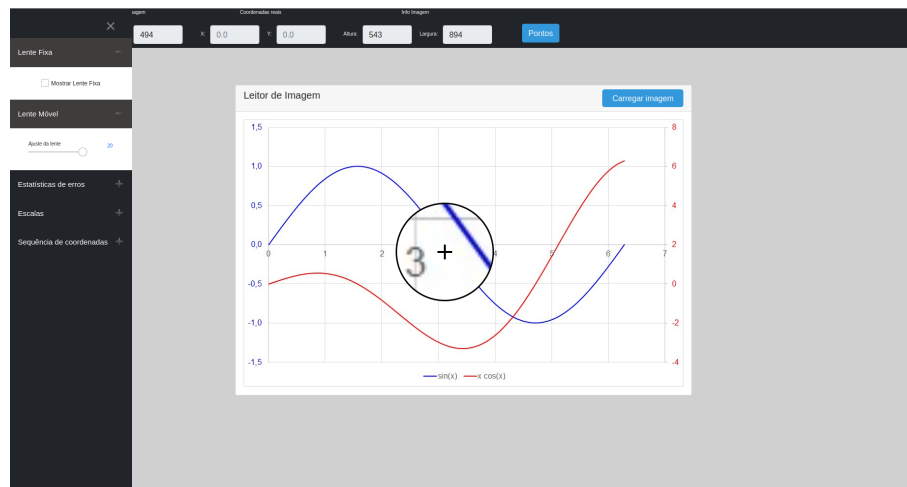


Figura 4.6 – Lente móvel sobre a imagem

#### 4.1.3 Escalas e Sequências de Coordenadas

Antes que se possa extrair coordenadas de uma imagem é necessária a definição de uma escala de referência, procedimento que consiste em ajustar duas equações para converter as coordenadas da imagem em coordenadas reais do gráfico. A definição de uma ou mais escalas é feita clicando-se no item “Nova Escala”, que abre uma caixa de diálogo, na qual pode-se atribuir um nome identificador para a escala, Figura 4.7.

Após atribuir um nome à escala, deve-se clicar em pelo menos quatro pontos de controle, cujos valores das abscissas e das ordenadas reais representadas gráfico devem ser digitados pelo usuário. Por isso é importante selecionar pontos de controle onde a leitura das duas coordenadas possa ser feita direta e inequivocamente, tal acontece nas marcas sobre os eixos coordenados e nos cruzamentos das linhas de grade do gráfico, caso estejam presentes.

Como a posição do clique em cada ponto de controle deve ser a mais precisa possível, além do recurso lente móvel, o usuário também pode fazer o ajuste fino dessa posição com as teclas de direção, finalizando com a tecla Enter, Figura 4.8.



