

Universidade Federal de Santa Catarina
Centro de Blumenau
Departamento de Engenharia de
Controle e Automação e Computação



Felipe Alfredo Nack

Sistema de Medição do Volume de Toras de Madeira Utilizando
Visão Computacional

Blumenau

2021

Felipe Alfredo Nack

Sistema de Medição do Volume de Toras de Madeira Utilizando Visão Computacional

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como parte dos requisitos necessários para a obtenção do Título de Engenheiro de Controle e Automação.

Orientador: Prof. Dr. Ebrahim Samer El Youssef

Coorientador: Prof. Dr. Maurício Edgar Stivanello

Universidade Federal de Santa Catarina

Centro de Blumenau

Departamento de Engenharia de
Controle e Automação e Computação

Blumenau

2021

Felipe Alfredo Nack

Sistema de Medição do Volume de Toras de Madeira Utilizando Visão Computacional

Trabalho de Conclusão de Curso apresentado à Universidade Federal de Santa Catarina como requisito parcial para a obtenção do título de Engenheiro de Controle e Automação.

Comissão Examinadora

Prof. Dr. Ebrahim Samer El Youssef
Universidade Federal de Santa Catarina
Orientador

Prof. Dr. Carlos Roberto Moratelli
Universidade Federal de Santa Catarina

Prof. Dr. Marcos Vinicius Matsuo
Universidade Federal de Santa Catarina

Blumenau, 30 de setembro de 2021

Dedico este trabalho a todos aqueles que, de alguma forma,
auxiliaram na continuidade da ciência.

Agradecimentos

Agradeço aos meus pais pelo apoio e suporte. Aos meus avós por toda a ajuda e preocupação. A Adrielle por todo o companheirismo, paciência e incentivo. Aos meus orientadores, Prof. Ebrahim Samer El Youssef e Prof. Maurício Edgar Stivanello, por todo o acompanhamento, lições, paciência e orientação, fundamentais nesse projeto. Ao Prof. Mauri Ferrandin, pelas oportunidades. A todos os servidores da UFSC campus Blumenau pelos ótimos serviços prestados. A R5 Controle e Automação pelos equipamentos, amostras e tempo disponibilizados para o desenvolvimento do projeto.

"Not all those who wander are lost."
(John Ronald Reuel Tolkien)

Resumo

Neste trabalho de conclusão de curso é realizado o desenvolvimento de um sistema capaz de estimar o volume de toras de madeira em uma linha produtiva. Esta medida é calculada pelo produto entre a área da seção reta da tora de madeira e o seu comprimento. A abordagem proposta utiliza um sistema de visão computacional para estimar a área da seção reta da tora, e usa dois sensores de distância a laser para estimar seu comprimento. O conjunto formado pela câmera e por um sensor é posicionado em uma lado da tora e conectado a um servidor, enquanto o outro sensor fica do outro lado e é conectado a um microprocessador que trata os dados adquiridos e os envia ao servidor, que é responsável por todo processamento. A câmera captura a imagem da face da tora, que é submetida a uma sequência de processamentos responsável por segmentá-la do resto da imagem. A área segmentada então é utilizada em conjunto com uma análise geométrica para realizar uma conversão de *pixels* para mm^2 . Essa conversão possui uma dependência com a distância entre câmera e tora, assim a informação do sensor acoplado a câmera é relevante para essa operação. O comprimento da tora é determinado com as medidas de distância dos sensores à tora que são posicionados há uma distância conhecida entre si e de tal forma que suas linhas de ação são aproximadamente colineares com o eixo longitudinal da tora. Testes de validação da solução da proposta são conduzidos em um protótipo construído para emular o cenário real na indústria. Os resultados apurados são satisfatórios e dão alicerce para a próxima etapa de desenvolvimento de um protótipo para testes na indústria.

Palavras-Chave: 1. Visão Computacional. 2. Toras de Madeira. 3. Estimativa de

Volume.

Abstract

This work involves the development of a system capable of estimating the volume of wood logs in a production line. This measure is calculated by the product between the area of the straight section of the wooden log and its length. The proposed approach uses a computer vision system to estimate the cross-sectional area of the log, and uses two laser distance sensors to estimate its length. The set formed by the camera and a sensor is placed on one side of the log and connected to a server, while the other sensor is on the other side and is connected to a microprocessor that handles the acquired data and sends them to the server, which is responsible for all processing. The camera captures the image of the log face, which is subjected to a sequence of processing responsible for segmenting it from the rest of the image. The segmented area is then used in conjunction with a geometric analysis to perform a conversion from *pixels* to mm^2 . This conversion has a dependence on the distance between camera and log, so the information from the sensor attached to the camera is relevant for this operation. The length of the log is determined by measuring the distance of the sensors to the log that are positioned at a known distance from each other and in such a way that their lines of action are approximately collinear with the longitudinal axis of the log. Validation tests of the proposed solution are conducted on a prototype built to emulate the real-world scenario in the industry. The results obtained are satisfactory and are the foundation for the next stage of developing a prototype for testing in the industry.

Keywords: 1. Computer Vision. 2. Wood Log. 3. Volume Estimation.

Lista de figuras

Figura 1 – Consumo aparente de painéis de madeira no Brasil entre os anos de 1961 e 2016.	14
Figura 2 – Torno laminador vendido pela empresa Hunter Engenharia.	15
Figura 3 – Lâminas de madeira.	17
Figura 4 – Máquina utilizada na produção de lâminas de madeira.	18
Figura 5 – Arredondador de toras.	18
Figura 6 – Laminador de toras.	19
Figura 7 – Tora de madeira após o processo de arredondamento.	21
Figura 8 – Condições para que a proposta apresente bons resultados. O autor. . .	21
Figura 9 – Câmera <i>pinhole</i>	24
Figura 10 – Simplificação do processo para adquirir uma imagem em uma câmera digital.	24
Figura 11 – Lentes finas ideais.	25
Figura 12 – Efeitos de distorção em lentes radiais.	26
Figura 13 – Padrão RGB Bayer.	27
Figura 14 – Espectros de cor.	28
Figura 15 – Operação morfológica de processamento de imagem.	29
Figura 16 – Exemplo de operações de erosão e dilatação.	30
Figura 17 – Processo de segmentação clássico realizado em imagens coloridas. . . .	30
Figura 18 – Classificação de elementos em uma imagem binária.	31
Figura 19 – Seleção de <i>blob</i> cuja classe seja $C = 1$	31
Figura 20 – Indústria de alimentos utilizando-se das tecnologias de visão computacional e <i>AI</i>	32
Figura 21 – <i>Smart Detection</i> treinado para encontrar pestes em um campo.	33
Figura 22 – Rebites presentes em latas metálicas.	34
Figura 23 – Exemplo de sensor de distância a laser coaxial.	34
Figura 24 – Exemplo de uma rede <i>MQTT</i>	36
Figura 25 – Barramento I ² C com um mestre e n escravos. O valor n é definido pelo tamanho do endereço, geralmente sendo 7, 10 ou 16 bits.	37
Figura 26 – Esboço de uma tora de madeira.	38
Figura 27 – Arquitetura geral do sistema.	39
Figura 28 – Sequência de aquisição e processamento.	40
Figura 29 – Cálculo de comprimento com dois sensores.	41
Figura 30 – Cálculo de comprimento com um sensor.	42

