

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
DEPARTAMENTO ACADÊMICO DE ELETRÔNICA  
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA  
ENGENHARIA DE COMPUTAÇÃO

RAFAEL JORGE TRINDADE

**APRIMORAMENTO DA ROBUSTEZ DE SISTEMA DE MEDIÇÃO  
DE NÍVEL DE CORPOS D'ÁGUA COM BASE EM IMAGENS  
DIGITAIS**

TRABALHO DE CONCLUSÃO DE CURSO

BRASIL  
2021

RAFAEL JORGE TRINDADE

**APRIMORAMENTO DA ROBUSTEZ DE SISTEMA DE MEDIÇÃO  
DE NÍVEL DE CORPOS D'ÁGUA COM BASE EM IMAGENS  
DIGITAIS**

Trabalho de Conclusão de Curso apresentado à Engenharia de Computação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Engenheiro de Computação.

Orientador: Prof. Dr. Bogdan Tomoyuki Nassu  
Universidade Tecnológica Federal do Paraná

BRASIL  
2021

**RAFAEL JORGE TRINDADE**

**APRIMORAMENTO DA ROBUSTEZ DE SISTEMA DE MEDIÇÃO DE NÍVEL DE  
CORPOS D'ÁGUA COM BASE EM IMAGENS DIGITAIS**

Trabalho de Conclusão de Curso de Graduação  
apresentado como requisito para obtenção do título de  
Bacharel em Engenharia de Computação da Universidade  
Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: 15 de julho de 2021

---

**BOGDAN TOMOYUKI NASSU**

Doutorado em Advanced Interdisciplinary Studies pelo Research Center for Advanced Science and  
Technology (University of Tokyo)  
Universidade Tecnológica Federal do Paraná (UTFPR)

---

**KEIKO VERONICA ONO FONSECA**

Doutorado em Engenharia Elétrica pela Universidade Federal de Santa Catarina  
Universidade Tecnológica Federal do Paraná (UTFPR)

---

**LEYZA BALDO DORINI**

Doutorado em Ciência da Computação pela Universidade Estadual de Campinas  
Universidade Tecnológica Federal do Paraná (UTFPR)

**CURITIBA**

**2021**

## AGRADECIMENTOS

Gostaria de declarar minha sincera gratidão às seguintes pessoas:

- ao professor orientador Dr. Bogdan Tomoyuki Nassu pela paciência, franqueza, pelo conhecimento compartilhado e por estar sempre presente ao longo do desenvolvimento deste trabalho;
- aos professores que, com sabedoria e sensibilidade, tornaram minha jornada suportável nos momentos que mais precisei;
- aos colegas de curso que cumpriram com vosso compromisso diante das dificuldades;
- à minha esposa, Luana Laitner, por ter me apresentado este caminho e por me fazer acreditar ser capaz de segui-lo. Sem você, esse dia não teria chegado;
- à minha mãe, Denise Guollo Jorge Trindade, pela inesgotável paciência e suporte oferecidos.

Vossas ações não passaram despercebidas e serei para sempre grato por vossa dedicação.

## RESUMO

TRINDADE, Rafael J.. Aprimoramento da robustez de sistema de medição de nível de corpos d'água com base em imagens digitais. 2021. 47 f. Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Brasil, 2021.

É seguro dizer que a maioria das cidades que são construídas próximas a rios, lagos ou bacias sofrem com enchentes. Nem todas elas são capazes de investir em soluções baseadas em sensoriamento para monitorar o comportamento destes corpos d'água de modo a prevenir ou reduzir perdas materiais e de vidas. Existem diversos sistemas propostos para realização desta tarefa, mas os poucos que se baseiam em imagens oferecem pouca flexibilidade e confiabilidade. Este trabalho é o aprimoramento do Trabalho de Conclusão de Curso de Thiago Nagaoka (2019) que resolve o problema de orientação utilizando o *pipeline* de algoritmos SIFT e RANSAC. O sistema é composto por uma régua especificamente projetada, uma câmera e um computador. Sobre o trabalho original, no qual foi proposto o modelo de régua aqui utilizado e uma forma de extraí-la da imagem capturada pela câmera, foi desenvolvido um novo processo que faz uso da frequência do padrão da régua para inferência da medida do nível d'água. Este foi também submetido a um conjunto mais adverso de condições durante os testes. Nos casos em que houve êxito na conclusão do processamento, foram atingidos erros na faixa entre 0 a 10% de erro. São também propostas correções para obter maior êxito e precisão ao realizar medições.

**Palavras-chave:** Medição do nível d'água. Visão computacional. Imagens.

## ABSTRACT

TRINDADE, Rafael J.. Improvements in robustness of an image-based water level measurement system. 2021. 47 f. Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Brasil, 2021.

It is safe to say that the majority of cities built near rivers, lakes or basins experience floodings. Not every one of them is capable of investing in sensor-based solutions to monitor the behavior of these bodies of water in order to prevent or reduce the loss of material goods or of life. There exist quite a few systems to handle the task, but the few image-based solutions lack flexibility and reliability. This work is an improvement of the Thiago Nagaoka undergraduate thesis' system that solves the orientation issue by using the SIFT and RANSAC pipeline. The system is composed of an specifically designed ruler, a camera and a computer. Over the original work, in which the ruler and it's extraction from the image captured by the camera here used, a new process that bases it's measurements over the ruler pattern's frequency was developed to infer the water level measure. During testing, it was also subjected to a more adverse set of conditions. The proposed system computes measurements with an error concentrated in the range of 0 to 10%, when the process runs successfully. Adjustments to increase the success rate and precision are also proposed.

**Keywords:** Water level measurement. Computer vision. Images.

## LISTA DE FIGURAS

Figura 1 – Protótipo da régua inserido em um recipiente com água. . . . .	2
Figura 2 – Remoção do fundo da imagem. . . . .	5
Figura 3 – Detecção da borda d'água. . . . .	5
Figura 4 – Caracteres da régua reconhecidos utilizando a <i>CNN</i> . . . . .	6
Figura 5 – Régua empregada na medição no nível d'água. . . . .	6
Figura 6 – Processo para extração da região de interesse. . . . .	7
Figura 7 – Busca pela coluna central da régua. . . . .	8
Figura 8 – Resultado. . . . .	9
Figura 9 – Esquemático da régua utilizada. . . . .	10
Figura 10 – Transformação da perspectiva do cenário. . . . .	11
Figura 11 – Resultado do <i>warping</i> . . . . .	12
Figura 12 – Sinal capturado do centro da figura. . . . .	13
Figura 13 – Detecção de possíveis <i>interfaces</i> com a água. . . . .	13
Figura 14 – Representação gráfica do algoritmo de eliminação de <i>outliers</i> . . . . .	14
Figura 15 – Construção do histograma de orientações. . . . .	17
Figura 16 – Construção de descritores a partir de gradientes. . . . .	17
Figura 17 – Imagens e seus respectivos pontos de interesse representados graficamente. . . . .	18
Figura 18 – Casamento de descritores entre duas imagens. . . . .	18
Figura 19 – Transformação da imagem utilizando a matriz de homografia obtida pelo algoritmo RANSAC. . . . .	19
Figura 20 – Modelo de construção da régua. . . . .	22
Figura 21 – Extração da <i>ROI</i> para imagens de entrada 20a (referência) e 20b (cena). . . . .	23
Figura 22 – Exemplo de extração de sinal a partir da imagem. . . . .	24
Figura 23 – Componentes de frequências que compõem o sinal 22b. . . . .	25
Figura 24 – Resultado do processo de otimização. . . . .	27
Figura 25 – Amplitude da componente de frequência $\Omega$ ao longo do sinal 22b. . . . .	28
Figura 26 – Local estimado da <i>interface</i> com a água. . . . .	29
Figura 27 – Protótipo da régua em escala reduzida. . . . .	31
Figura 28 – Amostras feitas de acordo com o que foi proposto na Subseção 5.3. . . . .	33
Figura 29 – Figura com representação visual da frequência calculada (senoide em vermelho) e região onde estima-se que esteja a interface com a água (linhas verticais nas cores ciano e vermelho). . . . .	35
Figura 30 – Figura com representação visual da frequência calculada (senoide em vermelho) e região onde estima-se que esteja a interface com a água (linhas verticais nas cores ciano e vermelho). . . . .	36
Figura 31 – Caso de falha ao detectar divisória com a água. . . . .	37

Figura 32 – Caso de falha da detecção da divisória com a água. . . . .	38
Figura 33 – Região da Figura 18 que evidencia mal rendimento, i.e. poucas características inambíguas . . . . .	39
Figura 34 – Histogramas de resultados. . . . .	40
Figura 35 – Comparação entre borda esquerda da região de interesse a medida que é processada. . . . .	41
Figura 36 – Amostras e respectivos resultados 1. . . . .	42
Figura 37 – Amostras e respectivos resultados 2. . . . .	43
Figura 38 – Amostras e respectivos resultados 3. . . . .	44



## LISTA DE TABELAS

Tabela 1 – Resultados de experimentos realizados. . . . .	34
---	----

## LISTA DE ALGORITMOS

Algoritmo 1 – Função de fitness utilizada para buscar amplitude máxima. . . . .	26
Algoritmo 2 – Busca pelo maior e mais rápido declínio. . . . .	29

## SUMÁRIO

<b>1 – INTRODUÇÃO</b> . . . . .	<b>1</b>
1.1 ORGANIZAÇÃO DO TRABALHO . . . . .	3
<b>2 – TRABALHOS RELACIONADOS</b> . . . . .	<b>4</b>
2.1 IMPLEMENTATION OF INVERSE PERSPECTIVE MAPPING FOR CAMERA- VISION WATER-LEVEL MEASUREMENTS . . . . .	4
2.2 MEDIÇÃO AUTOMÁTICA DE NÍVEL DE ÁGUA BASEADO EM IMAGENS: CASO DO RIO ITAJAÍ-AÇÚ . . . . .	4
2.3 MEDIÇÃO DE NÍVEL DE RIOS BASEADA EM ANÁLISE DE VÍDEO . . . . .	9
<b>3 – DEFINIÇÕES GERAIS</b> . . . . .	<b>16</b>
3.1 IMAGEM DIGITAL . . . . .	16
3.2 SCALE-INVARIANT FEATURE TRANSFORM (SIFT) . . . . .	16
3.3 CASAMENTO DE PONTOS DE INTERESSE . . . . .	17
3.4 RANDOM SAMPLE CONSENSUS (RANSAC) . . . . .	19
3.5 TRANSFORMAÇÃO EM SÉRIES DE FOURIER . . . . .	19
3.6 SHORT-TIME FOURIER TRANSFORM (STFT) . . . . .	20
<b>4 – MEDIÇÃO DO NÍVEL D'ÁGUA</b> . . . . .	<b>21</b>
4.1 COMPONENTES DA RÉGUA . . . . .	21
4.2 EXTRAÇÃO DA REGIÃO DE INTERESSE ( <i>ROI</i> ) . . . . .	22
4.3 <i>WARPING</i> AUXILIADO POR ANÁLISE DE FOURIER . . . . .	23
4.4 DETECÇÃO DA INTERFACE COM A ÁGUA . . . . .	26
4.5 CÁLCULO DO COMPRIMENTO DA RÉGUA . . . . .	28
<b>5 – EXPERIMENTOS E RESULTADOS</b> . . . . .	<b>31</b>
5.1 DESCRIÇÃO DA RÉGUA . . . . .	31
5.2 DESCRIÇÃO DO PROTÓTIPO DO SISTEMA PROPOSTO . . . . .	31
5.3 DESCRIÇÃO DO EXPERIMENTO . . . . .	32
5.4 RESULTADOS OBTIDOS . . . . .	32
5.5 DISCUSSÃO . . . . .	37
5.6 AMOSTRAS E RESULTADOS . . . . .	40
<b>6 – CONCLUSÃO</b> . . . . .	<b>45</b>
6.1 TRABALHOS FUTUROS . . . . .	45
<b>Referências</b> . . . . .	<b>46</b>

## 1 INTRODUÇÃO

O monitoramento de nível de corpos d'água é utilizado no planejamento de cidades e previsões econômicas para setores que da água dependem. A seca registrada em 2020 no estado do Paraná fez com que as autoridades adotassem políticas mais restritas e medidas de racionamento, com base no volume d'água disponível<sup>1</sup>. Há risco também de racionamento energético nas regiões Sudeste e Centro-Oeste devido à falta de chuvas. Enquanto isso, no estado do Amazonas registra-se cheia recorde dos rios e enchentes<sup>2</sup>.

Em quase todo o território, existe a possibilidade de acumulados de precipitação elevada em algum momento. O nível das águas pode também ser rapidamente afetado por marés ou por ação humana, nos raros casos de rompimento de barragens. Para minimizar perda de bens materiais e de vida humana, foram concebidos sistemas de alerta que funcionam com base no monitoramento do nível de rios, lagos e bacias. Estes podem disparar alertas ou soar alarmes caso seja concluído que há algum risco.

A tarefa básica que deve ser executada por estes sistemas é a medição do nível d'água. Diversos sistemas para realização disto já foram desenvolvidos, baseados em transdutores de pressão (Ribeiro E. P.; Braga S. M.; Fernantes C. V. S, 2015), borbulhadores (Szymanski C.; Sousa N.; Kesting F. L., 2016), sensores ultrassônicos (Andrade F., ) e até mesmo LIDARs (NINO et al., 2008). Enquanto sensores podem fornecer variados níveis de precisão, sua utilização implica em manutenção dessas unidades e das instalações para comunicação das leituras, o que pode resultar em custo e complexidade elevados. Em muitos casos, a forma mais eficiente, econômica e simples é a leitura periódica manual de réguas limnimétricas por alguém.

Existem outras abordagens que sugerem o uso de imagens para leitura de alguns tipos diferentes de réguas, como feito por (Xu et al., 2018) e (Hübner, 2013). Nestas foram utilizadas câmeras para capturar imagens destas réguas e, após ajustes, realizar a inferência da medida da água. Câmeras já instaladas poderiam ser utilizadas para realização destas medidas, reduzindo custos e facilitando a manutenção, ainda que forneçam resultados menos precisos que sensores e sistemas especializados em leitura do nível d'água. No entanto, essas abordagens necessitam que a régua esteja ortogonalmente alinhada com a imagens.

Em (Nagaoka, 2019) foi proposto uma solução que poderia resolver este tipo de problema. A solução é composta de um novo formato de régua, composto de uma imagem referência e uma seção alongada com faixas alternadamente nas cores branca e preta, uma boia presa ao eixo desta seção, um novo processo e algoritmos para medição. Porém, os algoritmos desenvolvidos se mostraram muito sensíveis a ruídos para inferência das medidas.

Neste trabalho é proposta uma alternativa construída sobre o trabalho de (Nagaoka,

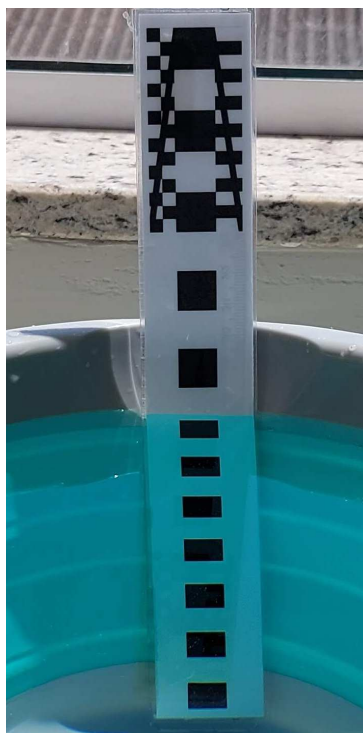
---

<sup>1</sup><http://www.terra.ufpr.br/portal/laboclima/2020/05/21/curitiba-enfrenta-sua-pior-seca-em-40-anos/>

<sup>2</sup><https://www.cnnbrasil.com.br/nacional/2021/06/05/rio-negro-registra-a-maior-cheia-em-119-anos-em-manaus>

2019) que tenta tornar as medições mais robustas, utilizando componentes de frequência presentes ao longo da régua. O sistema é composto por uma câmera (idealmente, com resolução maior que 680x480 pixels), de um computador e da régua, especialmente projetada para esse propósito, sem que haja necessidade de uma boia. A Figura 1 apresenta um protótipo da régua deste sistema inserida em um recipiente com água.

Figura 1 – Protótipo da régua inserido em um recipiente com água.



Fonte: Autoria própria

O processo começa com a captura de uma imagem que contenha a régua, podendo ser uma simples fotografia ou extraída de vídeo, como (Nagaoka, 2019) propõe. Para localizar a régua na imagem é utilizado o algoritmo SIFT (*Scale-Invariant Feature Transform*) para obter descritores da imagem referência, informada como parâmetro para o algoritmo. Repete-se o procedimento para a imagem da régua obtida com a câmera, realizam-se então casamentos entre os descritores e refina-se o resultado com o algoritmo RANSAC (*Random Sample Consensus*). O resultado deste algoritmo é uma matriz de homografia, utilizada para transformar a imagem capturada para a perspectiva da imagem referência informada. Este resultado é geralmente imperfeito, pois os algoritmos ou a implementação destes são não-determinísticos e as imagens capturadas *sempre* possuem ruído.

A abordagem de (Nagaoka, 2019) para solucionar este problema é tratar esta imperfeição que afeta o comprimento das faixas como um erro linear. Sobre esta premissa, foram propostas duas soluções, uma que busca por uma descontinuidade nesta linearidade para determinar a região de encontro com a água e outra que utiliza regressão linear para remoção destes *outliers*. As soluções propostas provaram ser muito sensíveis a perturbações, como

iluminação irregular e reflexões.

Neste trabalho é proposta uma solução para corrigir esta transformação imperfeita. Extrai-se o sinal de onda quadrada presente ao longo do eixo da régua e com um algoritmo de otimização procura-se a rotação desta imagem que gera a onda quadrada com picos mais elevados no espectro de frequência, sob a premissa de que isso é equivalente a encontrar o sinal periódico gerado por uma régua perfeitamente alinhada. Com a frequência encontrada, pode-se então avaliar o comportamento do sinal ao longo da régua em busca do começo e fim deste. Por fim, com frequência e comprimento do sinal computados associados ao comprimento métrico de um período desta onda obtêm-se o comprimento.

Os resultados obtidos ficaram concentrados na faixa de 0 a 10% de erro, equivalente a 0 e 20 milímetros, para uma régua de 288mm e faixas de 13mm. Observou-se consistência satisfatória sob um número de cenários adversos, atingindo assim o objetivo esperado.