Figura 4.7 – Criação de uma nova escala

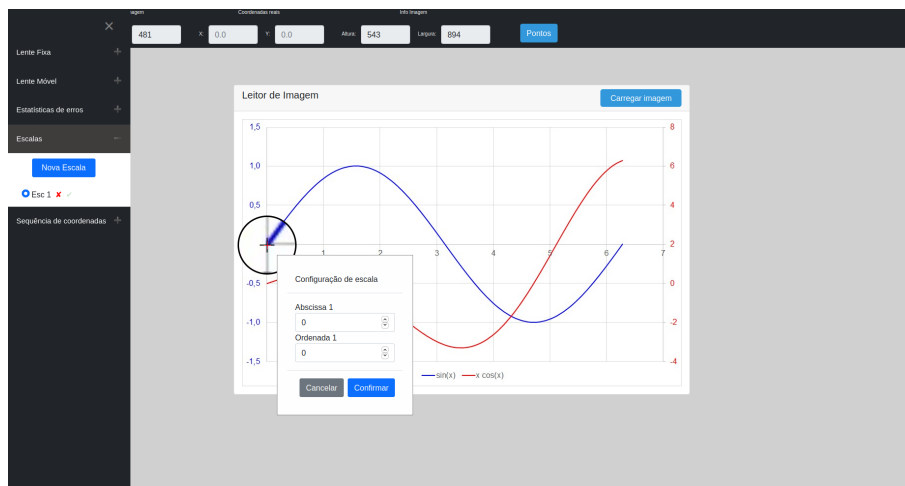


Figura 4.8 – Adicionando pontos de configuração da escala

A precisão desta definição pode ser conferida selecionando a opção “Mostrar Estatísticas de Erros” no menu lateral, como mostra a Figura 4.9, onde serão exibidos os resíduos absolutos e relativos, estes últimos expressos em percentagem, além do coeficiente de determinação ( $R^2$ ). Eventuais erros grosseiros ficarão patentes ao se observar estes parâmetros. Dada a subjetividade do que seria uma escala com precisão aceitável, o programa não recusará uma escala definida com precisão questionável, isso será uma decisão do usuário.

Após marcar os pontos de controle desejados, a definição da escala deve ser concluída, o que pode ser feito clicando-se no botão verde, ao lado do nome da escala. Isso cria uma sequência de coordenadas associada à escala, o qual pode ser visualizada na seção “Sequência de Coordenadas”, no menu lateral. O usuário pode criar mais sequências para a escala clicando no botão “Nova Sequência” que está inserido no *card* da escala e ao lado de cada sequência existe uma caixa de seleção que ativa ou desativa aquela sequência. A Figura 4.10 exibe uma escala concluída com a sua respectiva sequência.