Figura 31 – Conjunto de imagens utilizadas para os testes de segmentação e cálculo de área.	43
Figura 32 – Câmera utilizada em conjunto com a Raspberry.	44
Figura 33 – Imagem de teste C.	45
Figura 34 – <i>ROI</i> da Figura 33.	45
Figura 35 – Segmentação da Figura 34, por cor, sem tolerância.	46
Figura 36 – Subespaço de cor HSV	47
Figura 37 – Segmentação de cor utilizando um subespaço de cores.	47
Figura 38 – Resultados das operações de erosão e dilatação aplicadas na Figura 37.	48
Figura 39 – Resultados das operações de fechamento e abertura, feitas em sequência, aplicadas na Figura 37.	48
Figura 40 – Resultado final da segmentação.	50
Figura 41 – Resultados da operação de segmentação proposta.	50
Figura 42 – Modelo utilizado para elaborar as equações de conversão de unidade.	51
Figura 43 – Esp32-CAM.	53
Figura 44 – Raspberry Pi 3 Model B.	55
Figura 45 – Vista geral da bancada de testes.	58
Figura 46 – Vista do posicionamento da (A) câmera e do (B) sensor de distância a laser.	59
Figura 47 – Exemplos de imagens coletadas para os testes finais.	61
Figura 48 – Comparação entre os valores medidos e reais de $d1$	62
Figura 49 – Comparação entre os valores medidos e reais de A	62
Figura 50 – Comparação entre os valores medidos e reais de V	63
Figura 51 – Curva de tempo de processamento do sistema.	63
Figura 52 – Erro associado as medidas de $d1$, A e V feitas pelo sistema.	64

Lista de tabelas

Tabela 1 – Características da câmera.	43
Tabela 2 – Parâmetros da segmentação. O autor.	46
Tabela 3 – Características do sensor VL53L0X.	54
Tabela 4 – Medidas da bancada de testes.	58
Tabela 5 – Resultados preliminares para 4 objetos de teste.	60
Tabela 6 – Medidas da bancada de testes.	60

Lista de acrônimos

UFSC	<i>Universidade Federal de Santa Catarina</i>
IDE	<i>Integrated Development Environment</i>
GPIO	<i>General Purpose Input/Output</i>
ARM	<i>Acorn RISC Machine</i>
USB	<i>Universal Serial Bus</i>
MQTT	<i>Message Queuing Telemetry Transport</i>
EPL	<i>Eclipse Public License</i>
EDL	<i>Eclipse Distribution License</i>
ROI	<i>Region of Interest</i>
CVS	<i>Computer Vision System</i>
CCD	<i>Charge-coupled Device</i>
CMOS	<i>Complementary Metal Oxide on Silicon</i>
AI	<i>Artificial Intelligence</i>
ToF	<i>Time of Flight</i>
DSP	<i>Digital Signal Processor</i>
M2M	<i>Machine to Machine</i>
IoT	<i>Internet of Things</i>
GUI	<i>Graphical User Interface</i>

Sumário

1	INTRODUÇÃO	14
1.1	Objetivo geral	16
1.1.1	Objetivos específicos	16
1.2	Organização do documento	16
2	DESCRIÇÃO DO PROBLEMA	17
2.1	Processo produtivo da indústria de laminação de madeiras . .	18
2.2	Caracterização do problema	19
3	REVISÃO DE LITERATURA	23
3.1	Sistemas de visão	23
3.1.1	Aquisição, processamento e análise de imagens	23
3.1.1.1	Câmera	23
3.1.1.2	Lente	25
3.1.1.3	Sensor	25
3.1.1.4	Imagem e espaço de cor	26
3.1.1.5	Operações morfológicas	28
3.1.1.6	Classificação, representação e descrição	29
3.1.2	Utilização de sistemas de visão na indústria	32
3.2	Outras ferramentas utilizadas	34
3.2.1	Sensor de distância a laser	34
3.2.2	Protocolos de comunicação	35
3.2.2.1	Protocolo MQTT	35
3.2.2.2	Protocolo I2C	36
4	SISTEMA DE MEDIÇÃO DO VOLUME DE TORAS	38
4.1	Descrição geral da solução desenvolvida	38
4.1.1	Estimativa de comprimento	40
4.1.2	Estimativa de área	42
4.1.2.1	Segmentação	42
4.1.2.2	Conversão de unidade de área	49
4.1.3	Estrutura Computacional	52
4.1.3.1	Microcontrolador	52
4.1.3.2	Servidor e processamento	54
5	RESULTADOS	57

5.1	Metodologia de testes e protótipo	57
5.2	Validação do sistema	60
5.3	Considerações	60
6	CONSIDERAÇÕES FINAIS	65
6.1	Trabalhos futuros	65
	REFERÊNCIAS BIBLIOGRÁFICAS	67

1 Introdução

As primeiras civilizações precisavam encontrar abrigos de forma que pudessem se proteger tanto de animais selvagens quanto do extremo frio [BOCALON et al., 2014]. Inicialmente esses abrigos eram cavernas espalhadas ao redor das terras, contudo, ao passo que o ser humano precisava locomover-se, técnicas de construção precisariam ser criadas. Segundo [BOCALON et al., 2014] e [(US), 1987] a madeira exerce grande impacto nesse sentido, permitindo a fácil construção de abrigos, ferramentas e meios de transporte devido a sua fácil obtenção e manipulação quando comparada a outras matérias primas como os metais ou pedras.