## 1.1 ORGANIZAÇÃO DO TRABALHO

No Capítulo 2 são descritos os trabalhos melhor relacionados à abordagem adotada aqui, assim como pontos negativos destes. No Capítulo 3 estão dispostos os algoritmos utilizados, assim como algumas definições que são utilizadas ao longo do texto. No Capítulo 4 quatro está descrito como os algoritmos e definições do capítulo anterior são utilizados extração da régua da imagem, otimização da perspectiva e cálculo para encontrar as extremidades da régua. No Capítulo 5 são detalhadas as abordagens para captura de dados, de modo a produzi-los de forma descomplicada, e é explicado por que a abordagem para tal é eficaz. Por fim, no Capítulo 6 encontram-se as conclusões deste trabalho e sugestões para trabalhos futuros.

## 2 TRABALHOS RELACIONADOS

Esta seção apresentará alguns trabalhos que realizam a medição do nível d'água utilizando visão computacional. A Subseção 2.3 faz referência ao Trabalho de Conclusão de Curso de (Nagaoka, 2019), cujos desdobramentos motivaram a criação desta obra.

### 2.1 IMPLEMENTATION OF INVERSE PERSPECTIVE MAPPING FOR CAMERA-VISION WATER-LEVEL MEASUREMENTS

O trabalho de (Xu et al., 2018) propõe uma abordagem para medição do nível d'água utilizando operadores morfológicos e redes neurais convolucionais.

O primeiro passo é a segmentação com limiares para separar a régua do fundo da imagem. A Figura 2a mostra a imagem RGB convertida para escala de cinza. Faz-se conversão de RGB para a representação de cores HSV (Fig. 2b) e aplica-se método de Otsu (Otsu, 1979) sobre a componente H, que codifica o matiz de cada pixel. Isto é feito porque a régua possui marcações bem específicas com a cor vermelha. O resultado é a conversão de uma imagem em escala de cinza para uma imagem binária (Fig. 2c).

A seguir, é realizada a extração da porção da régua que fica acima d'água com base na distribuição espacial das cores, no espaço RGB, que este pode ter. Segundo o autor, as cores vermelha e azul são predominantes. As pequenas regiões resultantes do processo (Fig. 3b) são removidas, resultando na região de interesse (Fig. 3c).

Os caracteres da régua são identificados utilizando uma rede neural convolucional (Fig. 4), mas apenas a região com os caracteres "E" é utilizada. É tomado o comprimento em *pixels* de cada uma das ocorrências desta letra para obter o tamanho médio. Este valor é então utilizado para estimar o tamanho do caractere incompleto, considerando o quanto dele está para fora da água. Com esses valores, calcula-se o comprimento da régua, em pixels. Sabendo de antemão o tamanho do caractere "E" em milímetros, conclui-se a inferência da medida.

Os autores estabeleceram uma metodologia, mas não publicaram as medições realizadas. Nada foi dito quando à orientação e perspectiva da câmera com relação à régua. Supõe-se então que esses precisam ser mantidos alinhados para realizar a medição. Não há também dados que mostrem a precisão da inferência sob condições mais diversas de iluminação.

### 2.2 MEDIÇÃO AUTOMÁTICA DE NÍVEL DE ÁGUA BASEADO EM IMAGENS: CASO DO RIO ITAJAÍ-AÇÚ

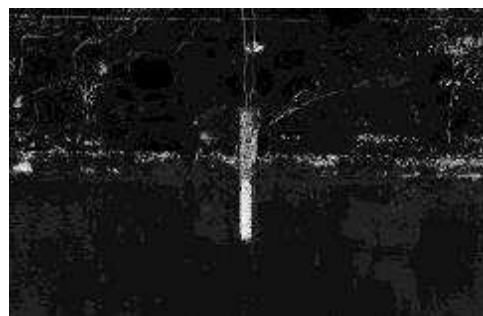
O trabalho de (Hübner, 2013) propõe uma alternativa para medição do nível do rio Itajaí-Açu utilizando visão computacional, um problema então solucionado com um conjunto

Figura 2 – Remoção do fundo da imagem.

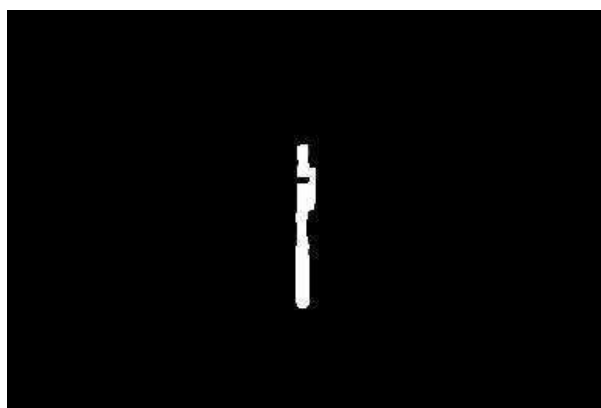
(a) Imagem convertida para escalas de cinza a partir do espaço de cores RGB.



(b) Canal H da imagem 2a.



(c) Imagem binária obtida a partir 2b com o método de Otsu.



Fonte: (Xu et al., 2018)

Figura 3 – Detecção da borda d'água.

(a) Região com cores da imagem original, extraída com 2.



(b) Canal H da imagem original.

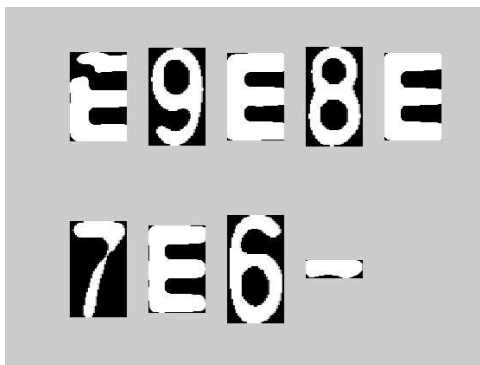


(c) Imagem binária obtida a partir 2b com o método Otsu.



Fonte: (Xu et al., 2018)



Figura 4 – Caracteres da régua reconhecidos utilizando a *CNN*.

Fonte: (Xu et al., 2018)

de sensores. Estes foram dispostos ao longo de uma régua feita de aço inox, fixada a um dos pilares de uma ponte que passa sobre o rio (Fig. 5).

Figura 5 – Régua empregada na medição no nível d'água.

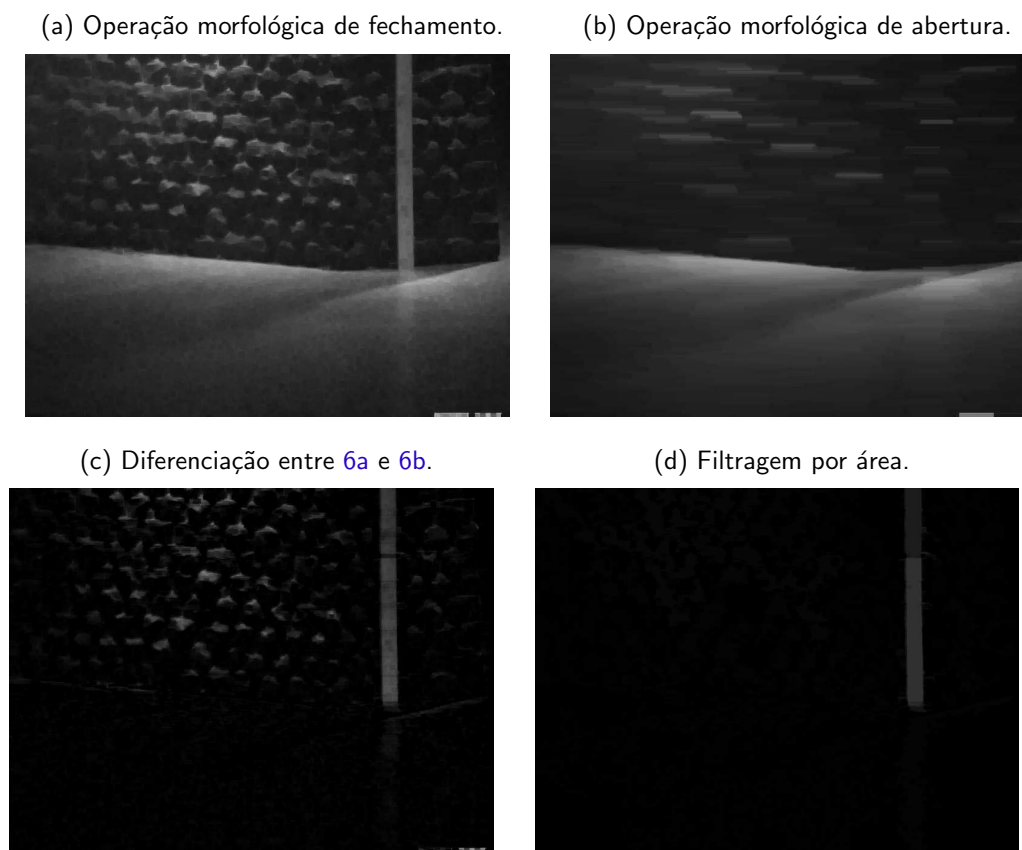


Fonte: (Hübner, 2013)

O primeiro passo do processamento é a remoção do *background* da imagem. Isto é feito com um operador morfológico de fechamento, de modo a fundir pequenas marcações escuras e também os números da régua, o que também homogeniza a imagem (Fig. 6a). Na sequência, um operador morfológico de abertura é utilizado e o resultado é a remoção da imagem da régua (Fig. 6b). O resultado dos últimos dois procedimentos é então diferenciado, resultando em realce na região onde a régua está presente (Fig. 6c). Finalmente, elementos visuais com uma área inferior a um determinado limite são filtrados, resultando somente na região de interesse (Fig. 6d).

O próximo passo é encontrar a coluna da imagem que melhor representa o centro da régua. A intuição do procedimento é que as colunas da imagem que pertencem à régua terão *pixels* de intensidade luminosa maior. Toma-se a imagem resultante do processo anterior (Fig. 7a). Os valores das intensidades de cada coluna desta imagem são então somados, resultando em 7b. Ruídos são então filtrados deste gráfico. Por fim, faz-se a média no eixo *x* das posições onde ocorrem a subida e descida da curva. O resultado pode ser observado em 7c.

Figura 6 – Processo para extração da região de interesse.



Fonte: (Hübner, 2013)

A determinação da linha d'água é feita verificando entre quais pontos do eixo  $y$ , na posição  $x_{col}$  encontrada no passo anterior, que ocorre a maior queda na intensidade luminosa. (Hübner, 2013) também empregou um filtro passa-baixa sobre os valores medidos temporalmente para desconsiderar *outliers* que viessem a aparecer. O resultado é denominado  $pos$ .

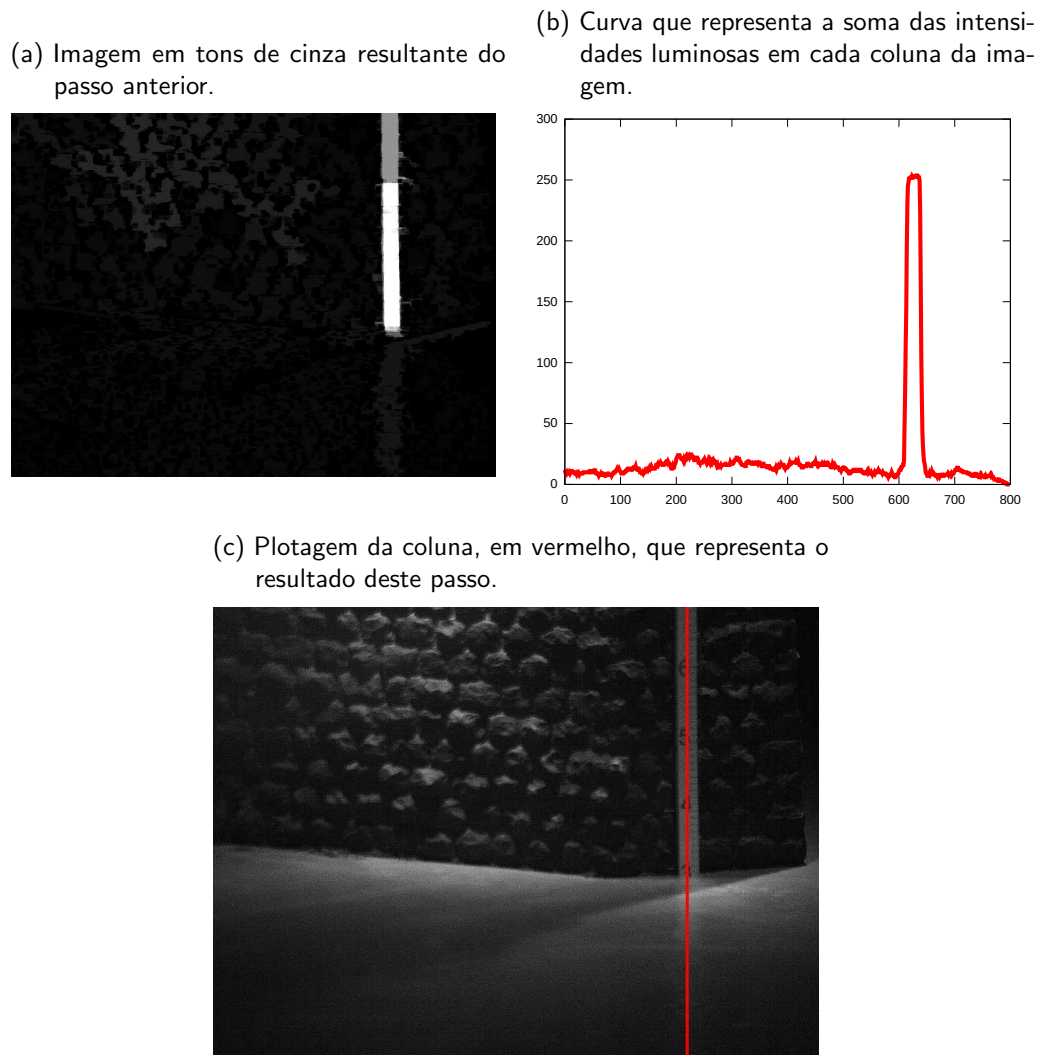
Por fim, para inferência da medida é preciso realizar a conversão de *pixels* para metros. Visto que a régua possui as marcações em metros, basta verificar qual são estes valores nas duas extremidades do eixo  $y$  para obtenção do coeficiente de crescimento. (Hübner, 2013) chegou a seguinte equação de reta com os valores encontrados:

$$n = (6,80 - 0,18) \frac{H - pos}{H} + 0,18,$$

tal que:

- $n$  = altura da água em metros
- 6,80 = altura da água na linha  $y = H$
- 0,18 = altura da água na linha  $y = 0$
- $H$  = comprimento máximo, em *pixels*, da imagem

Figura 7 – Busca pela coluna central da régua.



Fonte: (Hübner, 2013)

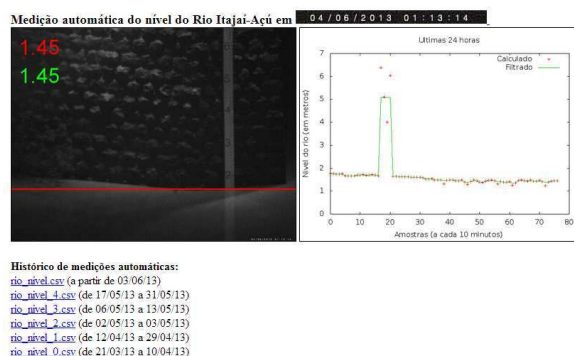
O resultado do procedimento pode ser visto na Figura 8a. 8b mostra um exemplo em que a filtragem dos valores auxilia na remoção de eventuais *outliers*.

Sobre uma base de 3096 imagens, foi obtido um resultado com erro médio de precisão menor que 5 centímetros. Para 63 destas imagens, foram feitas medições com erros maiores que 15 centímetros. Concluiu-se que um dos problemas que mais influenciaram nos resultados foi a falha ao detectar corretamente a linha de divisão com a água, o que impossibilitou que algumas medições fossem feitas.

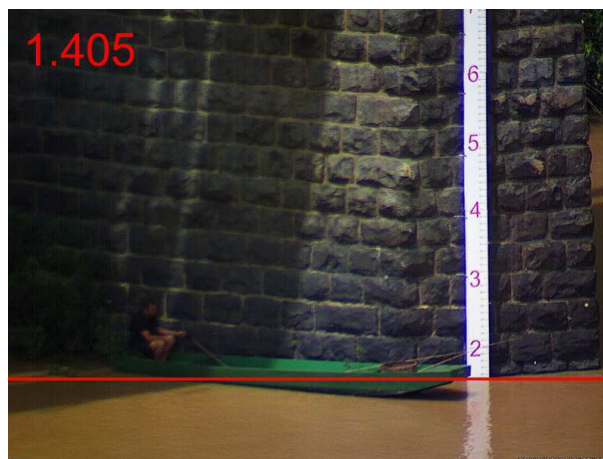
No entanto, esta abordagem exige o posicionamento adequado da câmera. Ao que tudo indica, o desvio de poucos graus no ângulo da imagem da câmera relativo à régua pode inviabilizar a medição. Variações na perspectiva da câmera com relação à régua também não foram abordadas.

Figura 8 – Resultado.

(a) Resultado do procedimento, onde também é exibido o processo de filtragem dos valores.



(b) Exemplo de *outlier*.