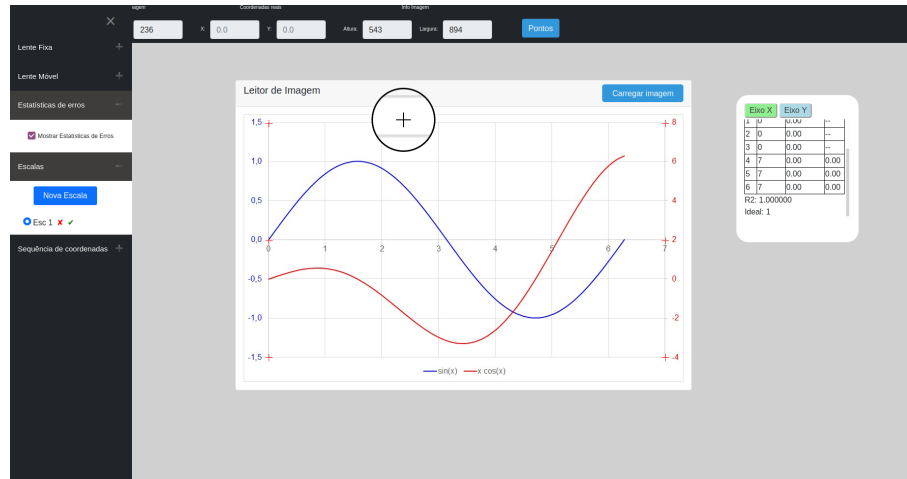


Figura 4.9 – Exibição das estatísticas de erros da escala

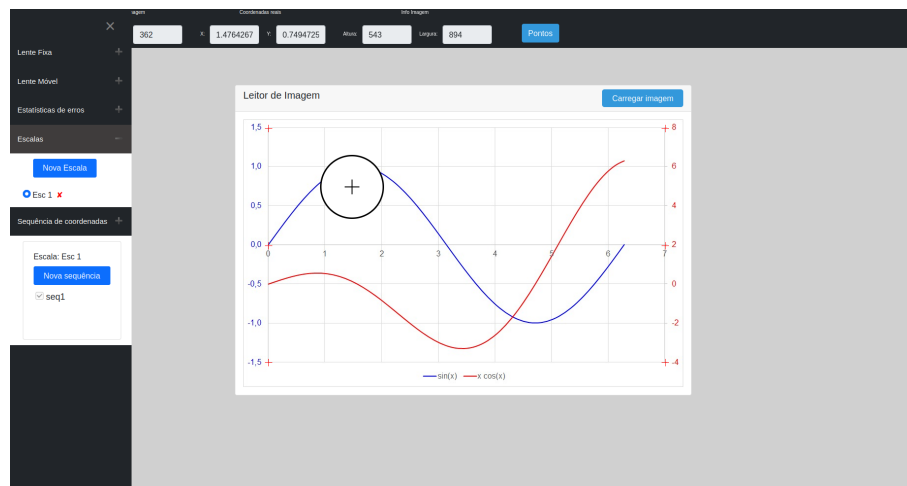


Figura 4.10 – Escala configurada

#### 4.1.4 Seleção dos pontos de dados

A partir de uma escala definida com precisão satisfatória, a precisão das coordenadas dos pontos selecionados passa a depender de o usuário clicar exatamente sobre o pixel desejado na imagem, mas essa tarefa representa uma certa dificuldade, na medida em que é influenciada por fatores como o tamanho e a resolução da tela e, especialmente, a visão e a habilidade motora do usuário. Influenciam ainda os fatores ligados ao *mouse*, como sua ergonomia, a superfície usada e algumas propriedades definidas no sistema operacional como velocidade e rastros.

Considerando essa dificuldade inerente ao processo de seleção dos pontos, não tratada nos programas analisados, implementou-se uma solução nesse sentido, a qual consiste em permitir que um ajuste fino da posição possa ser feito com as teclas de direção e Enter, após o clique do *mouse*. A Figura 4.11 exibe pontos selecionados sobre o gráfico.

Os pontos selecionados pelo usuário são exibidos sobre a imagem na cor verde e com um número ao lado de cada ponto referente a ordem de seleção, enquanto suas coordenadas

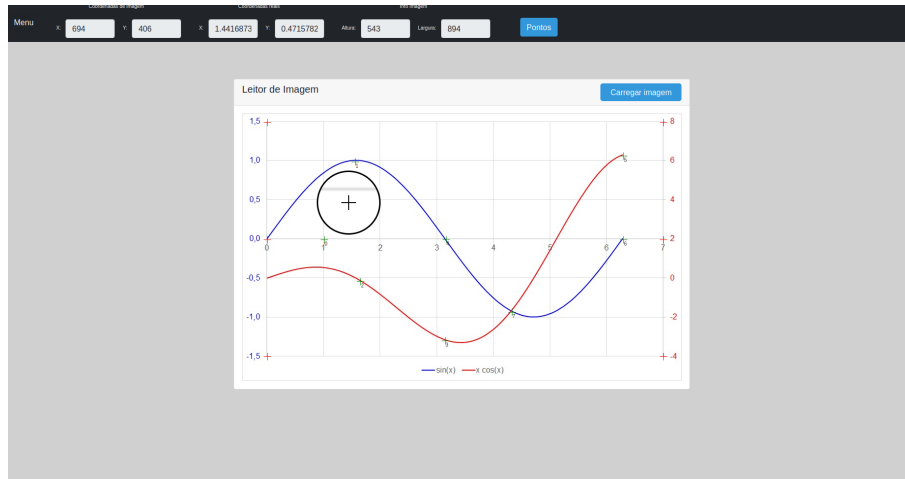


Figura 4.11 – Selecionando pontos da imagem

são armazenadas nas sequências de coordenadas referentes à escala atual e que estejam ativas no momento da seleção dos pontos. As coordenadas podem ser visualizadas com um clique no botão “Pontos”, no *header* do programa, Figura 4.12.