A manipulação da madeira, por ser um tipo de tecnologia milenar, sofreu diversas modificações no seu uso ao longo da evolução humana, indo desde abrigo até fonte de energia. Uma dessas modificações é o uso de painéis de madeira para a confecção dos mais diversos bens. Conforme o exposto em [MATTOS; GONÇALVES; LACERDA, 2008] o setor de painéis de madeira no Brasil apresenta grande relevância, produzindo em 2005 mais de 7.7 milhões de m^3 , valor bastante expressivo resultado de um crescimento de 9.5% ao ano desde 1995. Um dos fatores que impulsionou o setor foi a busca por alternativas para a madeira maciça, essa que dispõe de um preço bastante elevado. Na Figura 1 é apresentado o consumo aparente de alguns tipos de painéis de madeira no Brasil entre os anos de 1961 e 2016 [SANQUETTA et al., 2019].

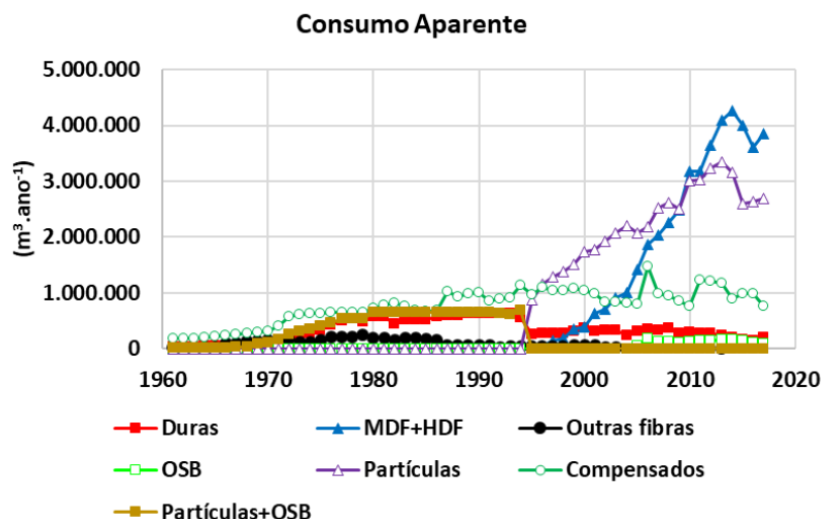


Figura 1 – Consumo aparente de painéis de madeira no Brasil entre os anos de 1961 e 2016. Fonte: [SANQUETTA et al., 2019].

Os painéis de madeira são geralmente construídos a partir de lâminas de madeira ou a partir da madeira em outro estágio de desagregação [MATTOS; GONÇALVES; LACERDA, 2008]. As lâminas de madeira por sua vez são obtidas através de um processo

de laminação que, no geral, cria muito desperdício e sofre de grandes variações a depender da matéria prima utilizada. O processo acontece geralmente em uma máquina conhecida como torno laminador, ilustrada na Figura 2.

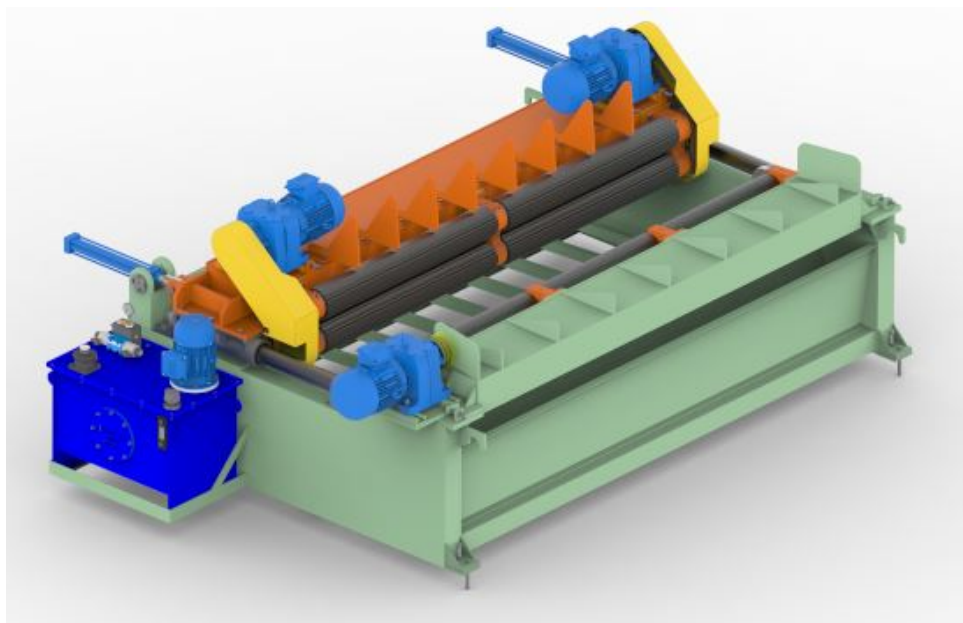


Figura 2 – Torno laminador vendido pela empresa Hunter Engenharia. Fonte: [HUNTER... , 2021].

Esses tornos são fornecidos por diversas fabricantes, naturalmente, algumas dessas máquinas conseguem realizar o processo com maior eficiência em relação a outras. Ainda que exista uma diferença no desempenho das máquinas, a eficiência do processo também depende da qualidade da matéria prima. Especificamente, como medida de eficiência costuma-se utilizar a razão entre a quantidade de matéria prima utilizada e a quantidade de produto fabricado, ambos em m^3 .

Atualmente, na empresa utilizada como base para esse trabalho, a medição do volume das toras é realizada manualmente por um operador, o que pode implicar em erros e em um tempo de execução elevado, os quais podem ser reduzidos com uma automatização do processo de medição. Além disso, um levantamento sobre soluções disponíveis no mercado mostrou que não há uma solução dedicada especificamente para o caso. Foi encontrada uma solução de visão computacional em que se deve adquirir um pacote de *hardware* e de serviço de desenvolvimento de software para a aplicação desejada, que é conhecida como Sherlock e pode ser vista em [SHERLOCK, 2021] com detalhes. A desvantagem desta solução é o grande custo envolvido. Esta demanda foi levantada pela empresa parceira R5 Controle e Automação, que forneceu os detalhes da problemática, equipamentos para o desenvolvimento e também as amostras para prototipagem. Os requisitos estão alinhados com o que se espera de um produto comercial e servem como base para o desenvolvimento do projeto.

Assim, neste trabalho é desenvolvida uma abordagem para estimar o volume de matéria prima que chega até o torno laminador, através de técnicas que envolvem tanto visão computacional quanto o emprego de sensores de distância. A solução aborda diversos pontos cruciais para que o sistema consiga estimar o volume com melhor precisão e tempo de resposta que a solução manual, e com custo atrativo do ponto de vista da aplicação do mercado brasileiro. Em particular, a proposta abordada nesse trabalho de conclusão de curso é completamente não invasiva ao funcionamento da máquina, ao passo que consegue retornar uma estimativa de matéria prima gasta no processo de laminação.