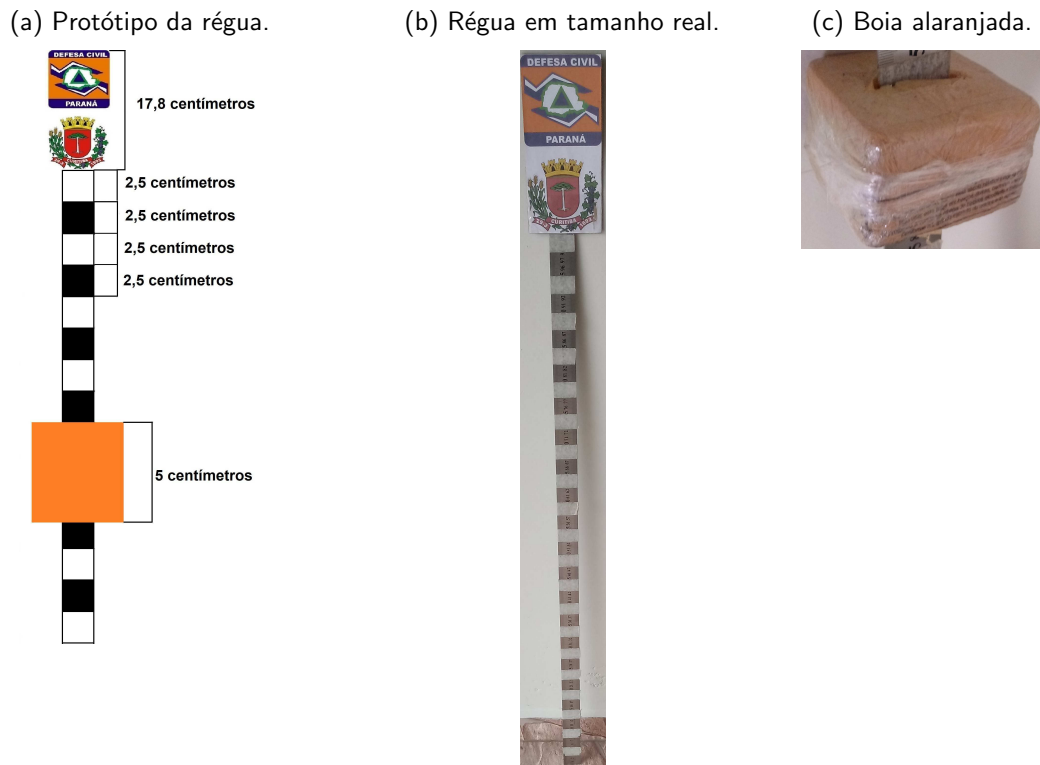
Fonte: (Hübner, 2013)

### 2.3 MEDIÇÃO DE NÍVEL DE RIOS BASEADA EM ANÁLISE DE VÍDEO

O trabalho de (Nagaoka, 2019) propõe uma solução diferente das demais aqui apresentadas no que concerne à localização e perspectiva da régua em relação à imagem da câmera. Os trabalhos expostos nas Seções 2.1 e 2.2 foram desenvolvidos sobre a premissa de que a câmera deveria estar precisamente ajustada com relação à régua. (Nagaoka, 2019) resolve este problema utilizando o método SIFT (Lowe, 2004) para buscar por determinadas características em uma imagem alvo e RANSAC (Fischler; Bolles, 1981) para mudar a perspectiva desta para um formato adequado, conseguindo assim isolar a régua do restante do cenário. O propósito deste trabalho é aprimorar os resultados obtidos através desta estratégia, portanto trabalho de (Nagaoka, 2019) será explicado com mais detalhes.

No trabalho dele, foi proposta uma codificação diferente para a régua, com a finalidade de destacar características importantes desta sob condições mais adversas de luminosidade. A Figura 9a resume a construção da régua. Empregou-se o uso de uma boia (Fig. 9c), para auxiliar na detecção da *interface* com a água.

Figura 9 – Esquemático da régua utilizada.



Fonte: (Nagaoka, 2019)

É importante ressaltar que o termo “vídeo” pode ser desconsiderado do restante processamento, visto que a partir dos *frames* de um dado vídeo é calculada a “imagem média”. Isto é feito realizando-se a média aritmética da soma dos *frames*:

$$A_{ij} = \frac{1}{N} \sum_{n=k}^{k+N} F_{ij}(n), \quad 0 < i < H, \quad 0 < j < L, \quad k + N \leq \text{quantidade total de frames}$$

tal que:

$$\begin{aligned} A_{ij} &= \text{pixel da imagem final A} \\ k &= \text{posição inicial da amostra do vídeo} \\ N &= \text{quantidade de frames} \\ F_{ij}(n) &= \text{valor do pixel } i,j \text{ do frame } F(n) \\ H &= \text{altura do frame} \\ L &= \text{largura do frame} \end{aligned}$$

Enquanto isso é útil para, por exemplo, eliminar gotas de chuva da imagem a ser processada, este não é o foco principal do trabalho.

Como mencionado anteriormente, a medição começa detectando-se a régua no cenário em que esta foi inserida. Utiliza-se o método SIFT sobre a imagem de referência (Fig 10a) e



sobre uma imagem do cenário (10b), obtém-se o que é representado pelas Figuras 10c e 10d, respectivamente.

Figura 10 – Transformação da perspectiva do cenário.



Fonte: (Nagaoka, 2019)

Os descritores gerados computando-se SIFT sobre essas imagens precisam então ser mapeados. Isto é feito utilizando um combinador de força bruta, o qual produz uma lista dos melhores casamentos entre os descritores das duas imagens. As Figuras 10c e 10d mostram em verde o resultado deste combinador.

Para realizar o *warping* a imagem da cena para a perspectiva, escala e orientação

Figura 11 – Resultado do *warping*.

Fonte: (Nagaoka, 2019)

da imagem de referência utiliza-se o algoritmo RANSAC para encontrar qual a homografia que realiza isto. Para as imagens da Figura 10 o resultado é mostrado na Figura 11. Nota-se que o resultado desta transformação não é perfeito. A extremidade da régua mais distante da imagem referência tende a ficar distorcida. Para posterior realização do cálculo do nível da água partiu-se da premissa que a distorção afeta o tamanho das faixas de forma linear, e.g. o comprimento em pixels das faixas brancas segue a equação  $y = 5x + 50$ , onde  $x$  é o índice da faixa branca e  $y$  o comprimento desta.

A medição do comprimento é feita sobre o sinal de onda quadrada presente ao longo do eixo da régua, obtido da imagem resultante do passo anterior (Fig. 12a). Isto é possível graças à codificação em faixas brancas e pretas de mesmo comprimento. O resultado da Figura 12b mostra este sinal e a Figura 12c mostra resultado da aplicação de um filtro passa-baixa sobre o sinal original. Este último passo demonstra ser necessário para adequada diferenciação e, por fim, localização das faixas.

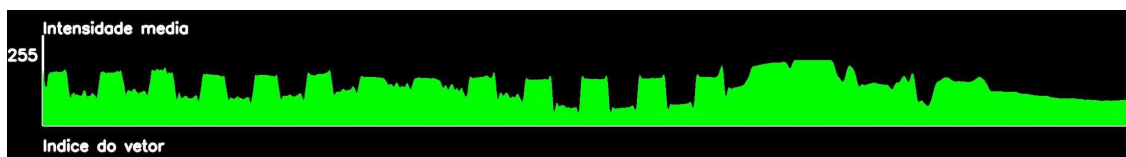
O algoritmo de verificação de padrões claros e escuros, descrito em (Nagaoka, 2019), foi desenvolvido para estimar pontos neste sinal que poderiam representar a *interface* com a água. Enquanto este não explicitamente foi utilizado para realização do cálculo da altura da água, ele foi utilizado para auxiliar o autor a avaliar o resultado que este poderia ter se combinado com o algoritmo de localização de interrupção, explicitado na mesma obra. A intuição por trás desse algoritmo é a verificação da largura das faixas por irregularidades na largura, usando a noção que há um acréscimo linear no comprimento das faixas e que estas se

Figura 12 – Sinal capturado do centro da figura.

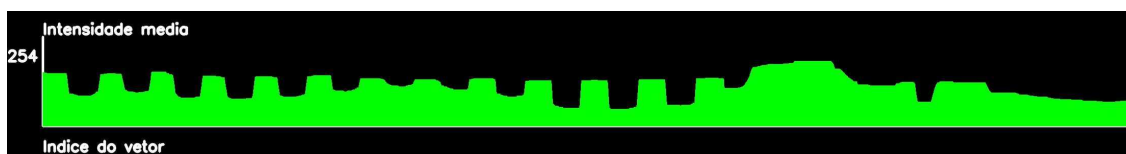
(a) Imagem resultante do processo anterior, rotacionada 90° no sentido anti-horário e convertida para escalas de cinza.



(b) Sinal gerado a partir da intensidade dos pixels da faixa central.



(c) Sinal da figura 12b submetido a um filtro passa-baixa.

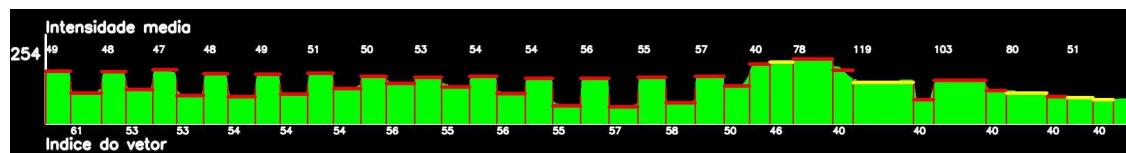


Fonte: (Nagaoka, 2019)

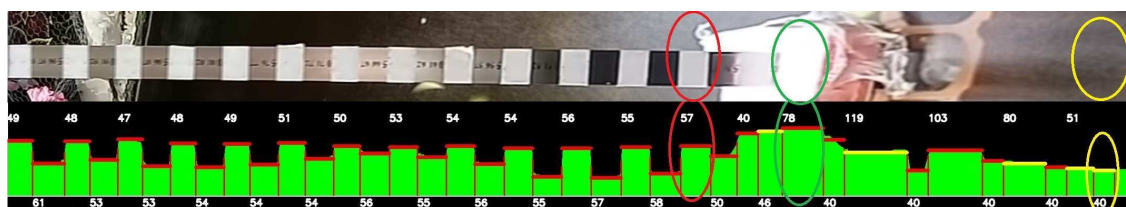
mostram presentes na diferença do valor do sinal. A Figura 13a mostra em retas horizontais amarelas as regiões onde há possível *interface* com a água e uma comparação deste resultado com a imagem real está na Figura 13b.

Figura 13 – Detecção de possíveis *interfaces* com a água.

(a) Em amarelo, as regiões em que o algoritmo detectou discrepância.



(b) Comparação dos resultados obtidos pelo algoritmo com a imagem real. Regiões circuladas são possíveis regiões de encontro com a água, de acordo com o algoritmo de localização de interrupção



Fonte: (Nagaoka, 2019)

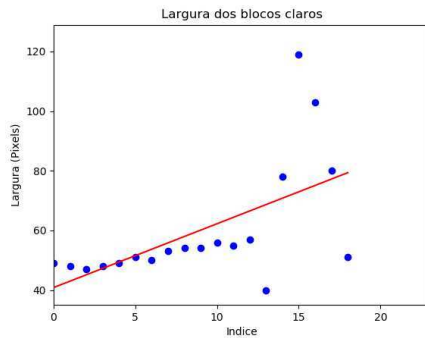
A largura de cada uma das faixas brancas, da esquerda para a direita, é considerada como o valor de  $y$  em uma função  $y = f(x)$ , tal que  $x$  representa o índice destes pontos. Realiza-se então a regressão linear destes pontos de modo a encontrar a reta que melhor descreve o comportamento deles. Se o último ponto, *i.e.* com maior índice, estiver a uma distância desta reta maior que um determinado limiar ele é então classificado como *outlier*,



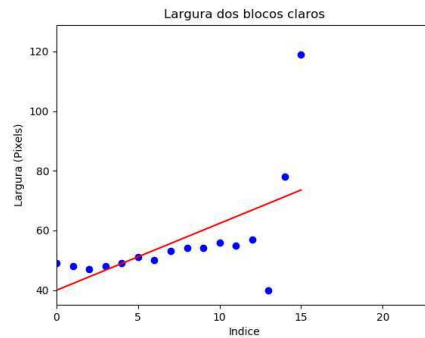
descartado e uma nova regressão é computada. Este ciclo continua até que não restem *outliers*. Um resultado desta abordagem pode ser observado na Figura 14.

Figura 14 – Representação gráfica do algoritmo de eliminação de *outliers*.

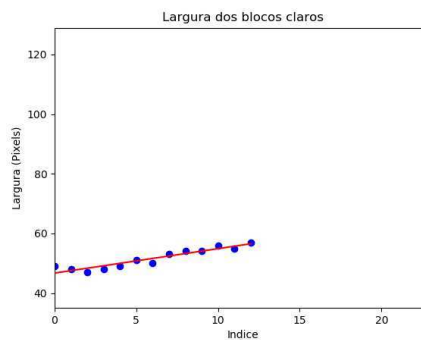
(a) Primeira regressão linear.



(b) Quarta regressão linear.



(c) Sétima regressão linear.



Fonte: (Nagaoka, 2019)

Por fim, após eliminação dos *outliers* realiza-se a inferência da altura da régua com base na relação:

$$H_{regua, gua} = H_{padro, fundo} - \left( \frac{l}{p} ax - h_{boia} \right),$$

tal que:

- $H_{padro, fundo}$  = altura tomada da base da imagem referência até o fundo do corpo d'água
- $a$  = taxa de crescimento da largura do bloco
- $l$  = largura da primeira faixa em centímetros
- $p$  = tamanho em *pixels* da primeira faixa
- $x$  = índice do último ponto resultante do algoritmo de eliminação de *outliers*
- $h_{boia}$  = altura da boia em centímetros

Enquanto o trabalho de (Nagaoka, 2019) propõe uma boa abordagem para detecção da régua, problema não resolvido pelos outros trabalhos, os algoritmos não parecem ter convergido

para solução sólida nos da detecção da borda com a água. Tivesse isso sido feito, poderia então ter sido combinada com a premissa de erro linear na largura das faixas para produzir resultados mais consistentes. Ademais, abordagens que fazem uso explícito de informações sobre a construção da régua como posição e dimensões das faixas brancas e pretas, tornam a solução pouco adaptável, visto que quaisquer alterações podem implicar em alterações nos algoritmos para funcionamento adequado.

### 3 DEFINIÇÕES GERAIS

Neste capítulo serão apresentados os principais conceitos de processamento de imagens acerca desse trabalho, assim como os algoritmos e métodos usados para obter os resultados.

#### 3.1 IMAGEM DIGITAL

Uma imagem digital é um arranjo retangular, com altura  $H$  e largura  $L$ , de *pixels*. Um *pixel* representa a menor unidade de área de uma imagem, composto apenas por uma cor. Todas as cores conhecidas podem ser decompostas em uma mistura de três cores básicas, vermelho (R), verde (G) e o azul (B), cada uma com intensidade variando entre 0 a 255. Portanto, as componentes RGB que compõem uma imagem digital podem ser descritas como:

$$R = \begin{bmatrix} r_{00} & \dots & r_{0n} \\ \vdots & \ddots & \vdots \\ r_{m0} & \dots & r_{mn} \end{bmatrix} \quad G = \begin{bmatrix} g_{00} & \dots & g_{0n} \\ \vdots & \ddots & \vdots \\ g_{m0} & \dots & g_{mn} \end{bmatrix} \quad B = \begin{bmatrix} b_{00} & \dots & b_{0n} \\ \vdots & \ddots & \vdots \\ b_{m0} & \dots & b_{mn} \end{bmatrix},$$

tal que  $0 \leq r_{mn}, g_{mn}, b_{mn} \leq 255$  e  $0 \leq m < L$  e  $0 \leq n < H$

Essas componentes de cor podem ser combinadas em uma única componente de escala de cinzas com a seguinte transformação  $Y = 0.299R + 0.587G + 0.114B$ , tal que  $0 \leq y_{ij} \leq 255$ .

#### 3.2 SCALE-INVARIANT FEATURE TRANSFORM (SIFT)

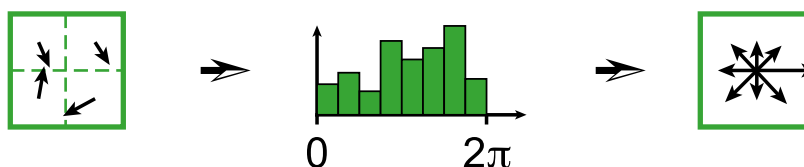
*Scale-Invariant Feature Transform* (Lowe, 2004) é um método para extração de características de imagens, no formato de pontos de interesse e seus respectivos descritores. Estes são invariantes na escala ou rotação da imagem e substancialmente robustos a uma diversidade de condições de distorção, iluminação, ruído ou perspectiva.

Pontos de interesse, ou *keypoints*, possuem coordenada  $(x,y)$  e um raio, cujo valor é proporcional ao fator de escala  $\sigma$  da região de interesse.

Um descritor é a concatenação dos valores de histogramas, os quais são computados a partir da magnitude e orientação de gradientes na imagem (Fig. 15). (Lowe, 2004) determinou experimentalmente que a configuração que obtém os melhores resultados deve ter  $4 \times 4$  histogramas de orientação, com 8 setores de orientação cada, totalizando  $4 \times 4 \times 8 = 128$  valores por *keypoint*. As figuras 16a e 16b exemplificam esse processo para uma configuração  $4 \times 4 \times 4$ .

<sup>1</sup>[https://en.wikipedia.org/w/index.php?title=Luma\\_\(video\)&oldid=978738504](https://en.wikipedia.org/w/index.php?title=Luma_(video)&oldid=978738504)

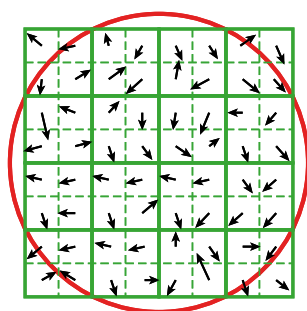
Figura 15 – Construção do histograma de orientações.



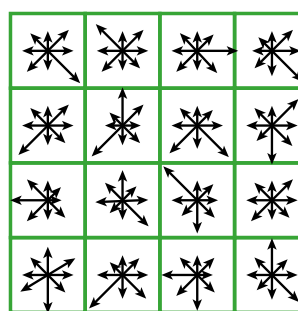
Fonte: [Wikipédia \(2020\)](#)

Figura 16 – Construção de descritores a partir de gradientes.

(a) gradientes da imagem, na configuração  $4 \times 4 \times 4$ .



(b) descritores do *keypoint*.



Fonte: [Wikipédia \(2020\)](#)

Uma das aplicações desta técnica, visto que ela fornece características invariantes, é a busca por similaridades entre imagens. As figuras 17a e 17b mostram pontos de interesse computados pelo método SIFT. Os descritores destes pontos de interesse serão posteriormente submetidos a busca para verificação de similaridades.

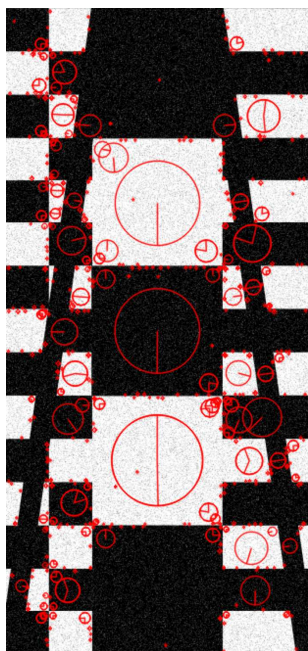
### 3.3 CASAMENTO DE PONTOS DE INTERESSE

O casamento entre *keypoints* de duas imagens, diga-se  $A$  e  $B$ , pode ser feito com o algoritmo *k-nearest neighbors*. Para todo o ponto  $a$  do conjunto  $A$ , este algoritmo encontrará os  $k$  pontos do conjunto  $B$  mais próximos ao ponto  $a$ . Toma-se a distância Euclidiana até o primeiro e segundo vizinhos mais próximos ao ponto,  $x_1$  e  $x_2$  respectivamente, e se  $\frac{x_1}{x_2}$  for menor que uma dada proporção  $r$ , tal que  $r < 1$  então temos que  $x_1$  está significativamente mais próximo do ponto do que  $x_2$ , condição essa necessária para atingir um casamento confiável (Lowe, 2004). Pela alta dimensionalidade do espaço de características haverá um número maior de falsos positivos com distância similar ao segundo vizinho mais próximo.