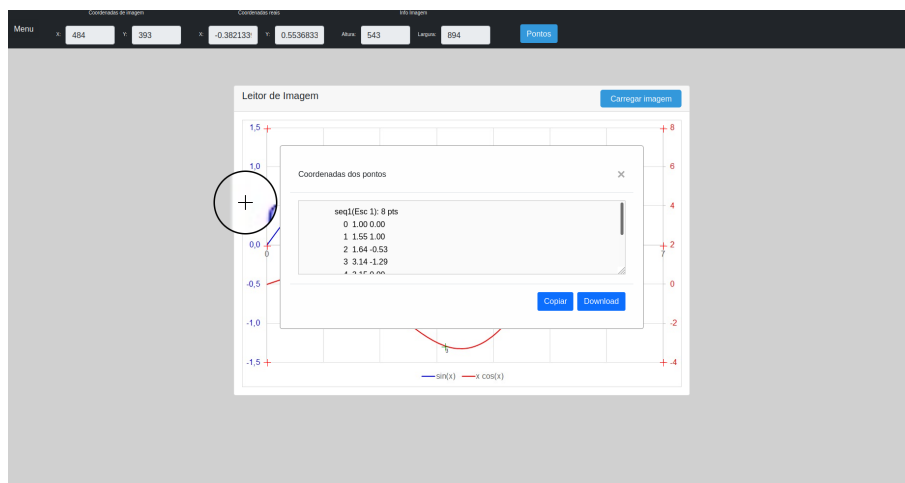


Figura 4.12 – Coordenadas dos pontos extraídos

Como é possível observar na Figura 4.12, o usuário pode clicar no botão Copiar caso queira colar o conteúdo da janela onde desejar, ou baixar o conteúdo como texto sem formatação e extensão TXT (texto sem formatação).

## 4.2 ANÁLISE DOS RESULTADOS

Para gerar resultados passíveis de análise foi elaborada a Figura 4.13-(a), medindo 640 *pixels* de largura e 570 *pixels* de altura, na qual foi plotada uma circunferência de raio unitário (200 *pixels*), centrada na origem e um quadrado de lados iguais a 2,2 (440 *pixels*), também centrado em (0, 0).

Próximo à circunferência, internamente, foram desenhadas 12 marcas verdes em forma de cruz, em intervalos regulares de  $30^\circ$ , sendo que a primeira marca forma um ângulo de  $15^\circ$  com o eixo das abscissas. Evitou-se marcar as quatro interseções da circunferência com os eixos coordenados porque os *pixels* nestes pontos são posicionados sem arredondamento, o que poderia superestimar a precisão dos resultados. A finalidade dessas marcas foi servir como referência visual para a seleção de 12 pontos bem distribuídos sobre a circunferência.

A imagem 4.13-(a) passou por duas transformações geométricas afins, sendo a primeira um cisalhamento horizontal para a direita e a segunda uma rotação de  $5^\circ$  no sentido anti-horário. O cisalhamento transformou a circunferência numa elipse e o quadrado num paralelogramo, como mostra a Figura 4.13-(b), na qual também são mostrados os cinco pontos de controle usados na definição da escala. A Figura 4.13-(c) mostra os pontos de dados, que são aqueles de onde as coordenadas foram extraídas e encontram-se nas duas primeiras colunas da Tabela 4.1.

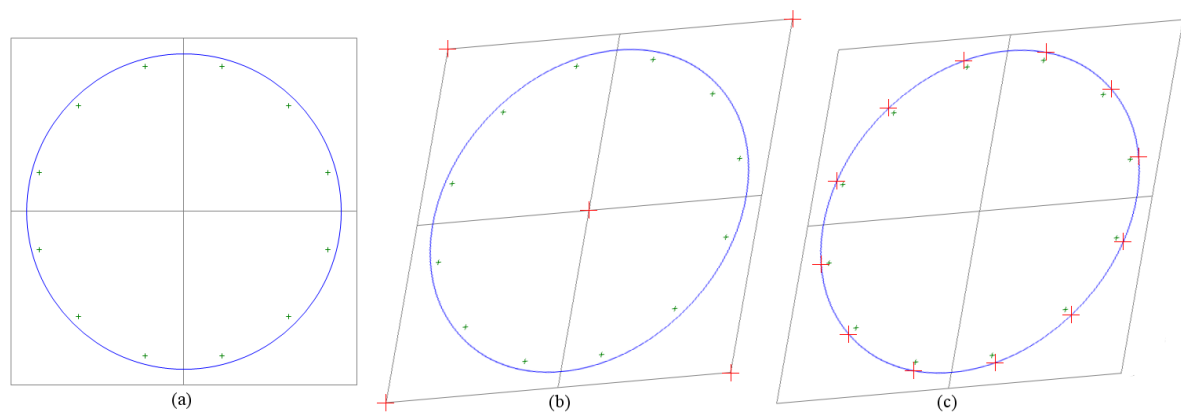


Figura 4.13 – (a) imagem de uma circunferência de raio unitário; (b) imagem transformada e pontos de controle; (c) pontos de extração de coordenadas