1.1 Objetivo geral

Desenvolver um sistema capaz de estimar o volume de toras de madeira antes do processo de laminação, que atenda os requisitos de exatidão e tempo de processamento definidos pela aplicação.

1.1.1 Objetivos específicos

Os objetivos específicos do projeto abrangem:

- Calcular o volume de toras de madeira cuja face tenha qualquer formato que respeite uma superfície convexa;
- Construir uma bancada de testes para validação do sistema desenvolvido;
- Ter boa exatidão e bom tempo de resposta, conforme discutido nos capítulos seguintes.

1.2 Organização do documento

O remanescente do trabalho está dividido em 5 capítulos. O Capítulo 2 contém a descrição do processo produtivo da indústria de laminação, assim como dos detalhes do problema, desafios e requisitos. Logo na sequência o Capítulo 3 aborda uma revisão de literatura que abrange a maioria dos conhecimentos teóricos necessários para o desenvolvimento da solução. Ainda no Capítulo 3 são expostos alguns exemplos da utilização de visão computacional na indústria. O Capítulo 4 trata dos métodos empregados e a estrutura geral do sistema proposto como solução. Tal capítulo é dividido nas três principais partes no qual o problema é decomposto. Já o Capítulo 5 aborda uma síntese dos resultados obtidos nos testes e uma discussão sobre a validade da solução proposta. Por fim, no Capítulo 6 estão apresentadas as conclusões finais do trabalho como um todo e possíveis trabalhos futuros baseados neste documento.

2 Descrição do Problema

É natural que a indústria como um todo busque, cada vez mais, otimizar seus processos produtivos, consumindo o mínimo de matéria-prima para gerar o máximo de produto. Uma consequência direta disso é o investimento no desenvolvimento de processos de automação de máquinas e os avanços no sensoriamento.

Para lidar com este objetivo é necessário mensurar a produtividade, e a forma de fazer isso varia com a natureza do processo produtivo. Neste trabalho, o tema é o processo de produção de lâminas de madeira em uma indústria madeireira. Para este caso o interesse é mensurar o rendimento de produto final, lâminas de madeira, produzidos com uma quantidade de matéria prima, toras de madeira pré-regularizadas. O volume em metros cúbicos de lâminas produzidos é bem conhecido, o volume das toras é estimado em um processo manual que é lento e contém erros que podem ser importantes. As lâminas de madeira mencionadas podem ser vistas na Figura 3 de forma ilustrativa. Esse tipo de produto tem a principal serventia de atuar como um revestimento, geralmente em móveis e painéis de madeira.



Figura 3 – Lâminas de madeira feitas através do processo de laminação de toras. Fonte: o autor.

Para contribuir com tal propósito, este trabalho é dedicado a desenvolver uma abordagem por meio visão computacional e sensoriamento de distancia a laser para estimar o volume das toras instantes antes que sejam processadas no produto final, de forma a agregar mais celeridade e precisão ao processo de medição do volume de toras de madeira.

2.1 Processo produtivo da indústria de laminação de madeiras

O processo de laminação de madeira é executado por uma máquina conforme é ilustrado na Figura 4. Essa máquina recebe a tora pré arredondada e produz lâminas de madeira.

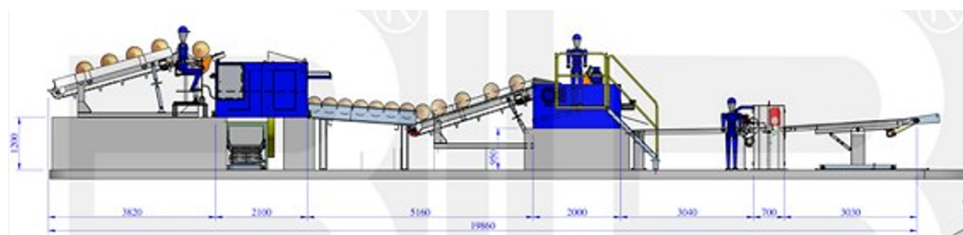


Figura 4 – Máquina utilizada na produção de lâminas de madeira. Fonte: adaptado de [BENECKE..., 2021].

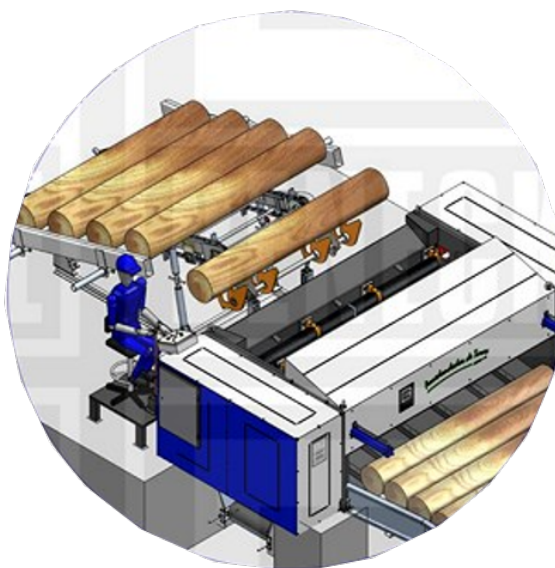


Figura 5 – Arredondador de toras. Fonte: adaptado de [BENECKE..., 2021].

O processamento das toras possui duas tarefas: o arredondamento de tora e a laminação de tora. A primeira tarefa (Figura 5), consiste em receber toras de madeira na sua forma natural e retirar suas cascas, tornando-as mais regulares e homogêneas. Já a segunda tarefa (Figura 6), consiste em processar as toras de madeira arredondadas em uma grande lâmina por meio de uma lâmina de metal cortante cujo ângulo de ataque determina a espessura da lâmina.