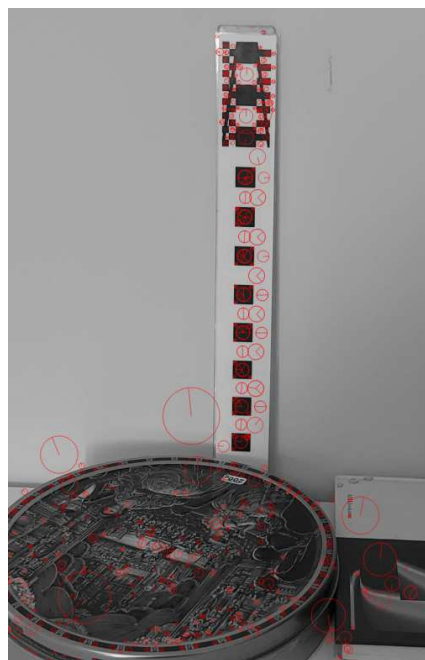
A figura 18 é o resultado deste casamento aplicado aos descritores das figuras 17a e 17b.

Figura 17 – Imagens e seus respectivos pontos de interesse representados graficamente.

(a) Imagem com *keypoints* isolada.

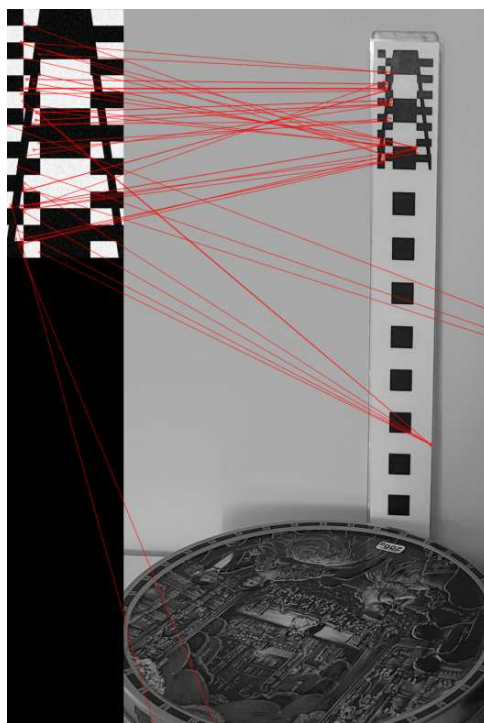


(b) Imagem com *keypoints* inserida em contexto.



Fonte: Autoria própria

Figura 18 – Casamento de descritores entre duas imagens.

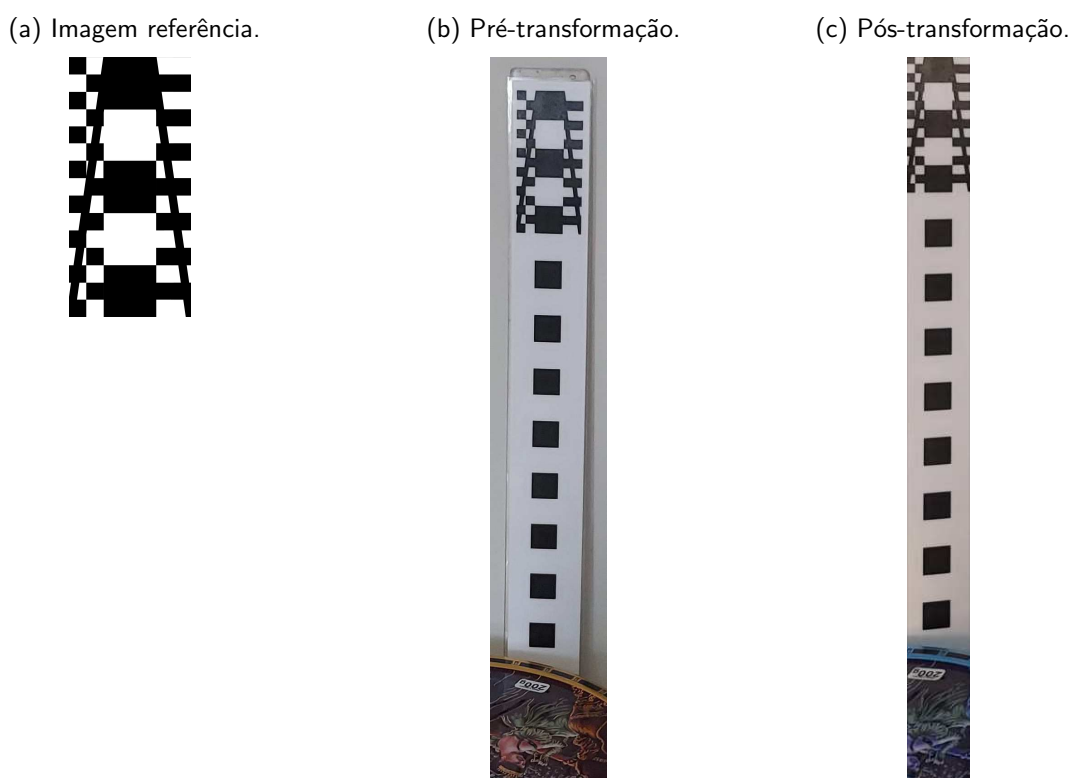


Fonte: Autoria própria

### 3.4 RANDOM SAMPLE CONSENSUS (RANSAC)

RANSAC (Fischler; Bolles, 1981) é um algoritmo que tenta encontrar a melhor transformação dos pontos (*fitting*) através da amostragem aleatória dos dados. Todas essas amostras geram uma função de *fitness* e com isso um conjunto de *inliers* e *outliers*. Um esquema de votação é utilizado para encontrar a melhor função de *fitness*. Este esquema opera sobre as premissas de que pontos ruidosos não possuem voto consistente para qualquer modelo e de que existem pontos suficientes que concordam em um modelo ótimo. Este algoritmo pode ser interpretado como uma técnica de remoção de *outliers*. O RANSAC é utilizado para encontrar a matriz de homografia que melhor transforma as *features* encontradas em uma imagem para a perspectiva da outra. Este algoritmo foi utilizado para transformar a figura 19b para a perspectiva da figura 19a, resultando na figura 19c.

Figura 19 – Transformação da imagem utilizando a matriz de homografia obtida pelo algoritmo RANSAC.



Fonte: Autoria própria

### 3.5 TRANSFORMAÇÃO EM SÉRIES DE FOURIER

Transformação em séries de Fourier é um método para expressar uma dada função em suas componentes periódicas. Chamamos de Transformada de Fourier a função que tem como entrada um sinal periódico e como saída uma soma de componentes de frequência <sup>2</sup>:

<sup>2</sup>[https://ccrma.stanford.edu/~jos/mdft/Fourier\\_Transform\\_FT\\_Inverse.html](https://ccrma.stanford.edu/~jos/mdft/Fourier_Transform_FT_Inverse.html)

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt, \quad \omega \in (-\infty, \infty)$$

Esta equação também possui uma variante para funções de tempo discreto, conhecida como Transformada de Fourier Discreta<sup>3</sup>:

$$X(\omega) = \sum_{n=0}^{N-1} x[n]e^{-j\omega n}, \quad k = 0, 1, 2, \dots, N - 1$$

A vantagem da variante discreta é que esta possui um algoritmo para rápida computação do seu resultado, chamado de *Fast Fourier Transform (FFT)* (Cooley; Tukey, 1965)

### 3.6 SHORT-TIME FOURIER TRANSFORM (STFT)

*Short-time Fourier Transform* ou *Windowed Fourier Transform* é o emprego da Transformada de Fourier em janelas de um dado sinal, de modo a obter o comportamento temporal das frequências presentes, resultando num *espectrograma*. (Smith, 2007)<sup>4</sup> define a STFT de sinais em tempo discreto como:

$$X_m(\omega) = \sum_{n=-\infty}^{\infty} x[n]\omega(n - mR)e^{-j\omega n}, \quad (1)$$

tal que:

- $x[n]$  = sinal na amostra  $n$
- $\omega(n)$  = função de janelamento de comprimento  $m$
- $m$  = comprimento  $m$  da janela, medido em pixels
- $X_m(\omega)$  = DTFT do sinal janelado centrado em  $mR$
- $R$  = tamanho do salto, em amostras, entre DFTFs sucessivas

<sup>3</sup>[https://ccrma.stanford.edu/~jos/mdft/Discrete\\_Time\\_Fourier\\_Transform.html](https://ccrma.stanford.edu/~jos/mdft/Discrete_Time_Fourier_Transform.html)

<sup>4</sup>[https://ccrma.stanford.edu/~jos/sasp/Short\\_Time\\_Fourier\\_Transform.html](https://ccrma.stanford.edu/~jos/sasp/Short_Time_Fourier_Transform.html)

## 4 MEDIÇÃO DO NÍVEL D'ÁGUA

Este capítulo apresentará o procedimento para inferência do nível d'água. A Seção 4.1 traz detalhes da construção da régua, assim como modificações ao modelo proposto por (Nagaoka, 2019). A Seção 4.2 detalha a detecção da régua posicionada em um cenário, similarmente ao trabalho que o antecede. Na Seção 4.3 é descrito como o resultado da seção anterior é aprimorado. A Análise de Fourier, em particular a Transformada Janelada de Fourier (ou *STFT*) será utilizada para detecção da *interface* da régua com a água. Este capítulo termina com a Seção 4.5, onde será demonstrada a forma simplificada que o cálculo do comprimento pode tomar.

### 4.1 COMPONENTES DA RÉGUA

A régua deve ser composta por duas partes:

- uma imagem a ser posicionada no topo da régua, chamada de *imagem referência*, a qual é utilizada para detecção da região de interesse;
- uma região alongada contendo um sinal periódico, utilizada para cálculo da altura e determinação da interface com a água.

Para que haja um casamento adequado de descritores é necessário que a imagem referência possua *alguma* característica única, i.e. *alguma* região que contenha características não encontradas em outras partes da imagem. Quanto maiores as características, maiores são as chances de combinação com uma referência inserida em um cenário com baixa resolução. Quanto mais detalhada, melhores são os casamentos de descritores em um cenário de alta resolução. Um balanço deve ser atingido para obter boa performance.

A região do sinal periódico deve possuir frequência alta o suficiente para realização de medidas adequadas, mas não tão alta a ponto de ser interpretada como ruído da imagem. A resolução do aparato de captura deve ser respeitada. Na prática, as faixas da régua na imagem obtida pela câmera devem ser distinguíveis para alguém que à observa.

Neste trabalho, foi produzida uma imagem referência que estende o sinal da outra parte da régua. Isso foi feito para dar ao procedimento a capacidade de inferência em situações em que somente a imagem referência não esteja submersa. Vale ressaltar que a abordagem adotada para medição torna o procedimento resistente a eventuais diferenças na fase entre os sinais presentes nas duas partes da régua, contanto que a frequência seja a mesma.

O formato escolhido para a imagem referência está disposto na Figura 20a. A faixa central é composta de dois períodos e meio da frequência presente no restante da régua, número este tomado de forma arbitrária. Isso é feito para que seja possível lidar com situações em que a régua ficasse quase que completamente submersa e uma frequência adequada não pudesse ser calculada somente com a outra parte da régua. Também permite que um número mínimo



de períodos seja exigido para realização da medição, através da contagem destes.

Os lados esquerdo e direito são compostos por um padrão quadriculado e retangular, respectivamente, cortados por uma diagonal na cor preta. O intuito foi criar características inambíguas para que o processo de localização da imagem referência ocorra com sucesso.

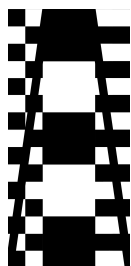
Para que a medida seja correta é **crucial** que a régua mantenha-se reta sob condições adversas. Recomenda-se montagem sobre estruturas rígidas, e.g. coluna de ponte.

Não foi empregada uma boia neste trabalho. Apoiando-se sobre a intuição do algoritmo da Seção 4.4, deseja-se provar que a régua pode ser utilizada para realizar medições sem uso de boia.

A régua utilizado neste trabalho é representado pela Figura 20b.

Figura 20 – Modelo de construção da régua.

(a) Padrão com mesclagem de sinal periódico.



(b) Régua completa, em escala reduzida.



Fonte: Autoria própria

## 4.2 EXTRAÇÃO DA REGIÃO DE INTERESSE (ROI)

O procedimento para extração da região de interesse começa com a conversão da imagem de referência,  $I_{ref}$ , e da imagem da régua no cenário,  $I_{real}$ , para escala de cinza. Em seguida, computam-se os pontos de interesse e seus descritores utilizando o método SIFT, como elaborado na Seção 3.2.

Com a finalidade de casar as similaridades entre as duas imagens, é necessário submeter estes descritores a um combinador de força bruta. Este procedimento é realizado exatamente como descrito na Seção 3.3. Em [Feature Matching — OpenCV Python Tutorials \(2021\)](#) descreve como isso pode ser feito através de bibliotecas do projeto OpenCV.

É utilizado o algoritmo RANSAC (Seção 3.4) para encontrar o casamento entre os pontos que melhor transforma a imagem referência da régua  $I_{real}$  para a perspectiva, rotação e escala da imagem referência informada  $I_{ref}$ . O resultado deste algoritmo é a função de

homografia entre as imagens. [OpenCV: Feature Matching + Homography to find Objects \(2021\)](#) demonstra de forma prática como o casamento de descritores aliado a este algoritmo é utilizado para obtenção deste resultado.

Esta matriz é então utilizada para transformar a imagem. Ela, a imagem  $I_{real}^T$ , altura  $h2$  e largura  $l3$  desejadas são entradas da função de transformação de perspectiva *warpPerspective*, como esclarecido em [OpenCV: Geometric Image Transformations \(2021\)](#). A saída desta função é, por fim, a região de interesse. O resultado obtido é exibido na Figura 21.

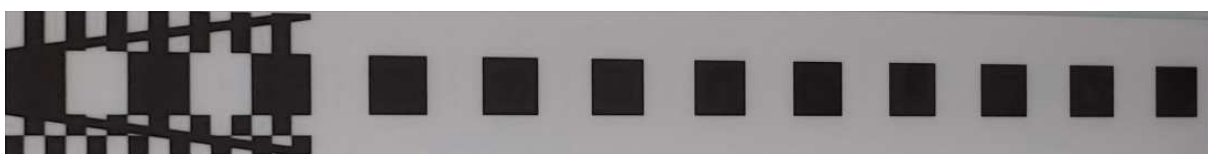
$$h = altura(I_{ref}^T) \quad (2)$$

$$l = \arg \max(altura(I_{real}^T), largura(I_{real}^T)) \quad (3)$$

Com esse resultado, é possível também obter uma linha central da imagem, definida simplesmente por:

$$c = \lfloor \frac{h}{2} \rfloor \quad (4)$$

Figura 21 – Extração da *ROI* para imagens de entrada 20a (referência) e 20b (cena).



Fonte: Autoria própria

### 4.3 WARPING AUXILIADO POR ANÁLISE DE FOURIER

Para fins de notação, o resultado do processo descrito 4.2 será representado pela matriz  $M$  (5).

Como apresentado em ([Nagaoka, 2019](#)), a intensidade luminosa dos *pixels* na faixa central produz um sinal de onda quadrada. Para filtrar ruídos da captura e reduzir erros, a amostragem 6 é realizada.

$$A = (a_{ij}) \in \{\mathbb{Z}^* \cap [0, 255]\}^{m \times n} \quad (5)$$

$$x[k] = \frac{1}{255N} \sum_{p=c-\lfloor N/2 \rfloor}^{c+\lfloor N/2 \rfloor} (a_{pk}), \quad 0 \leq k < n \quad (6)$$

tal que:

- $x[k]$  = valor do sinal no *pixel*  $k$   
 $N$  = largura da faixa  
 $c$  = valor calculado pela equação 4  
 $a_{pk}$  = *pixel* da linha  $p$  e coluna  $k$  da matriz  $A$

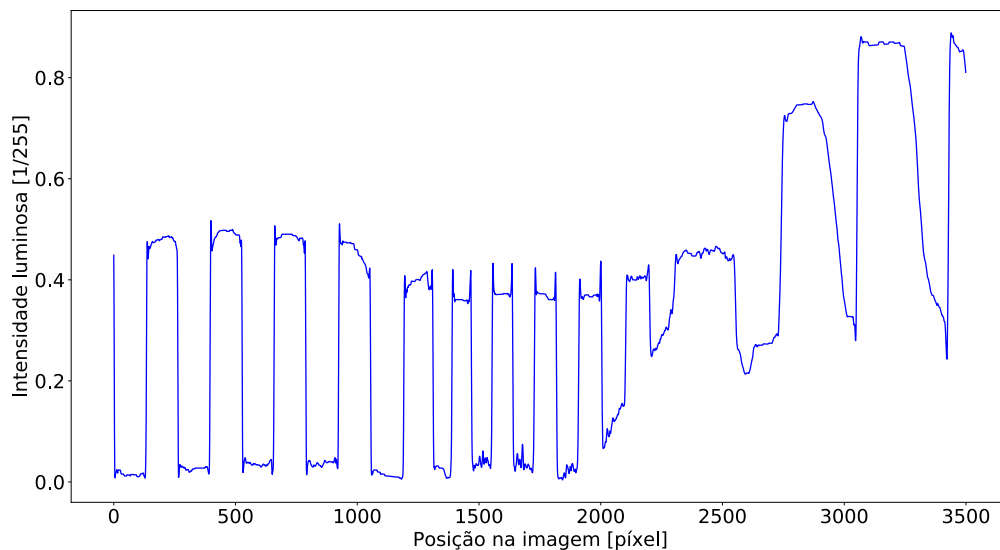
O resultado dessa amostragem é exemplificado pela Figura 22b.

Figura 22 – Exemplo de extração de sinal a partir da imagem.

(a) Imagem transformada para perspectiva.



(b) Sinal extraído da imagem 22a.