Pode-se observar na Tabela 4.1 que as coordenadas dos 12 pontos foram obtidas de modo satisfatoriamente preciso, pois o raio calculado mais distante da unidade foi 0,9972, o que corresponde a um erro relativo de 0,28% para menos. Considerando-se que o raio da circunferência mede 200 *pixels* e que a posição do clique possa diferir um *pixel* da posição ideal, o erro relativo máximo é da ordem de 0,5%, para este caso. Os valores dos coeficientes de determinação (0,999998 para escala das abscissas e 0,999996 para a escala das ordenadas) corroboram a qualidade do ajuste das equações, que refletem na precisão das coordenadas obtidas. Além de uma escala confiável, a precisão das coordenadas depende diretamente da posição do clique, mas nem sempre é fácil selecionar um determinado pixel entre seus vizinhos, uma vez que o efeito *antialiasing*, usado para evitar o serrilhamento (degraus) em imagens digitais, torna suas inten-

Tabela 4.1 – Resultados da extração das coordenadas de 12 pontos sobre uma circunferência de raio unitário

$x$	$y$	$r = \sqrt{x^2 + y^2}$	%
0,9664	0,2596	1,0007	0,07
0,7120	0,7016	0,9996	-0,04
0,2488	0,9705	1,0019	0,19
-0,2745	0,9609	0,9993	-0,07
-0,7084	0,7042	0,9989	-0,11
-0,9606	0,2708	0,9980	-0,20
-0,9677	-0,2468	0,9987	-0,13
-0,7107	-0,7036	1,0001	0,01
-0,2483	-0,9676	0,9990	-0,10
0,2708	-0,9625	0,9999	-0,01
0,7089	-0,7013	0,9972	-0,28
0,9611	-0,2679	0,9977	-0,23

sidades de cor próximas. Se as coordenadas desejadas estiverem sobre uma curva desenhada com espessura par de *pixels* esta escolha se torna ainda mais difícil.

## 5 CONCLUSÃO

Este trabalho iniciou-se com uma pesquisa dos softwares *digitizers* disponíveis na internet e, após uma análise de cada um deles e posterior comparação entre os mesmos, verificou-se que algumas funcionalidades ou melhorias importantes poderiam ser implementadas, a fim de facilitar e tornar mais preciso o processo de configuração de escalas e seleção de pontos.

Considerando a pesquisa feita no Capítulo 2, foram implementadas funcionalidades visando facilitar a configuração das escalas, como a possibilidade de configurar mais de uma escala em uma mesma sessão, definir os pontos de controle com o teclado do computador, para garantir maior precisão na marcação dos pontos e, portanto, uma configuração mais próxima do ideal. Também implementou-se a funcionalidade de *zoom* móvel, que é similar a uma lupa real e acompanha o cursor do mouse, permitindo ao usuário visualizar de forma mais clara a região do gráfico onde o cursor se encontra.

Dada a importância da configuração de escalas confiáveis, implementou-se uma funcionalidade que consiste em disponibilizar para cada equação ajustada uma estatística básica contendo os coeficientes de determinação e os desvios absolutos e relativos associados às coordenadas dos pontos de controle. Assim, além de mais transparência, é possível ter mais segurança quanto ao nível de confiabilidade dos dados, que continua sob responsabilidade do usuário.

Quanto aos pontos de dados, que são o que realmente interessam, as implementações citadas anteriormente neste capítulo também podem ser aplicadas para melhorar a precisão no momento de selecioná-los. Implementou-se ainda uma funcionalidade na qual o usuário pode definir mais de uma sequência de coordenadas para uma mesma escala, o que permite agrupar os pontos de dados em sequências diferentes conforme a necessidade.

A digitalização das coordenadas de uma circunferência de raio unitário, a partir de uma imagem propositadamente cisalhada e rotacionada, mostrou que o modelo matemático usado nas transformações geométricas afins é adequado para representar as equações de escala, na medida em que proporcionou coeficientes de determinação bastante próximos da unidade e erros relativos máximos inferiores ao esperado.

Por fim, considerando-se a metodologia adotada e os resultados obtidos, tanto no que concerne ao *digitizer* implementado como também aos dados que com ele se obtiveram, pode-se concluir que os objetivos inicialmente propostos foram plenamente atingidos, embora se vislumbre um significativo espaço para vários avanços, como mostrado no próximo item.