Entre os setores onde ocorre o arredondamento e a laminação encontram-se as toras de madeira já regularizadas e homogêneas. Estas são consideradas como a matéria prima para produção das lâminas de madeira e objetos cuja quantidade deseja-se determinar. É

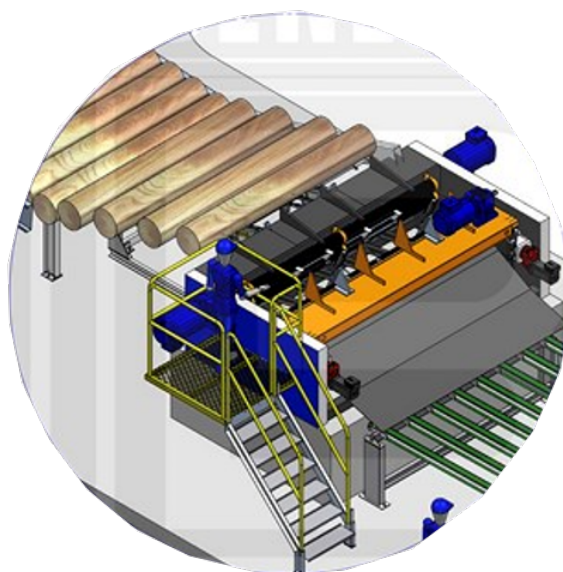


Figura 6 – Laminador de toras. Máquina com a mesma função daquela apresentada na Figura 2. Fonte: adaptado de [BENECKE... , 2021].

importante ressaltar que entre os processos de arredondamento e laminação existe uma esteira responsável por transportar as toras de madeira. A laminação de uma tora de madeira pode levar até 10 segundos dependendo do tamanho da matéria prima, e esse tempo faz com que a esteira de transporte fique parada enquanto uma tora está sendo laminada. Assim, o processo de medição das toras pode ser realizado em paralelo ao processo produtivo, durante os momentos nos quais as toras estão paradas, sem efetuar nenhum tipo de interferência.

2.2 Caracterização do problema

O problema abordado pode ser descrito objetivamente como: Determinar o volume de matéria prima pré-processada, ou seja, volume de cada tora arredondada antes de serem transformadas em lâminas, com precisão e velocidade maiores do que aquelas já atingidas com o processo manual.

Conforme já mencionado, no levantamento bibliográfico e de mercado não foi possível encontrar uma solução estabelecida para o problema. Foi encontrado um produto que conta com arquitetura de hardware de visão computacional para qual se pode contratar um serviço de desenvolvimento software para aplicação desejada. Assim, entende-se que de fato não é uma solução estabelecida para a demanda e pode ter um custo maior do que o desenvolvimento de uma solução dedicada.

Para cumprir o objetivo do trabalho, deve-se resolver as questões enumeradas a seguir, cujas soluções em alguns casos são triviais e outros demandam análise e desenvolvimento mais aprofundado, e são tratadas nos próximos capítulos.

1. Qual grandeza física é mais adequada para mensurar a quantidade de matéria prima, toras arredondas?

Há duas opções: peso ou volume. Devido a quantidade de produto final ser medida em volume (m^3), a melhor opção é medir o volume das toras em metros cúbicos também.

2. Há algum período e espaço dentro do processo em que se possa realizar medições para determinar o volume de cada tora, sem gerar interferências?

Conforme visto no fim da Seção 2.1, existe um momento em que a esteira transportadora fica parada, permitindo que um sistema de medição seja acoplado, atuando em paralelo a linha produtiva sem oferecer nenhum tipo de interferência.

3. Como estimar com precisão e agilidade o volume das toras arredondas, cujo exemplo de imagem podem ser visto na Figura 7? Dado que elas:

- não têm um formato geométrico ordinário;
- são diferentes umas das outras tanto em termos da suas seções e no comprimento;
- possuem variação de cores;
- estão inseridas em um ambiente com vapor, acrescentando ruído a imagem.

O desenvolvimento da solução dessa questão é o tema principal deste trabalho e é detalhado nos próximos capítulos. Assume-se como condições delimitantes para resolver este problema:

- a) A área de seção reta da tora (A) deve permanecer constante ao longo de seu comprimento, característica presente nos cilindroides;
- b) O plano cuja seção reta da tora se encontra deve ser perpendicular ao eixo longitudinal da tora (z), conforme indica a Figura 8.
- c) A tora pode estar livre na esteira sem garantia de que esteja sempre alinhada a uma borda.

No geral as duas primeiras condições são razoáveis e reais na indústria, sendo que a primeira condição é garantida através do processo de arredondamento e a segunda condição é garantida pelo fornecedor da matéria-prima. A terceira condição é generalista, pois pode haver casos em que as toras estejam alinhadas em um batente em uma das extremidades da esteira e a solução proposta pode ser simplificada, conforme será discutido no Capítulo 4.



Figura 7 – Tora de madeira após o processo de arredondamento. Fonte: o autor.

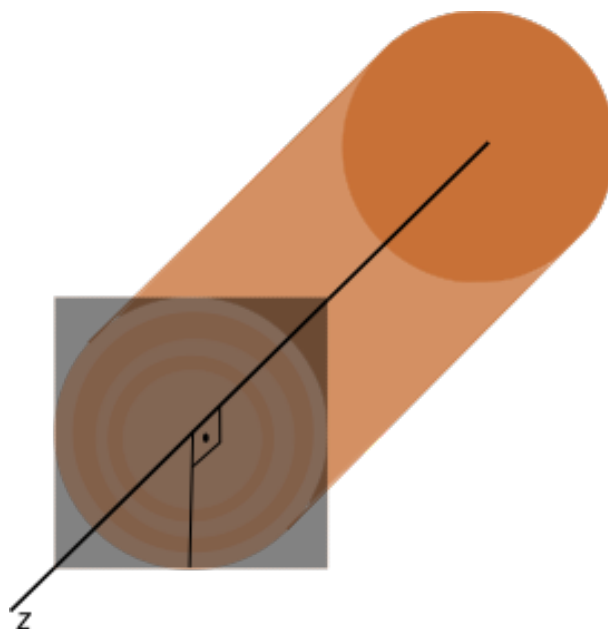


Figura 8 – Condições para que a proposta apresente bons resultados. O autor.

Por fim, a estimativa deve apresentar razoável exatidão, garantindo que seja viável estimar o volume das toras de madeira através do sistema proposto ao invés de utilizar medição manual.

Observando tais condições, temos que os principais requisitos funcionais da problemá-

tica são definidos como:

- Medir o volume da tora de madeira;
- Processar as informações em sincronia com a linha produtiva;
- A estimativa de volume de cada tora deve ocorrer em até 5 segundos, que é o tempo médio da laminação de toras pequenas;
- Possuir um erro médio menor que 10%;

3 Revisão de literatura

Esse capítulo tem como principal objetivo introduzir alguns conceitos básicos, de forma que facilite o entendimento do documento como um todo, abordando tópicos como: visão computacional, sensoriamento e protocolos de comunicação.