Fonte: Autoria própria

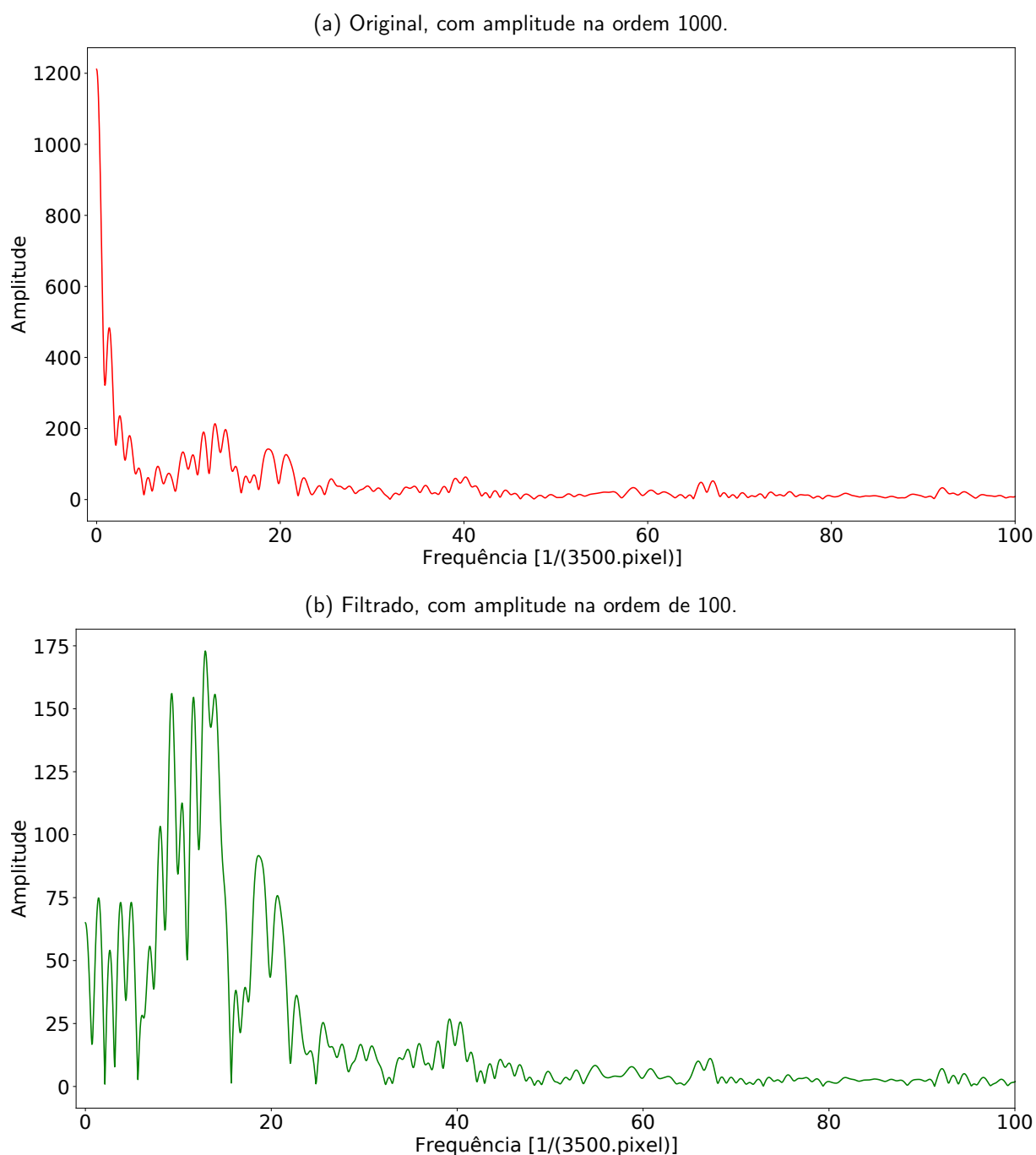
A seguir, são extraídas as componentes de frequência do sinal através do algoritmo *FFT* (3.5). A saída dessa transformada é um conjunto de frequências **discretas** e suas respectivas amplitudes. Por este motivo, não é possível expressar adequadamente a frequência deste sinal se a quantidade de amostras for igual ao comprimento deste. Isso pode ser feito com a técnica *zero-padding* (Smith, 2007)<sup>1</sup> para extensão do sinal. O intuito aqui é utilizar a DTFT para medição de frequências em tempo contínuo.

Antes de continuar com o processamento, é necessário filtrar as componentes de frequência muito baixas ( $f < 2$  ciclos). Isto é implementado com um filtro *Butterworth* (Smith,

<sup>1</sup>[https://ccrma.stanford.edu/~jos/mdft/Zero\\_Padding.html](https://ccrma.stanford.edu/~jos/mdft/Zero_Padding.html)

2007)<sup>2</sup>. A diferença pode ser vista entre as Figuras 23a e 23b, sobre o sinal estendido via *zero-padding*. O valor 3500 presente na escala do eixo  $x$  destas figuras é o comprimento da imagem transformada.

Figura 23 – Componentes de frequências que compõem o sinal 22b.



Fonte: Autoria própria

Equipado da Figura 23 e do entendimento que a Transformada de Fourier produz componentes de frequência no eixo  $x$  e suas respectivas amplitudes no sinal no eixo  $y$ , pode-se

<sup>2</sup>[https://ccrma.stanford.edu/~jos/filters/Butterworth\\_Lowpass\\_Filter\\_Example.html](https://ccrma.stanford.edu/~jos/filters/Butterworth_Lowpass_Filter_Example.html)

explicar a intuição do procedimento a seguir. *Rotacionar imagem em torno dos eixos  $(x,y,z)$  em busca da orientação que produz a mais alta amplitude dentre as componentes de frequências.* Trabalha-se sob a premissa que o sinal de onda quadrada produzido pela régua, quando perfeitamente ajustada, possui a maior amplitude na mesma frequência desta onda quadrada, pois esta será a única frequência periódica presente. Para encontrar o resultado mais próximo deste ideal, utiliza-se um algoritmo de otimização. Este trabalha com uma função de *fitness* que rotaciona a imagem em torno dos três eixos e para cada rotação extrai-se o sinal, sua respectiva Transformada de Fourier e a amplitude predominante resultante desta.

Em experimentos iniciais, para resolver o problema de otimização inserido neste trabalho foi utilizado o algoritmo *Particle Swarm Optimization* (Kennedy; Eberhart, 1995), sob a premissa de que o espaço de busca poderia ser imenso. No entanto, notou-se que a extração da região de interesse (detalhada na Seção 4.2) reduz significativamente o tamanho deste espaço, portanto limitam-se os ângulos de rotação com que o algoritmo de otimização deve trabalhar. Foram testados outros algoritmos de otimização<sup>3</sup> e isso foi comprovado. Destes, o que obteve melhor desempenho foi o algoritmo de Powell. O algoritmo da Direção Conjugada de Powell (Powell, 1964) foi escolhido dentre uma coleção<sup>4</sup> por melhor pesquisar o espaço de soluções e atingir resultado similar ao *PSO*. Aplicado ao sinal da Figura 23 este algoritmo produz o resultado apresentado na Figura 24b. A Figura 24c compara as componentes de frequências dos sinais obtidos das Figuras 24a (linha verde) 24b (linha vermelha).

---

**Algoritmo 1:** Função de fitness utilizada para buscar amplitude máxima.

---

**Input:**  $roi, \phi, \theta, \gamma$

**Output:**  $amplitude_{max}$

$imagem' \leftarrow$  rotação de  $roi$  ao longo do eixo  $x$  em  $\phi$  radianos

$imagem' \leftarrow$  rotação de  $roi$  ao longo do eixo  $y$  em  $\theta$  radianos

$imagem' \leftarrow$  rotação de  $roi$  ao longo do eixo  $z$  em  $\gamma$  radianos

$espectro \leftarrow FFT(imagem')$

$amplitude_{max} \leftarrow \arg \max espectro(x)$

---

#### 4.4 DETECÇÃO DA INTERFACE COM A ÁGUA

Esta seção descreve uma solução para outro problema encontrado por (Nagaoka, 2019), a busca pela *interface* com a água. A detecção é realizada a partir da análise do comportamento, *i.e* amplitude da componente de frequência, ao longo da linha horizontal central da região de interesse. Primeiramente, a métrica que fornece dados para tal análise é a Transformada Janelada de Fourier (3.6). Da definição 1, obtém-se  $X_m(\omega)$  para  $R = 1$  e  $m$  calculado pela equação (7):

<sup>3</sup><https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html>

<sup>4</sup><https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html>

Figura 24 – Resultado do processo de otimização.

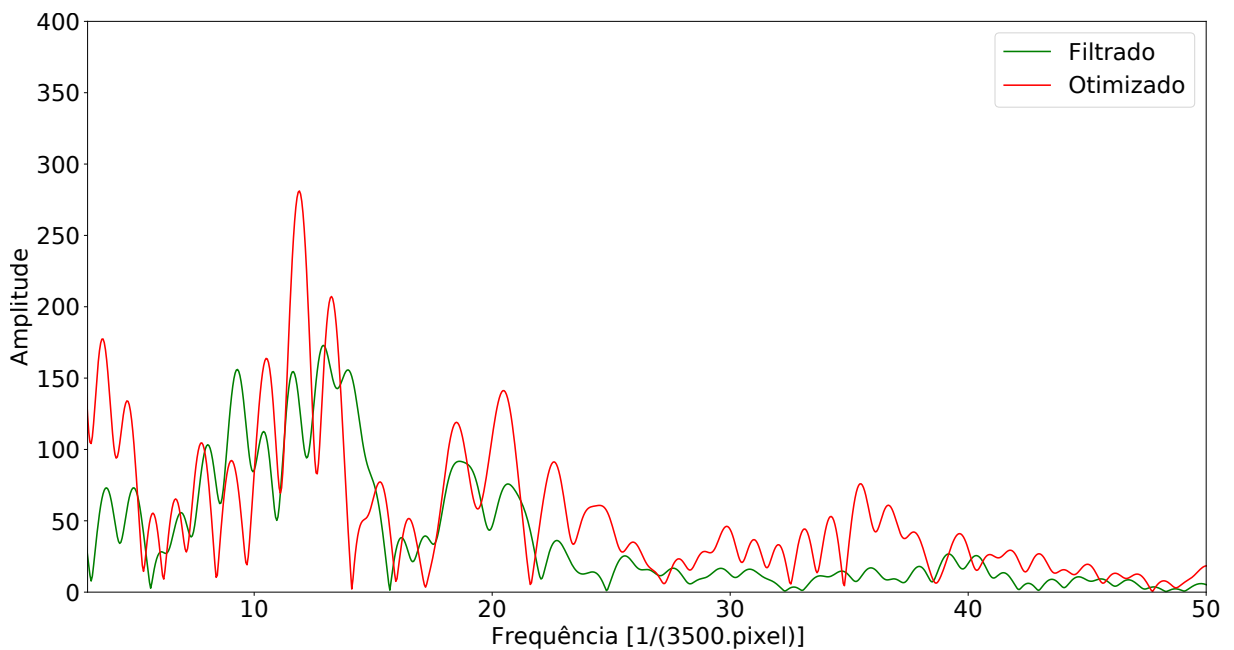
(a) Imagem da régua antes do processo de otimização.



(b) Figura 24a rotacionada para direção conjugada de Powell.



(c) Comparativo entre componentes de frequência do sinal pré (verde) e pós (vermelho) otimização.



Fonte: Autoria própria

$$m = \frac{f_{optimal}}{resolution \|x_n\|}, \quad (7)$$

tal que:

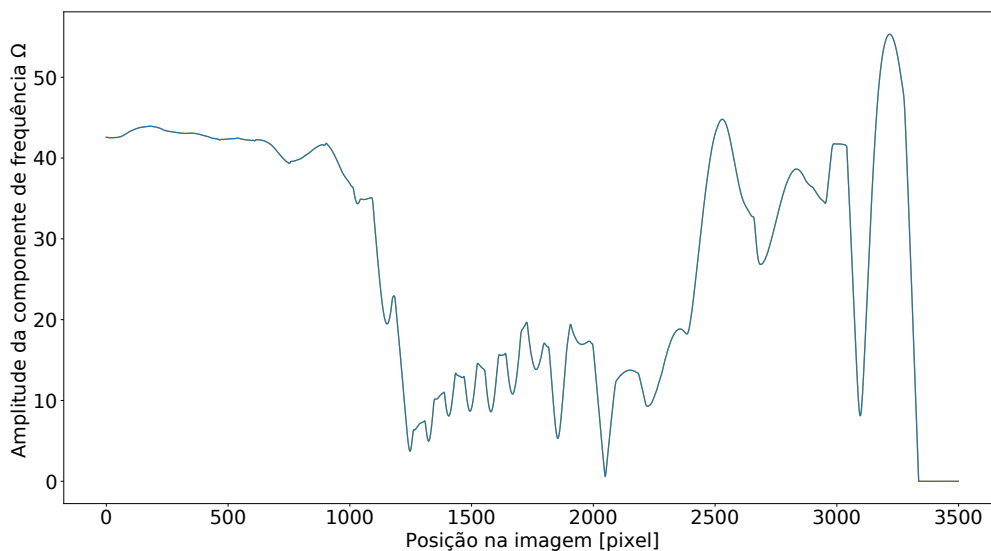
- $m$  = comprimento de um período de  $f_{optimal}$  em pixels
- $f_{optimal}$  =  $\arg \max(X(\Omega))$
- $\Omega$  = frequência obtida através do processo 4.3
- $\|x_n\|$  = cardinalidade do sinal  $x_n$
- $resolution$  = constante utilizada para extensão do sinal (*zero-padding*)

O produto  $resolution\|x_n\|$  e o fator de ajuste da frequência para a nova resolução do sinal.  $X_m(\omega)$  é um espectrograma que expressa a amplitude das componentes de frequência presentes na janela  $\omega(n-mR)$  ao longo do sinal. Fazemos uso dele para obter o comportamento de  $\Omega$ . Para isso, toma-se 8:

$$X_m(\omega)[m], \quad 0 \leq n \leq \|x_n\| - m \quad (8)$$

A Figura 25 traz um exemplo de amplitude obtida através de 8.

Figura 25 – Amplitude da componente de frequência  $\Omega$  ao longo do sinal 22b.



Fonte: Autoria própria

Foi observado durante os experimentos que divisórias com a água caracterizam-se neste gráfico como um rápido decréscimo na amplitude componente de frequência  $\Omega$  comparado à amplitude total desta, i.e.  $amplitude(\Omega)_{\max} - amplitude(\Omega)_{\min}$ . Para encontrar o índice que representa a primeira ocorrência deste evento, foi desenvolvido algoritmo 2:

Também foi observado que a interface com a água está próxima do ponto pré-declínio adicionado de um período de oscilação. A Figura 26b mostra o resultado do algoritmo 2, levando em conta essa observação.

#### 4.5 CÁLCULO DO COMPRIMENTO DA RÉGUA

Para encontrarmos o comprimento da régua fora da água, precisa-se saber onde esta começa, termina e a quantos períodos de oscilação isto equivale. Por fim, converte-se este valor para milímetros informando ao sistema o comprimento de um período em milímetros. Utiliza-se:

**Algoritmo 2:** Busca pelo maior e mais rápido declínio.

---

```

Input:  $X_m(n), m$ 
Output:  $start$ 
 $f \leftarrow X_m(m)$ 
 $k \leftarrow \lfloor \frac{m}{2} \rfloor$ 
 $d \leftarrow \max(X_m(m)) - \min(X_m(m))$ 
 $i \leftarrow 0$ 
while  $i < \|f\| - k$  do
   $range \leftarrow \{i, i + k\}$ 
   $j \leftarrow i + \arg \max(f[range])$ 
  if  $f(j) - f(i + k) \geq d.LOSS$  then
     $start \leftarrow f(j)$ 
    break
  end
end

```

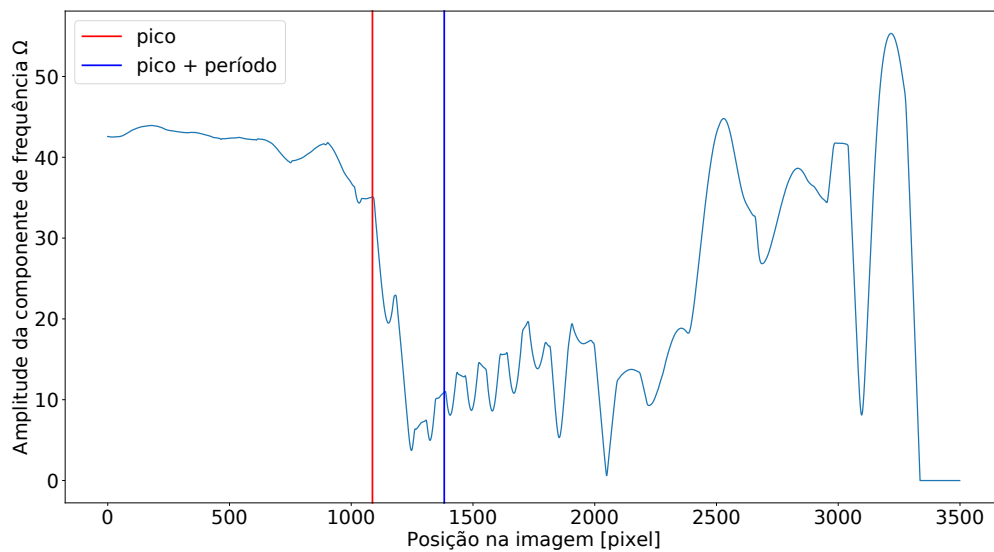
---

Figura 26 – Local estimado da *interface* com a água.

(a) Imagem rotacionada para obtenção de argumento máximo no espectro de frequências.



(b) Comportamento da frequência predominante, com linhas verticais que delimitam região onde estima-se que se encontre a interface com a água.



Fonte: Autoria própria



1. Frequência ( $f$ ), em *ciclos/pixel*. Obtida através do procedimento descrito em 4.3
2. Posição da interface com a água( $x$ ), em *ciclos/pixel*. Obtida através do procedimento descrito em 4.4
3. Posição de início da régua ( $x_0$ )
4. Comprimento de um período ( $C$ ), em milímetros.