## 5.1 TRABALHOS FUTUROS

Com o decorrer do desenvolvimento do presente trabalho, além das ideias de melhorias vislumbradas inicialmente, novas ideias de funcionalidade foram surgindo, sendo que algumas delas estão presentes nos *digitizers* mais sofisticados. Surpreendentemente, a implementação de algumas delas podem ser feitas com relativamente pouca alteração no núcleo matemático, mas trazem algum impacto na remodelagem da interface com o usuário. A seguir, são listadas algumas sugestões de funcionalidades:

- Implementar o ajuste de escala em imagens fotográficas, que contêm distorção de perspectiva, em maior ou menor grau.
- Implementar o ajuste de escala em imagens alteradas algoritmicamente ao arrastar-se, em qualquer direção, um ou mais vértices de uma imagem retangular.
- Implementar a visualização de linhas de grade personalizáveis para facilitar a seleção de pontos em intervalos regulares.
- Implementar a possibilidade de ajustar polinômios às coordenadas extraídas, uma vez que o arcabouço matemático usado resolver as equações de escala podem fazê-lo, com poucas alterações.
- Implementar a capacidade de armazenar todo o contexto de uma sessão do programa, para recuperá-lo posteriormente e o trabalho retomado.
- Implementar a transformada de *Hough* para reconhecimento automático e parametrização dos eixos e das linhas de grade, quando presentes.
- Implementar um modo de seleção de pontos onde o software identifica a curva do gráfico e sobre ela seleciona a quantidade de pontos equidistantes solicitada pelo usuário. Uma solução geral neste sentido encerra um nível de complexidade bem acima das sugestões anteriores.



## REFERÊNCIAS

- BUILTWITH. **jQuery Usage Statistics**. 2021. Acessado 09/04/2021. Disponível em: <<https://trends.builtwith.com/javascript/jquery>>.
- FAN, Hongfei et al. Covscode: A novel real-time collaborative programming environment for lightweight ide. *Multidisciplinary Digital Publishing Institute*, p. 1–2, 2019.
- FRIENDLY, Michael et al. A brief history of data visualization. In: \_\_\_\_\_. [S.l.: s.n.], 2008. p. 15–56.
- HABGOOD, Ken; AREL, Itamar. A condensation-based application of cramer’s rule for solving large-scale linear systems. *Journal of Discrete Algorithms*, p. 98–109, 2012.
- HETEM JUNIOR, Annibal. **Computação gráfica**. Rio de Janeiro, RJ: LTC, 2006.
- KAYNAK, Okyay; YIN, Shen. Big data for modern industry: Challenges and trends [point of view]. **Proceedings of the IEEE**, v. 103, p. 143–146, 02 2015.
- KEAHEY, T. Alan. Using visualization to understand big data. p. 2–3, 2013.
- LYTRAS, Miltiades D.; VISVIZI, Anna. Big data and their social impact: Preliminary study. **Multidisciplinary Digital Publishing Institute**, p. 3–6, 09 2019.
- NDIA, John; MUKETHA, Geoffrey; OMIENO, Kelvin. A survey of cascading style sheets complexity metrics. **SSRN Electronic Journal**, p. 22–24, 01 2019.
- OGREAN, Claudia. Relevance of big data for business and management. exploratory insights (part i). **Studies in Business and Economics**, v. 13, p. 153–163, 08 2018.
- OHURU, Michael. Javascript how we got here. an indepth history. p. 1–18, 09 2020.
- ROGERS, Alan; DAWSON, John A. Which digitizer? **Area**, [Royal Geographical Society (with the Institute of British Geographers), Wiley], v. 11, n. 1, p. 69–73, 1979. ISSN 00040894, 14754762. Disponível em: <<http://www.jstor.org/stable/20001441>>.
- SHAH, Mahamad Qaium. Responsive web development using the twitter bootstrap framework. p. 31–32, 2015.
- VINCE, John. **Matrix Transforms for Computer Games and Animation**. Bournemouth, UK: Springer, 2012.
- W3COUNTER. **Browser & Platform Market Share**. 2021. Acessado 29/04/2021. Disponível em: <<https://www.w3counter.com/globalstats.php?year=2021&month=3>>.