3.1 Sistemas de visão

Nessa seção são abordados alguns conceitos referentes aos sistemas de visão computacional, incluindo temas como câmera, lente, sensor, espaço de cor, operações morfológicas e outros. Para cada tema existe uma breve descrição em conjunto de figuras ilustrativas.

3.1.1 Aquisição, processamento e análise de imagens

A análise e processamento de imagens é um tópico recorrente em pesquisas que envolvem visão computacional [SIEGWART; NOURBAKHS; SCARAMUZZA, 2011]. No geral, o campo de visão computacional está interessado em transformar imagens ou extrair características delas, utilizadas posteriormente como informações para algum fim. Naturalmente, o campo de visão computacional faz a análise de imagens que são muitas vezes obtidas por câmeras, essas que podem possuir as mais variadas arquiteturas e funcionalidades [CORKE; KHATIB, 2011][SZELISKI, 2010][CIPOLLA; BATTIATO; FARINELLA, 2010].

3.1.1.1 Câmera

Um dos modelos de câmera mais simples é conhecido como câmera *pinhole* (câmera escura) [FORSYTH; PONCE, 2011][NUSSENZVEIG, 2014][WÖHLER, 2012]. A câmera *pinhole* consiste em um pequeno buraco feito em uma caixa escura, permitindo que a luz atravesse a abertura e exiba uma imagem inversa em seu interior, em acordo com a Figura 9.

Naturalmente câmeras modernas não possuem um funcionamento tão simples e tendem a seguir o esquema discutido por [SZELISKI, 2010] na Figura 10. Note que o processo de *demosaiçing* consiste em estimar as cores faltantes adquiridas pelo filtro de Bayer, discutido posteriormente na Figura 13.

Comumente os *CVSs* (*Computer Vision Systems*) tratam diversas abordagens para que no fim seja possível extrair alguma característica ou ainda modificar a imagem, como visto em [MARTYNENKO, 2006], que elabora uma sequência bastante utilizada na área de segmentação. Em [MARTYNENKO, 2006] ainda é visto que, normalmente, um dos

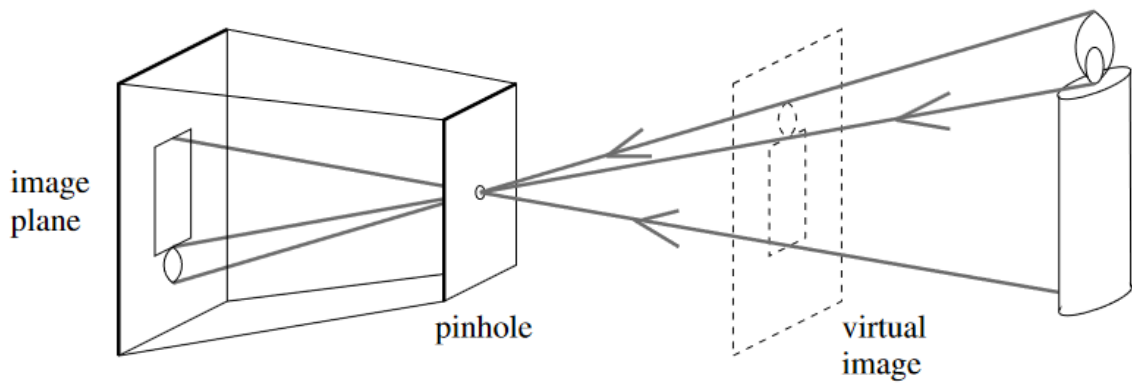


Figura 9 – Câmera *pinhole*. Fonte: [FORSYTH; PONCE, 2011].

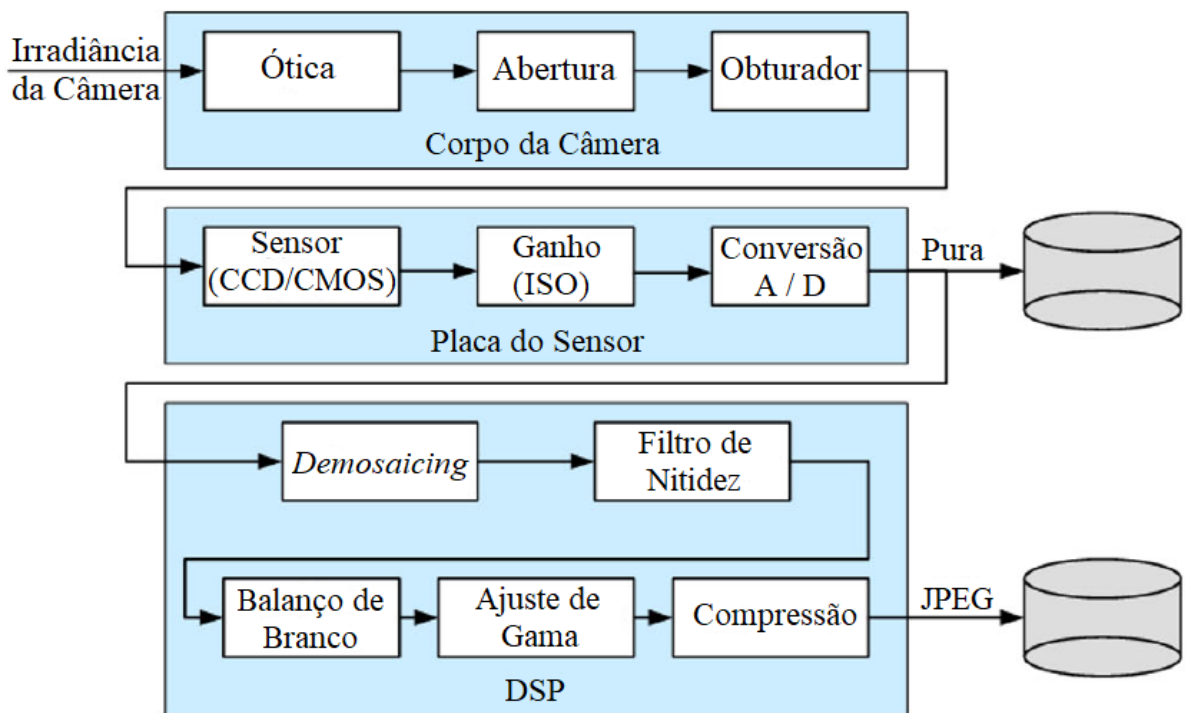


Figura 10 – Simplificação do processo para adquirir uma imagem em uma câmera digital. Fonte: adaptado de [SZELISKI, 2010].

primeiros passos para uma análise de imagem é a segmentação. Contudo, ainda que a segmentação seja responsável por grande parte da solução do problema, tratada com mais detalhes posteriormente, outra parte do problema é resolvida através do conhecimento e análise dos componentes físicos básicos de um *CVS*. Destacam-se entre os componentes de uma câmera a lente e o sensor, que desempenham papel importante durante o processo de análise de imagens e conversão de unidades [SZELISKI, 2010]. Conforme afirmação contida em [FORSYTH; PONCE, 2011] a importância do uso de lentes em câmeras reais, se deve ao fato destas contornarem os problemas existentes no modelo *pinhole*: imagens borradas e fraca luminosidade.