Com isso, conclui-se que o comprimento  $L$  em milímetros é 9:

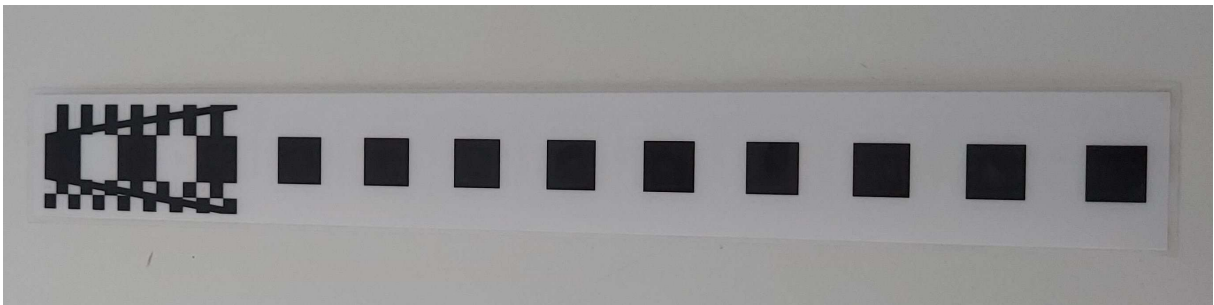
$$L = \frac{(x - x_0)f}{C} \quad (9)$$

## 5 EXPERIMENTOS E RESULTADOS

### 5.1 DESCRIÇÃO DA RÉGUA

Para os experimentos aqui apresentados, foi desenvolvido um protótipo em escala reduzida. As dimensões da imagem referência são 63 milímetros de altura por 30 de largura. O período da oscilação tem comprimento de 25 milímetros. A imagem referência e a seção alongada somam 288 milímetros de comprimento. O protótipo foi impresso em um papel A4, recortado e revestido em plástico. Para torná-lo mais opaco, o revestimento plástico foi lixado com uma lixa d'água de grão 1200. O protótipo foi então montado sobre material acrílico para proporcionar rigidez ao conjunto. O resultado está presente na imagem 27.

Figura 27 – Protótipo da régua em escala reduzida.



Fonte: Autoria própria

### 5.2 DESCRIÇÃO DO PROTÓTIPO DO SISTEMA PROPOSTO

O sistema foi desenvolvido na linguagem Python, por ser compatível com sistemas operacionais *Unix* e *Windows* e por ser a ferramenta mais utilizada para processamento de dados. Fornece bibliotecas e *frameworks*, com suporte e comunidade ativos, para resolução de diversos problemas modernos.

Uma dessas ferramentas é o OpenCV (Bradski G., 2000), projeto que fornece bibliotecas de funções focadas em visão computacional. Este permitiu fácil resolução dos problemas dispostos na Seção 4.2. Também foram utilizadas as bibliotecas SciPy (Virtanen et al., 2020) e NumPy (Harris et al., 2020), as quais fornecem ferramentas para processamento de sinais e cálculo matricial. Para visualização de gráficos, foi utilizada a biblioteca Matplotlib (Hunter, 2007).

As seguintes versões são utilizadas:

1. *Python*, versão 3.8.7
2. *opencv-contrib-python*, versão 4.4.0.46
3. *SciPy*, versão 1.6.0

4. *NumPy*, versão 1.20.1
5. *Matplotlib*, versão 3.3.4

Para captura das imagens, fotos foram tiradas utilizando a câmera de um celular *Galaxy S10 Lite*. A resolução das imagens capturadas por este é 4000x1800. O processamento dos dados foi feito em um *laptop* equipado com um processador *Intel Core i7-8565U* e 16 GB de memória *RAM*.

### 5.3 DESCRIÇÃO DO EXPERIMENTO

Foram realizados 17 experimentos foram realizados com o protótipo em escala reduzida. Essa abordagem pode produzir dados válidos porque imagens não capturam as dimensões físicas de objetos nelas presentes, tampouco são capazes de produzir tais valores reais sem informações acerca do que representam.

Com isso em mente, os experimentos foram feitos com o protótipo descrito na Seção 5.1 disposto em um recipiente com água. A régua foi fotografada em montagens que tinham o intuito de verificar a consistência dos valores calculados sob condições tanto favoráveis quanto adversas. Foram reproduzidos cenários que combinam:

1. iluminação solar parcial ou totalmente incidente
2. ângulos diferentes entre o plano da régua e a câmera
3. presença/ausência de boia
4. turbidez do fluido em que a régua foi submersa, reproduzido adicionando-se terra à água

Em alguns casos, estes foram combinados de modo a maximizar tanto a presença de reflexões da régua sobre a superfície da água quanto a porção submersa deste, deformada devido às propriedades óticas distintas entre o corpo d'água e o ar.

### 5.4 RESULTADOS OBTIDOS

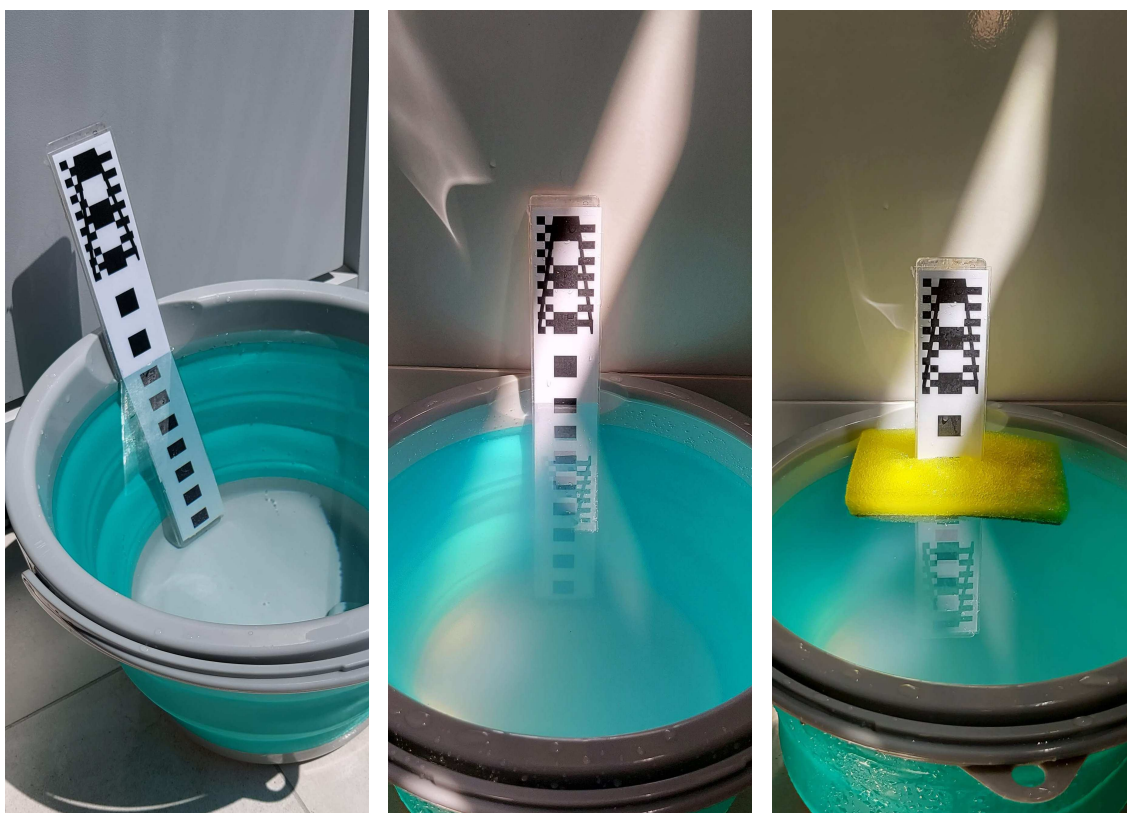
O processo de empenamento coloca a imagem da régua em uma visão frontal, com dimensões  $310 \times 3500$  pixels. A faixa central ocupa aproximadamente 140 desses 310 pixels e a largura desta faixa é 1.3cm, resultando na proporção  $140\text{pixels}/1.3\text{cm} = 107.7\text{pixels}/\text{cm}$ . É importante ressaltar que o tamanho real da régua não é importante, dado que as proporções da imagem resultante mantenham-se as mesmas.

Para realização da amostragem do sinal ao longo da faixa central da imagem, foi utilizada uma faixa com largura de 10 *pixels* ( $N = 10$ ) e perda na amplitude do sinal na janela de 35% ( $LOSS = 0.35$ ). A rotação ao longo dos eixos  $(x, y, z)$  foi limitada para o intervalo  $(\pm 5^\circ, \pm 10^\circ, \pm 0^\circ)$ , respectivamente.

Resultados estão dispostos na Tabela 1. Os comprimentos calculados tidos como aceitáveis, i.e. com *erro* arbitrariamente definido em até 10% , variaram entre 1 e 17 milímetros dos respectivos valores reais.

Figura 28 – Amostras feitas de acordo com o que foi proposto na Subseção 5.3.

- (a) Ângulo oblíquo. Amostra 5 da Tabela 1  
 (b) Incidência de luz solar parcial. Amostra 9 da Tabela 1  
 (c) Presença de boia. Amostra 7 da Tabela 1



- (d) Secção submersa. Amostra 15 da Tabela 1  
 (e) Água turva. Amostra não está presente na tabela, pois a matriz de homografia encontrada distorceu a imagem (erro)



Tabela 1 – Resultados de experimentos realizados.

Amostra	$t_p$ (s)	$t_o$ (s)	$t_d$ (s)	$t_{total}$ (s)	min(mm)	max(mm)	med(mm)	real(mm)	erro(%)
1	0.24	2.40	0.08	2.72	366	391	378	375	0.80
2	0.38	1.07	0.18	1.63	346	371	358	375	4.53
3	0.59	1.10	0.09	1.78	646	671	658	675	2.52
4	0.76	0.75	0.51	2.02	262	275	268	263	1.90
5	0.73	0.80	0.49	2.01	108	122	115	121	4.96
6	1.19	2.24	1.44	4.87	98	111	104	110	5.45
7	1.24	1.34	0.54	3.12	77	90	83	107	22.43
8	1.25	0.73	0.49	2.47	108	121	114	122	6.56
9	1.27	2.33	0.66	4.26	103	116	109	104	4.81
10	1.25	1.67	0.38	3.29	104	117	110	117	5.98
11	1.34	2.87	0.64	4.85	101	114	107	116	7.76
12	1.58	1.74	0.83	4.15	115	128	121	122	0.82
13	1.34	1.84	1.59	4.77	82	95	88	98	10.20
14	1.26	3.20	1.04	5.51	114	127	120	118	1.69
15	1.35	1.25	0.43	3.03	118	131	124	122	1.64
16	1.27	0.72	1.27	3.26	125	138	131	121	8.26
17	1.30	2.00	1.38	4.68	116	129	122	117	4.27

Fonte: Autoria própria

Na Figura 28a foi testada a capacidade de transformação da imagem quando capturada de um ângulo oblíquo. Nas Figuras 28b, 28c e 28e, a incidência de iluminação solar parcial, presença de reflexão da régua sob a superfície da água. Nas Figuras 28a, 28b, 28c, 28d é possível também observar a porção submersa da régua. Na Figura 28c foi utilizada uma esponja para simular uma boia, com o objetivo verificar como esta poderia influenciar no resultado. Por fim, na Figura 28e foi adicionada terra à água para simular a condição de uma régua introduzida em um corpo d'água turva.

Por trazer pouca informação útil para análise da eficácia do algoritmo aqui proposto, quatro amostras não foram adicionados à Tabela 1. Para duas delas, não foi encontrada a matriz de homografia, o que impede que o processo seja concluído. Para as outras duas restantes, foram encontradas matrizes de homografia que distorciam completamente a imagem. Apesar destas amostras não terem sido listadas, serão contabilizadas como total insucesso, i.e. amostras cujo resultado tem erro de 100%.

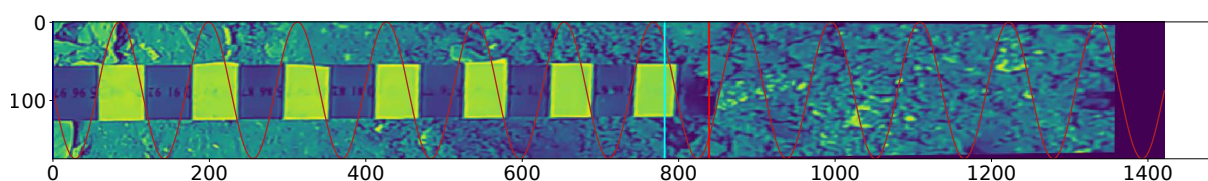
- $t_p$  = tempo para transformação de perspectiva  
 $t_o$  = tempo para otimização  
 $t_d$  = tempo para encontrar a *interface* com a água  
 $t_{total}$  = tempo total de processamento  
 $min$  = comprimento mínimo (ponto de início do decréscimo + meio período)  
 $max$  = comprimento máximo (ponto de início do decréscimo + um período completo)  
 $med$  = média aritmética entre comprimentos mínimo e máximo  
 $real$  = comprimento real  
 $erro$  = erro do valor médio em relação a medida real

As amostras 1 a 3 foram capturadas de dados do trabalho de (Nagaoka, 2019) para comparar a efetividade entre as abordagens. A medida real para estas amostras, presente na Tabela 1 é um valor aproximado, i.e. com base nas imagens e no comprimento das faixas foi estipulado o valor que melhor representasse a realidade. O autor de (Nagaoka, 2019), com uma régua de 100cm de comprimento, concluiu que as melhores estimativas tinham uma diferença de 2.5cm com o valor real, enquanto que as piores, 7.5cm. Com o método proposto neste trabalho obteve-se 0.1cm e 2.4cm, respectivamente, com uma régua de 28.8cm de comprimento. As Figuras 30a) e 30b) expõe as amostras 1 e 3, respectivamente.

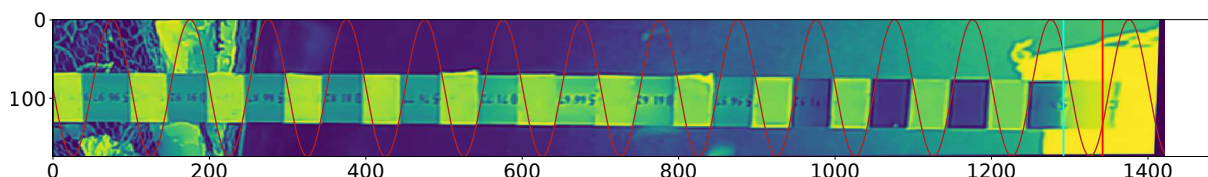
Levando em conta o tamanho das régua utilizados para calcular a diferença percentual, esses valores se transformam em 2.5% e 7.5% de (Nagaoka, 2019) contra 0.4% e 10.67%.

Figura 29 – Figura com representação visual da frequência calculada (senoide em vermelho) e região onde estima-se que esteja a interface com a água (linhas verticais nas cores ciano e vermelho).

(a) Amostra 3 da Tabela 1.



(b) Amostra 2 da Tabela 1.



Fonte: Autoria própria

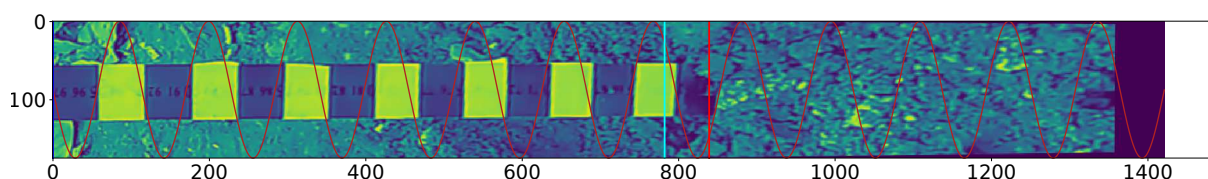
As amostras 7 e 13 apresentaram erro significativo. Nestes casos, o algoritmo 2 falhou ao encontrar a divisória com a água de forma satisfatória. Isso pode ser atribuído a uma



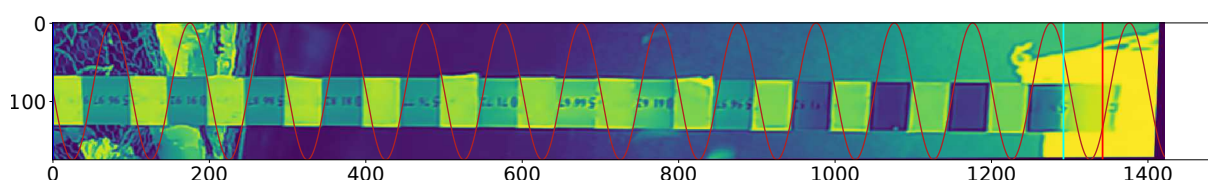
perda na amplitude no sinal menor que o especificado pela variável  $LOSS$  (Fig. 31a) ou por interferência da iluminação externa sobre a régua, fazendo com que o comportamento do sinal torne-se irregular (Fig. 32a, amostra 13).

Figura 30 – Figura com representação visual da frequência calculada (senoide em vermelho) e região onde estima-se que esteja a interface com a água (linhas verticais nas cores ciano e vermelho).

(a) Amostra 7 da Tabela 1.



(b) Amostra 13 da Tabela 1.



Fonte: Autoria própria

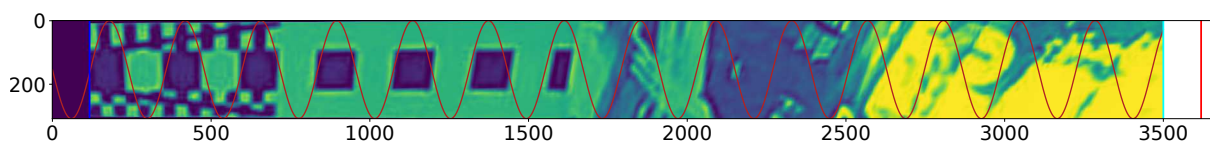
O valor  $LOSS = 0.35$  foi determinado experimentalmente como o valor que trouxe os melhores resultados. Valores mais baixos que este resultam numa ocorrência maior de falsos negativos, i.e. oscilações no comportamento do sinal podem acionar a condição de interrupção do algoritmo e valores maiores não capturam tão bem transições mais sutis.

Houve casos em que falhas no processo de extração de região de interesse levaram à total distorção da imagem. Isso pode ser atribuído à qualidade da captura, mas também à qualidade da imagem de referência. Enquanto que a imagem referência escolhida cumpre com os requisitos necessários para o funcionamento do projeto, ela parece não produzir muitas *features* únicas quando processada pelo algoritmo *SIFT*. A Figura 33 mostra que a maioria dos casamentos se limitou à faixa esquerda da imagem referência, o que pode indicar que o algoritmo *RANSAC* funciona graças a esta pequena região. Com isso entende-se que mais características únicas na imagem referência, i.e. da perspectiva do algoritmo *SIFT*, são mais importantes que características repetidas, **independentemente** da escala ou vizinhança destas.