3.1.1.2 Lente

Ao contrário da câmera *pinhole* proposta na Figura 9 as câmeras convencionais tendem a possuir uma lente biconvexa [NUSSENZVEIG, 2014], que permite que os raios sejam focados de tal forma que uma imagem melhor é gerada, se comparada com a câmera *pinhole*, comportamento esse indicado na Figura 11 onde é encontrada uma lente idealmente fina. Como esperado, os raios possuem comportamento similar, apesar de não igual, a aquele encontrado na câmera *pinhole*, informação essa que é importante e é utilizada na Seção 4.1.2.1 para criar a solução do problema deste trabalho.

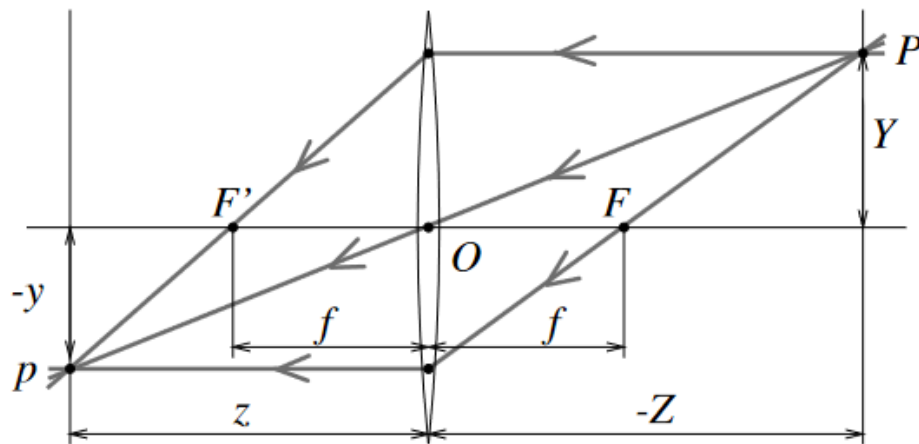


Figura 11 – Lentes finas ideais. Os raios que passam por O não sofrem o fenômeno de refração, ao passo que os raios paralelos ao eixo óptico (z) são redirecionados para o ponto focal F' . Fonte: [FORSYTH; PONCE, 2011].

Perceba que lentes idealmente finas não representam da melhor forma sistemas reais, que são melhor representados por lentes que possuem uma certa espessura. Lentes que não são idealmente finas trazem consigo o fenômeno de distorção, abordado com detalhes em [SZELISKI, 2010], que pode comprometer as realizadas na imagem capturada. Os principais efeitos de distorção são conhecidos como *pincushion* e *barrel*, ilustrados na Figura 12 em respectiva ordem da esquerda para a direita. Os efeitos de distorção podem ser compensados através de um modelo quártico (na maioria das vezes) como apresentado em [SZELISKI, 2010] e [WÖHLER, 2012], contudo, os efeitos de distorção se mostraram de pouca relevância aparente nesse trabalho e por isso são ignorados.

3.1.1.3 Sensor

O sensor de uma câmera funciona de forma análoga a retina do olho [CORKE; KHATIB, 2011] e desempenha um papel tão fundamental quanto a lente na qualidade da imagem adquirida, além de ditar as transformações geométricas conforme visto na Seção 4.1.2.1. De acordo com [SZELISKI, 2010], a luz incidente em um sensor é coletada

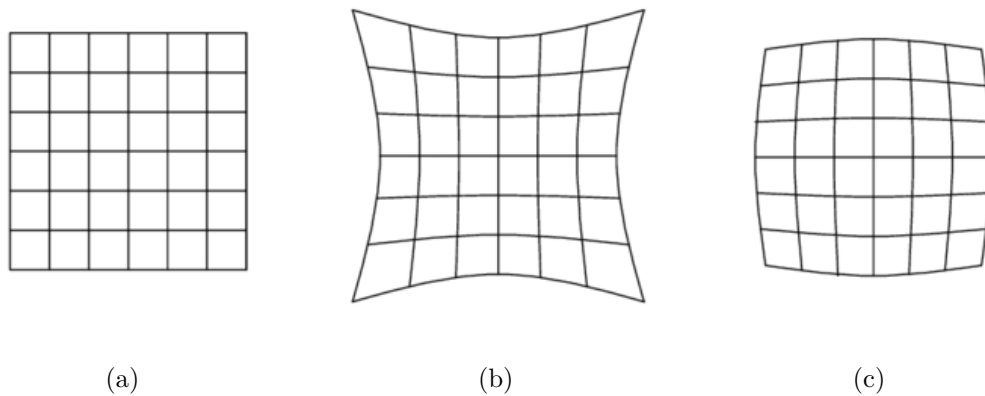


Figura 12 – Efeitos de distorção em lentes radiais: (a) padrão com ausência de distorção, (b) padrão com distorção do tipo *pincushion*, (c) padrão com distorção do tipo *barrel*. Perceba que o efeito de distorção tem um comportamento bastante periférico. Fonte: [PAREDES, 2016]

por uma área ativa de sensoriamento durante um período de tempo chamado de exposição, logo antes de ser enviada para um sistema de ganho. Em câmeras modernas é comum que os sensores sejam do tipo *CCD* (*Charge-coupled Device*) ou do tipo *CMOS* (*Complementary Metal Oxide on Silicon*). Algumas características que podem ser necessárias conhecer para o desenvolvimento de uma solução de visão computacional são:

- Tempo de exposição: tempo no qual é feita a leitura, controlando a quantidade de luz que chega ao sensor;
- Passo de amostragem: espaçamento físico entre as células de um sensor;
- Fator de preenchimento: fração de área teórica de sensoriamento;
- Ganho analógico: antes da conversão analógico-digital existe um ganho para a informação lida pelo sensor.