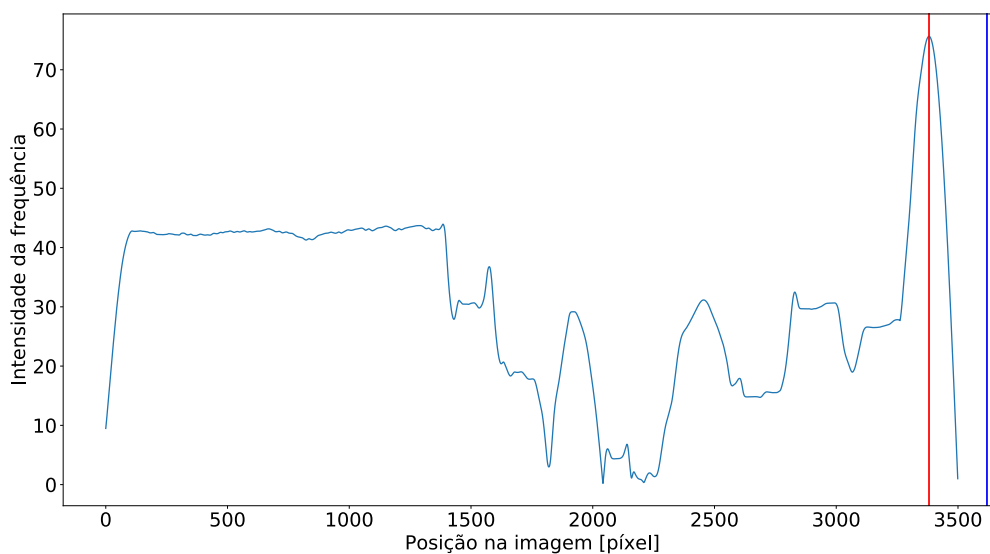
Não menos relevante é o fato de que restrições de desempenho impostas sobre o algoritmo *RANSAC* possam reduzir sua eficácia, afinal *robusto* é diferente de imune a condições adversas.

Figura 31 – Caso de falha ao detectar divisória com a água.

(a) Amostra 7 da Tabela 1.



(b) Intensidade da frequência ótima ao longo da régua da Figura 31a.



Fonte: Autoria própria

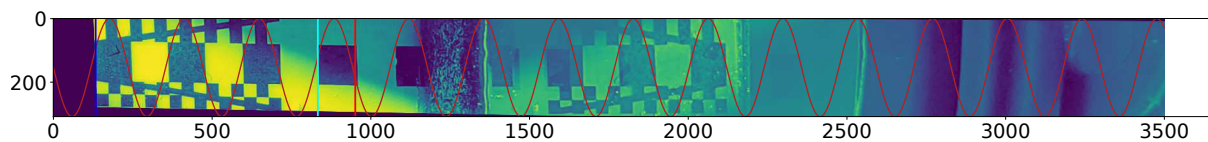
## 5.5 DISCUSSÃO

Os histogramas da Figura 34 sugerem que o método proposto por este trabalho é capaz de obter resultados confiáveis, pois observa-se consistência nas medições obtidas. O histograma 34a mostra a distribuição de erro das 21 medidas feitas. Os valores estão na sua maioria presentes na faixa 0 a 10% (a), duas medições na faixa 10% a 30% (b) e 4 medições falharam (c). As medições na faixa (b) compartilham de um mesmo problema, que é a falha na detecção da borda com a água. Os erros correspondentes a faixa (c) ocorreram por falha total ao casar descritores da imagem referência ou produziram uma matriz de homografia que distorceu completamente a imagem. Enquanto que a imagem referência escolhida serviu como parte do aparato de medição no que concerne à localização da frequência ótima, esta possui características similares em vários setores, i.e. quadrados brancos e pretos repetidos em diversos setores, resultando em falso positivo ao casar descritores. Conclui-se que para melhor desempenho desta parte do processo seja reprojeta esta imagem, levando em consideração o equilíbrio entre resolução e não-ambiguidade mencionados na Seção 4.1. A região central

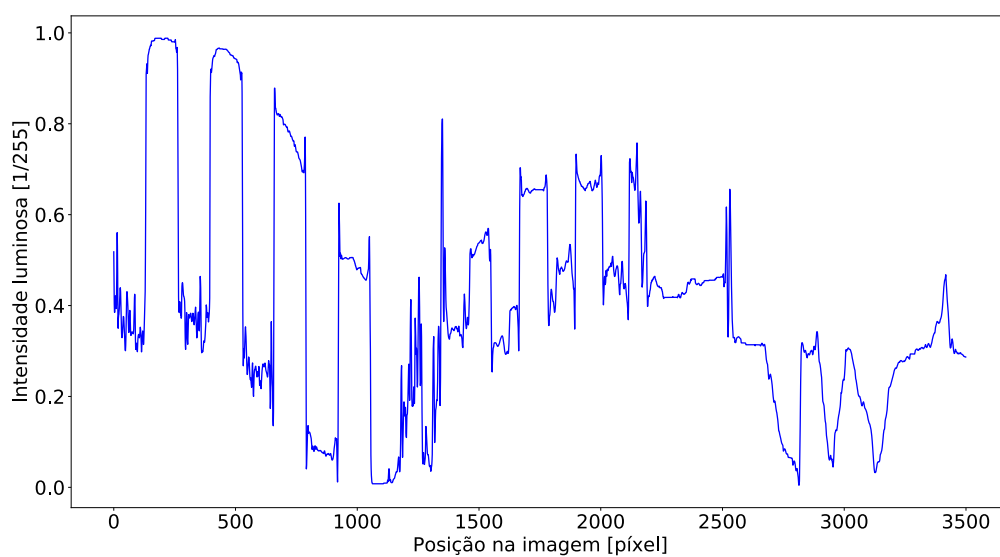


Figura 32 – Caso de falha da detecção da divisória com a água.

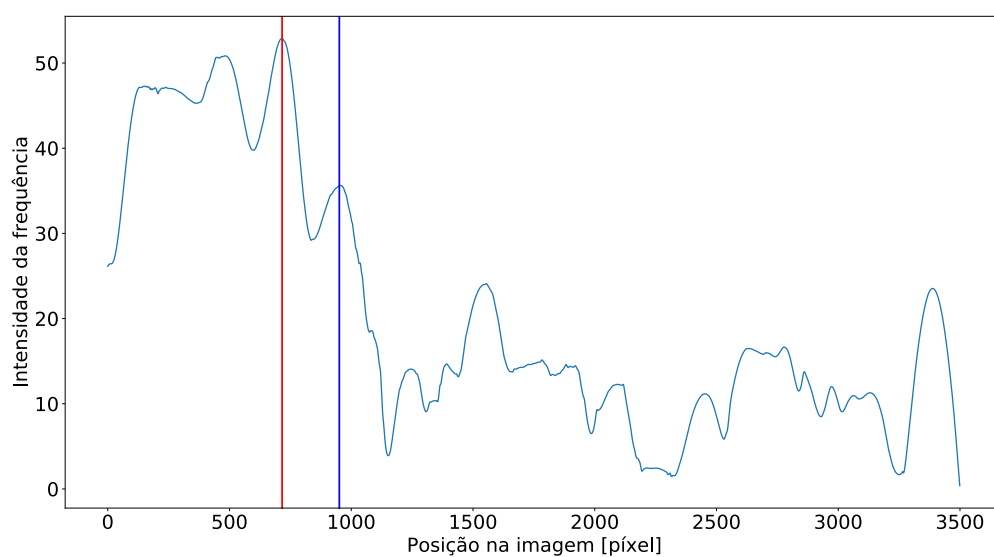
(a) Amostra 13 da Tabela 1.



(b) Sinal ao longo da régua da Figura 32a.

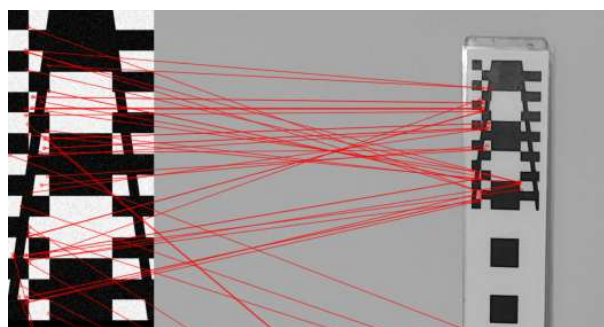


(c) Intensidade da frequência ótima ao longo da régua da Figura 32a.



Fonte: Autoria própria

Figura 33 – Região da Figura 18 que evidencia mal rendimento, i.e. poucas características inambíguas



Fonte: Autoria própria

vertical desta imagem, que possui o mesmo padrão de quadros do restante da régua, pode ser mais fina para oferecer uma região maior para *features*, melhorando o desempenho da localização da região de interesse.

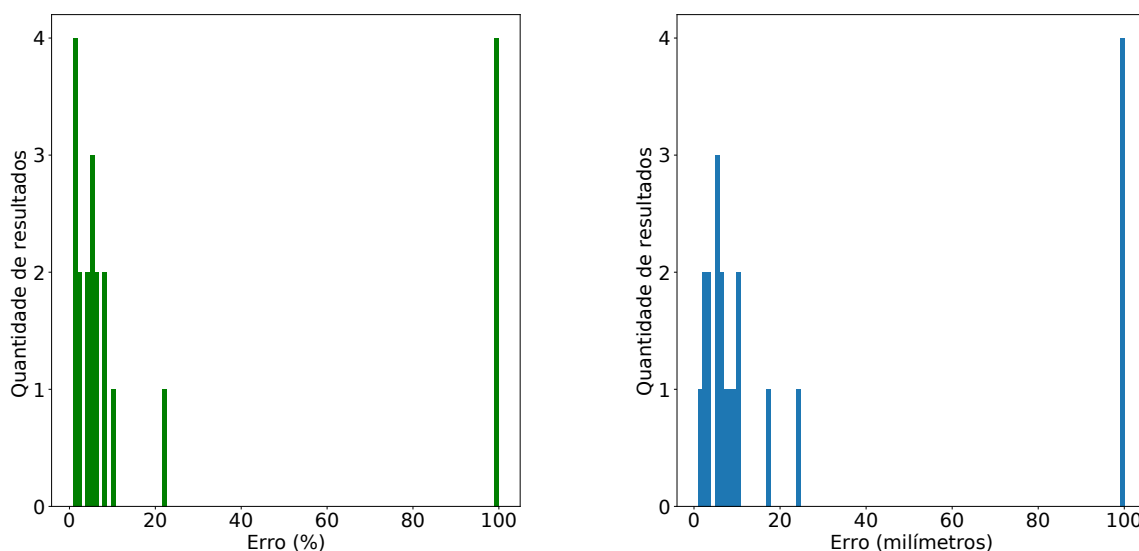
O histograma 34b mostra a relação entre as medidas e o erro em milímetros. Parte das amostras foi obtida do trabalho de (Nagaoka, 2019). Por sua vez, estas representam valores de uma régua com seção limnométrica de comprimento 1000 milímetros. Os erros obtidos variaram entre 3mm e 17mm. Estes resultados concordam com a intuição que um modelo em escala reduzida pode servir como ferramenta de análise e verificação de um modelo em escala real. No entanto, ainda é importante observar a influência da escala na variação do resultado final.

A remoção da imagem referência (Fig. 35b), quando necessário, e a rotação da imagem resultante da busca pela frequência ótima (Fig. 35c) também podem interferir negativamente no resultado se uma seção da régua acabar sendo removida. Pode-se notar que isso ocorre comparando o começo da faixa intercalada (Fig. 35). Da Figura 35a para a Figura 35b, nota-se que a primeira faixa branca está mais curta. Da Figura 35b para a Figura 35c, nota-se que a primeira faixa branca foi completamente removida da imagem. Modificar o algoritmo de rotação de modo à garantir que a imagem inteira está dentro do quadro de rotação pode ser implementado de forma simples e trazer melhora significativa no desempenho.

Reflexões e transparência não produzem impacto significativo no resultado. Da Figura 22 conclui-se que a refração da luz proveniente da porção submersa da régua altera as dimensões e possivelmente o ângulo desta porção, conseqüentemente alterando a frequência do sinal neste trecho. Mesmo que pouco da régua esteja fora da água, o impacto da refração da luz e da turbulência do fluido acabam por produzir distorções que reduzem significativamente a chance do algoritmo de otimização favorecer a frequência que corresponde a seção submersa. Das Figuras 32a e 32b, conclui-se que a reflexão da régua sobre a superfície d'água não tem impacto negativo sobre o processo de otimização, pois ainda que a reflexão fosse perfeita, ela produziria um sinal com a mesma característica que a porção da régua acima da superfície. A reflexão da régua sobre a água parada pode ser compreendida como a mesma que ocorreria se um espelho **reto** substituisse a superfície deste líquido.

Figura 34 – Histogramas de resultados.

- (a) Histograma de erros percentuais. Casos de não funcionamento foram inseridos como ocorrências de erro 100%.
- (b) Histograma de erros em milímetros. Casos de não funcionamento foram inseridos com erro 100mm.



Fonte: Autoria própria

De modo geral, os passos descritos no Capítulo 4 podem ser refinados de forma individual para melhora no desempenho do processo, visto que há pouca acoplagem entre esses. Ademais, para implementação nos diversos cenários em que a proposta deste trabalho pode ser útil, recomenda-se que:

- Seja realizada a calibragem da câmera para remover efeitos de lente, e.g. distorção radial
- Construção de um medidor reto e resistente a deformações que possam ser causadas intencionalmente ou por condições naturais como vento ou corrente d'água
- Nenhum passo da medição leva em conta as cores do medidor. Portanto este pode ser iluminado artificialmente para realização da medição, para realização de medições noturnas, por exemplo
- Substituição da fotografia por "imagem média de vídeo", como foi feito no trabalho de (Nagaoka, 2019), para realização de medição sob chuva

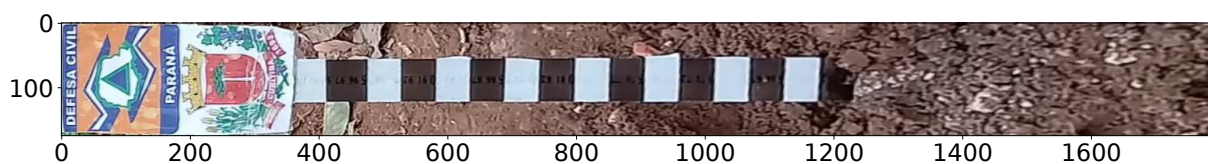
## 5.6 AMOSTRAS E RESULTADOS

Esta seção traz as Figuras 36, 37 e 38. Cada imagem da régua nestas figuras representa uma linha e estas linhas seguem a seguinte ordem:

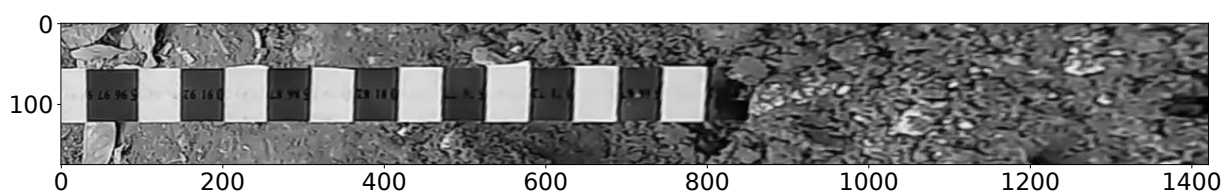
1. Amostra  $n$ , após transformação com matriz de homografia
2. Empenamento da amostra  $n$  com auxílio do algoritmo de otimização

Figura 35 – Comparação entre borda esquerda da região de interesse a medida que é processada.

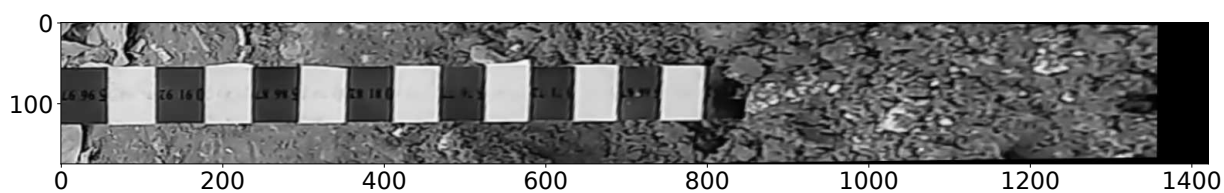
(a) Região de interesse.



(b) Região de interesse sem imagem de referência.



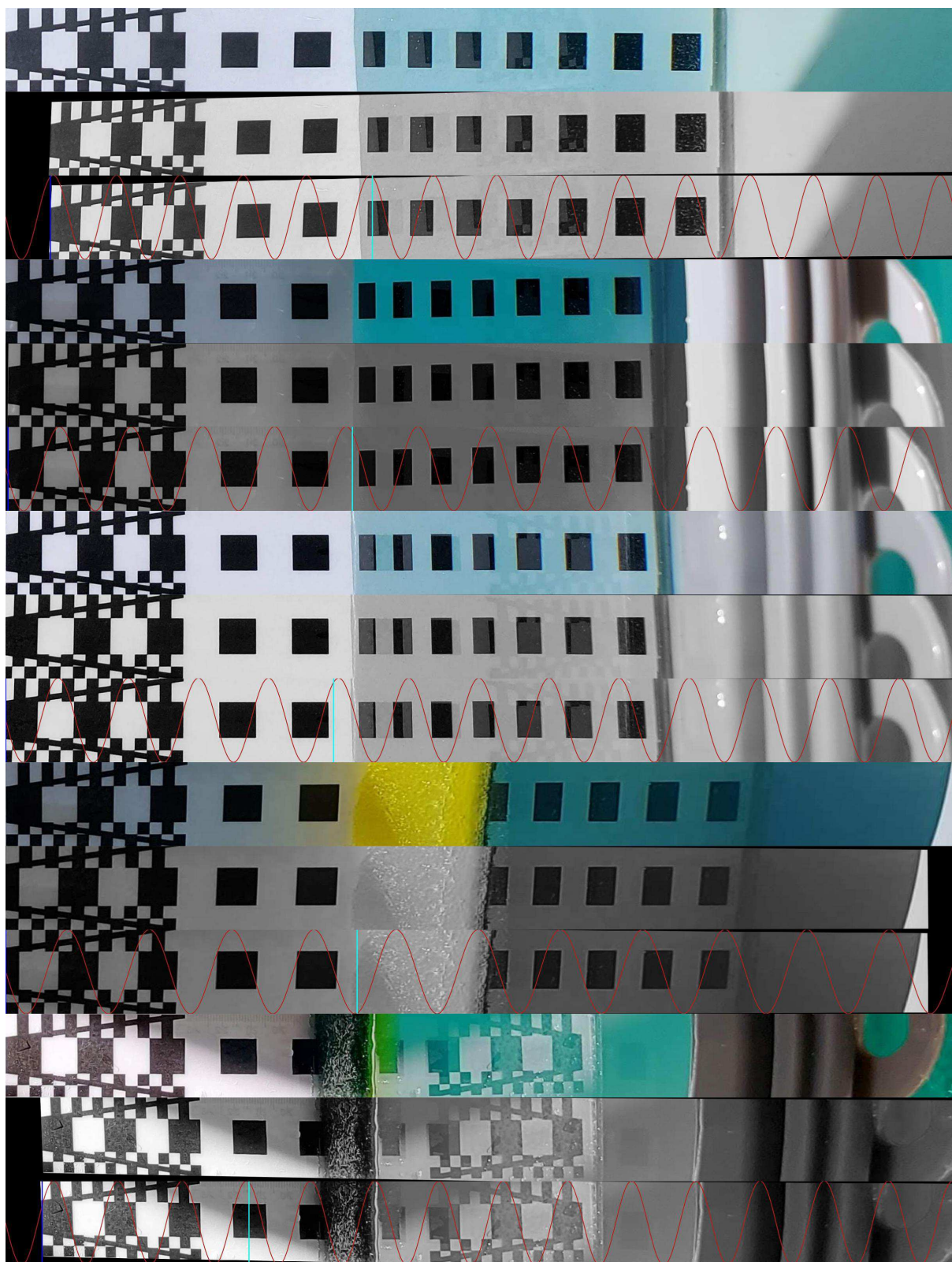
(c) Região de interesse sem imagem de referência rotacionada para ângulo ótimo.



Fonte: Autoria própria

3. Empenamento da amostra  $n$  com auxílio do algoritmo de otimização, com marcações para indicar frequência (senoide), início (linha azul), região de encontro com a água (linha ciano)

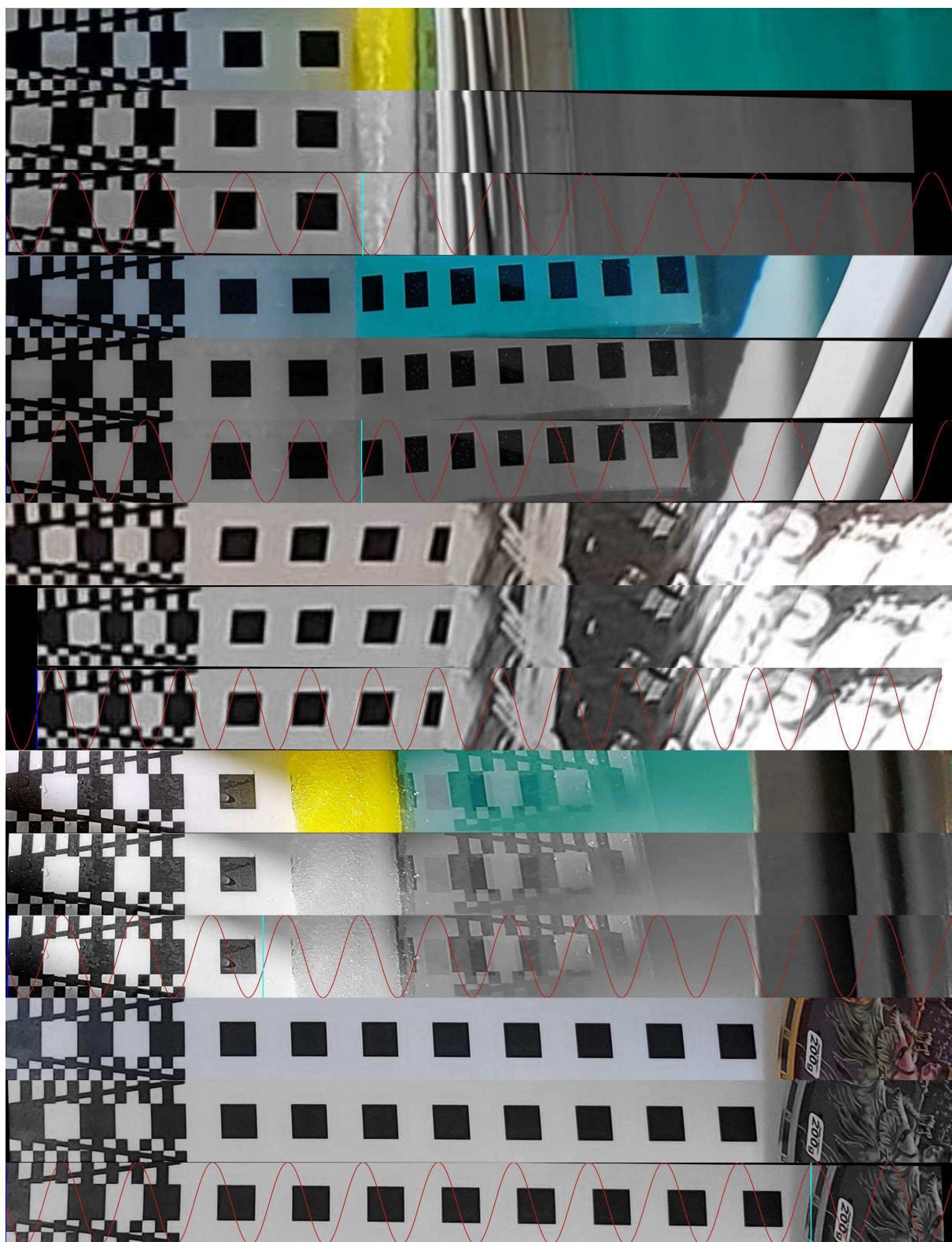
Figura 36 – Amostras e respectivos resultados 1.



Fonte: Autoria própria

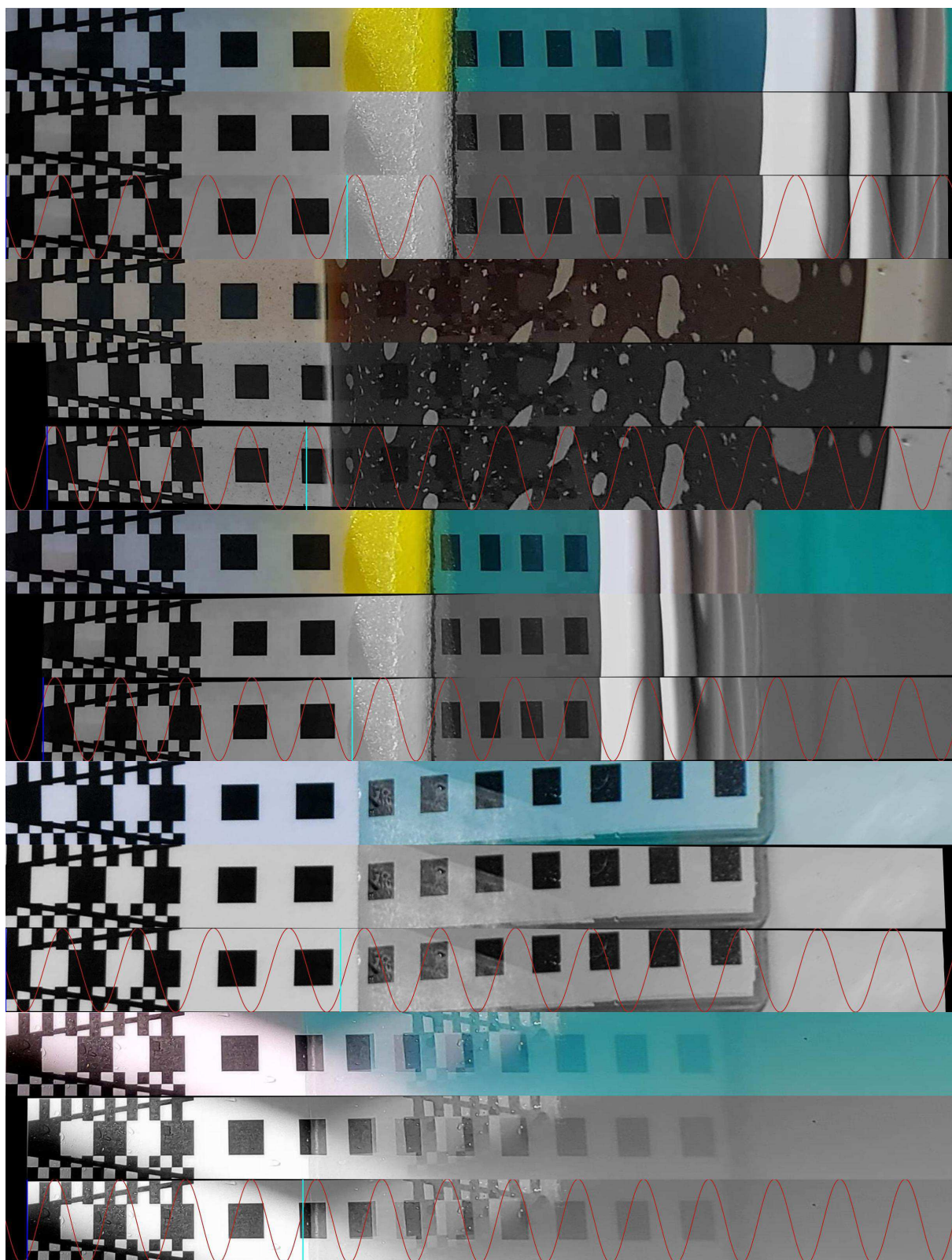


Figura 37 – Amostras e respectivos resultados 2.



Fonte: Autoria própria

Figura 38 – Amostras e respectivos resultados 3.



Fonte: Autoria própria



## 6 CONCLUSÃO

Neste trabalho foi apresentada uma alternativa sobre o trabalho de (Nagaoka, 2019) para medição do nível de corpos d'água. Comparado à obra original, observa-se maior consistência nas medições realizadas, i.e. a metodologia proposta obteve uma taxa aceitável (16 de um total de 21 amostras) de medições realizadas com sucesso sem perda significativa de precisão, i.e. com erro percentual menor que 10%, sob condições mais adversas, como ângulos não-oblíquos, iluminação solar parcial, reflexões, transparência, etc. Foi também comprovado que a régua projetada pode ser utilizada para medição sem que uma boia seja empregada, o que facilita a implantação do sistema.

Os resultados das medições realizadas com o novo processo e dados do trabalho de (Nagaoka, 2019), para o qual uma régua de 1 metro foi projetada, reforçam a hipótese que o sistema pode ser projetado e testado com dados provenientes de uma miniatura da régua.

Das medições que foram concluídas com sucesso, observa-se que ficaram com erro concentrado na região de 0 a 10%. Com uma régua de 288mm de comprimento, erros variaram de 1 a 24 milímetros comparados com os valores reais. Alguns sistemas de alerta poderiam fazer uso deste instrumento de medição para cumprirem com seu propósito, ainda que o aparato não forneça medições precisas.

### 6.1 TRABALHOS FUTUROS

Recomenda-se para próximos trabalhos a pesquisa e implementação de uma fonte de dados simulados, i.e. imagens renderizadas. As dificuldades e custos da construção de um medidor em tamanho real que se mantenha reto sob vento ou água, a dificuldade de instalação e o número absoluto de possíveis cenários em que este medidor pode ser submetido para somente então capturar imagens, o que traz outras complexidades, sugerem que alguma alternativa mais simples possa ser implementada. O mercado<sup>1</sup> já oferece soluções para isso, mas nem sempre pode-se ou quer-se arcar com este custo. Computadores preparados para jogos modernos, algo relativamente fácil de se encontrar em posse de estudantes, já oferecem o poder de processamento para lidar com este tipo de tarefa, bastaria que alguma solução de *software open source* ou para não-comercial fosse disponibilizada.

A escassez de dados limita também as abordagens utilizadas para resolução dos problemas. Sugere-se que mais dados sejam obtidos não somente para teste, mas para implementação de aprendizado de máquina para melhor encontrar a interface com a água ou até mesmo a região de interesse.

---

<sup>1</sup><https://unity.com/products/unity-simulation>



## Referências

- Andrade F. **Sistema De Monitoração De Alarme Contra Alagamentos**. BR1020160234255A2. Disponível em: <<https://gru.inpi.gov.br/pePl/servlet/PatenteServletController?Action=detail&CodPedido=1396628&SearchParameter=BR%2010%202016%20023425-5%20%20%20%20%20%20>>. Citado na página 1.
- Bradski G. The OpenCV Library. **Dr. Dobb's Journal of Software Tools**, 2000. Citado na página 31.
- Cooley, J. W.; Tukey, J. W. An algorithm for the machine calculation of complex Fourier series. **Mathematics of Computation**, v. 19, n. 90, p. 297–297, maio 1965. ISSN 0025-5718. Disponível em: <<http://www.ams.org/jourcgi/jour-getitem?pii=S0025-5718-1965-0178586-1>>. Citado na página 20.
- Feature Matching — OpenCV Python Tutorials. **Feature Matching — OpenCV Python Tutorials**. 2021. Disponível em: <[https://opencv-python-tutroals.readthedocs.io/en/latest/py\\_tutorials/py\\_feature2d/py\\_matcher/py\\_matcher.html](https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_matcher/py_matcher.html)>. Citado na página 22.
- Fischler, M. A.; Bolles, R. C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. **Communications of the ACM**, v. 24, n. 6, p. 381–395, jun. 1981. ISSN 0001-0782, 1557-7317. Disponível em: <<https://dl.acm.org/doi/10.1145/358669.358692>>. Citado 2 vezes nas páginas 9 e 19.
- Harris, C. R. et al. Array programming with NumPy. **Nature**, v. 585, p. 357–362, 2020. Citado na página 31.
- Hunter, J. D. Matplotlib: A 2d graphics environment. **Computing in Science & Engineering**, IEEE COMPUTER SOC, v. 9, n. 3, p. 90–95, 2007. Citado na página 31.
- Hübner, K. L. Medição automática de nível de água baseado em imagens: Caso do rio Itajaí-Açú. Trabalho de Conclusão de Curso - UDESC, 2013. Citado 6 vezes nas páginas 1, 4, 6, 7, 8 e 9.
- Kennedy, J.; Eberhart, R. Particle swarm optimization. In: **Proceedings of ICNN'95 - International Conference on Neural Networks**. [S.l.: s.n.], 1995. v. 4, p. 1942–1948 vol.4. Citado na página 26.
- Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. **International Journal of Computer Vision**, v. 60, n. 2, p. 91–110, nov. 2004. ISSN 0920-5691. Disponível em: <<http://link.springer.com/10.1023/B:VISI.0000029664.99615.94>>. Citado 3 vezes nas páginas 9, 16 e 17.
- Nagaoka, T. Medição De Nível De Rios Baseada Em Análise De Vídeo. Trabalho de Conclusão de Curso - UTFPR, 2019. Citado 16 vezes nas páginas 1, 2, 4, 9, 10, 11, 12, 13, 14, 21, 23, 26, 35, 39, 40 e 45.
- Sigifredo Nino, Andre Beaulieu, Jean-Louis Dumont, Ciero Bart Di, Ronald Fisher, Nathalie Marcotte, Sylvain Nadeau, Trevor Nowlan, Andre Raymond e Richard Steele. **Lidar-based level measurement**. 2008. Patente n° WO2008024910A3. Disponível em: <<https://patents.google.com/patent/WO2008024910A3/en>>. Citado na página 1.

OpenCV: Feature Matching + Homography to find Objects. **OpenCV: Feature Matching + Homography to find Objects**. 2021. Disponível em: <[https://docs.opencv.org/master/d1/de0/tutorial\\_py\\_feature\\_homography.html](https://docs.opencv.org/master/d1/de0/tutorial_py_feature_homography.html)>. Citado na página 23.

OpenCV: Geometric Image Transformations. **OpenCV: Geometric Image Transformations**. 2021. Disponível em: <[https://docs.opencv.org/4.4.0/da/d54/group\\_\\_imgproc\\_\\_transform.html#gaf73673a7e8e18ec6963e3774e6a94b87](https://docs.opencv.org/4.4.0/da/d54/group__imgproc__transform.html#gaf73673a7e8e18ec6963e3774e6a94b87)>. Citado na página 23.

Otsu, N. A threshold selection method from gray-level histograms. **IEEE Transactions on Systems, Man, and Cybernetics**, v. 9, n. 1, p. 62–66, 1979. Citado na página 4.

Powell, M. J. D. An efficient method for finding the minimum of a function of several variables without calculating derivatives. **The Computer Journal**, v. 7, n. 2, p. 155–162, 01 1964. ISSN 0010-4620. Disponível em: <<https://doi.org/10.1093/comjnl/7.2.155>>. Citado na página 26.

Ribeiro E. P., Braga S. M. e Fernandes C. V. S. **Transdutor Para Medição De Nível De Água Em Rios E Reservatórios**. 2015. Patente nº BR1020150319576A2. Disponível em: <<https://gru.inpi.gov.br/pePI/servlet/PatenteServletController?Action=detail&CodPedido=1363031&SearchParameter=BR%2010%202015%20031957-6%20%20%20%20%20%20&Resumo=&Titulo>>. Citado na página 1.

Smith, J. O. **Mathematics of the Discrete Fourier Transform (DFT)**. <http://www.w3k.org/books/>: W3K Publishing, 2007. ISBN 978-0-9745607-4-8. Citado 3 vezes nas páginas 20, 24 e 25.

Szymanski C., Sousa N. e Kesting F. L. **Aperfeiçoamento Introduzido Em Medidor De Nível De Fluidos Tipo Borbulhador**. 2016. Patente nº BR2020160161415U2. Disponível em: <<https://gru.inpi.gov.br/pePI/servlet/PatenteServletController?Action=detail&CodPedido=1387464&SearchParameter=BR%2020%202016%20016141-5%20%20%20%20%20%20>>. Citado na página 1.

Virtanen, P. et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. **Nature Methods**, v. 17, p. 261–272, 2020. Citado na página 31.

Wikipédia. **Scale-invariant feature transform**. 2020. Page Version ID: 177813849. Disponível em: <[https://fr.wikipedia.org/w/index.php?title=Scale-invariant\\_feature\\_transform&oldid=177813849](https://fr.wikipedia.org/w/index.php?title=Scale-invariant_feature_transform&oldid=177813849)>. Citado na página 17.

Xu, Z. et al. Water Level Estimation Based on Image of Staff Gauge in Smart City. In: **2018 iee smartworld, ubiquitous intelligence & computing, advanced & trusted computing, scalable computing & communications, cloud & big data computing, internet of people and smart city innovation (smartworld/scalcom/uic/atc/cbdcom/iop/sci)**. Guangzhou, China: IEEE, 2018. p. 1341–1345. ISBN 978-1-5386-9380-3. Disponível em: <<https://ieeexplore.ieee.org/document/8560211/>>. Citado 4 vezes nas páginas 1, 4, 5 e 6.