É bastante comum também que os sensores apresentem o formato conhecido como padrão RGB Bayer (Figura 13). Entre as especificações de um sensor são encontradas diversas características, como aquelas apresentadas posteriormente na Tabela 1, tais quais ditam diretamente o comportamento de uma análise de imagem. Características como tamanho de *pixel* e resolução do sensor são geralmente encontradas em *datasheets* de câmeras industriais e servem como fator decisivo na escolha da câmera.

3.1.1.4 Imagem e espaço de cor

No geral, imagens obtidas por câmeras convencionais (coloridas) são representadas por três matrizes, todas elas possuindo o mesmo tamanho $h \times w$, sendo w a largura em *pixels* da imagem e h a altura, também em *pixels*, da imagem. Uma fotografia feita

G	R	G	R	rGb	Rgb	rGb	Rgb
B	G	B	G	rgB	rGb	rgB	rGb
G	R	G	R	rGb	Rgb	rGb	Rgb
B	G	B	G	rgB	rGb	rgB	rGb

(a) *Layout* do filtro de cor. (b) Valores dos *pixels* interpolados, onde valores desconhecidos são representados por letras minúsculas. A reconstrução desses valores é feita através de um processo chamado *demosaicing*.

Figura 13 – Padrão RGB Bayer. Fonte: [SZELISKI, 2010].

na resolução conhecida como *Full HD*, por exemplo, é representada por três matrizes de tamanho 1080×1920 . Cada uma das três matrizes é responsável por representar um parâmetro utilizado na reconstrução da imagem, sendo que esses parâmetros no geral estão dispostos conforme o espectro **RGB** (*Red Green Blue*) (Figura 14a), conferindo a cada matriz uma das cores desse espectro. Apesar do espectro **RGB** ser um dos mais comuns para representar imagens, existem outros que conferem uma série de facilidades e permitem executar um trabalho mais intuitivo, como por exemplo o espectro **HSV**.

Quando a representação é feita através do espectro **RGB**, no geral, cada elemento das matrizes é um número inteiro de 8 *bits*, ou seja, seu valor pode variar entre 0 e 255, permitindo 256^3 combinações de cores. Em contrapartida, o espectro de cores **HSV** trabalha com valores diferentes, tendo na sua primeira matriz – H (matiz) – valores que podem variar entre 0° e 360° , na sua segunda matriz – S (saturação) – valores que podem variar entre 0 e 100% e na sua última matriz – V (valor) – também valores que podem variar entre 0 e 100%. De forma bastante simplificada é possível definir que o valor da matriz H controla qual cor é selecionada no cilindro do espectro **HSV** (note que H é um valor de ângulo), ao passo que o valor da matriz S define a pureza da cor e o valor da matriz V define o brilho da cor, ou seja, para valores baixos de saturação (S) a cor é mais acinzentada e para valores baixos de valor (V) a cor é mais escura. Perceba então que se o objetivo for selecionar uma cor e suas derivadas, o espectro **HSV** é muito mais intuitivo, afinal, basta escolher uma cor e definir uma tolerância aos seus arredores, seja para ângulo (H) ou para saturação (S) e valor (V). Na Seção 4.1.2.1 é apresentada uma abordagem para a segmentação da área da seção reta da tora de madeira através de uma manipulação do espectro **HSV**.

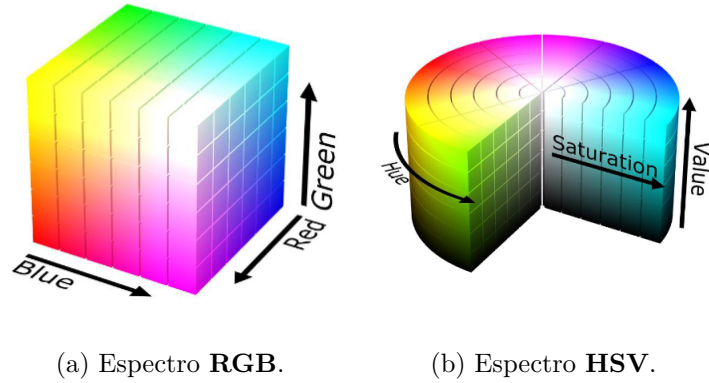


Figura 14 – Espectros de cor. Fonte: [POPOV; OSTAREK; TENISON, 2018].

3.1.1.5 Operações morfológicas

A operação morfológica é uma classe de operações que tem como objetivo utilizar um elemento estruturante, chamado por algumas literaturas de *kernel*, em regiões da imagem com o objetivo de executar algum tipo de modificação em seu formato. Uma definição formal da operação morfológica é dada por [CORKE; KHATIB, 2011] através da Equação (3.1).

$$\mathbf{O}[u, v] = f(\mathbf{I}[u + i, v + j]), \forall (i, j) \in S, \forall (u, v) \in \mathbf{I} \quad (3.1)$$

Sendo S o elemento estruturante, cujo formato geralmente é dado por uma região quadrada de tamanho $w \times w$, onde w é um número ímpar descrito por $w = 2h + 1$, sendo que $h \in \mathbb{Z}^+$, isto é, h é $\frac{w-1}{2}$. \mathbf{O} é a imagem de saída, \mathbf{I} é a imagem de entrada e o par (u, v) define uma coordenada na imagem. Note que a Figura 15 indica que a operação morfológica é definida apenas no elemento selecionado (vermelho) que fica contido dentro do elemento de estrutura.

A operação morfológica de erosão tem como premissa erodir elementos em uma imagem binária, isto é, diminuí-los de acordo com o elemento estruturante. O elemento estruturante é posicionado de forma que esteja centralizado em cada par de coordenadas (u, v) na imagem \mathbf{I} , ao passo que o *pixel* equivalente de coordenada (u, v) na imagem \mathbf{O} seja resultado dessa operação, onde $\mathbf{O}(u, v) = 0$ caso exista um elemento 0 dentro do elemento estruturante e $\mathbf{O}(u, v) = 1$ caso contrário. A operação de dilatação por outro lado tem como principal premissa dilatar os elementos de acordo com o *kernel* escolhido, funcionando de forma similar ao processo de erosão, contudo, segue o funcionamento de $\mathbf{O}(u, v) = 1$ caso exista um elemento 1 dentro do elemento estruturante e $\mathbf{O}(u, v) = 0$ caso contrário. Na Figura 16 encontram-se imagens no formato binário, onde cinza significa o valor 0 e branco significa o valor 1, permitindo visualizar a aplicação das operações de erosão e dilatação com *kernels* de tamanhos diferentes.

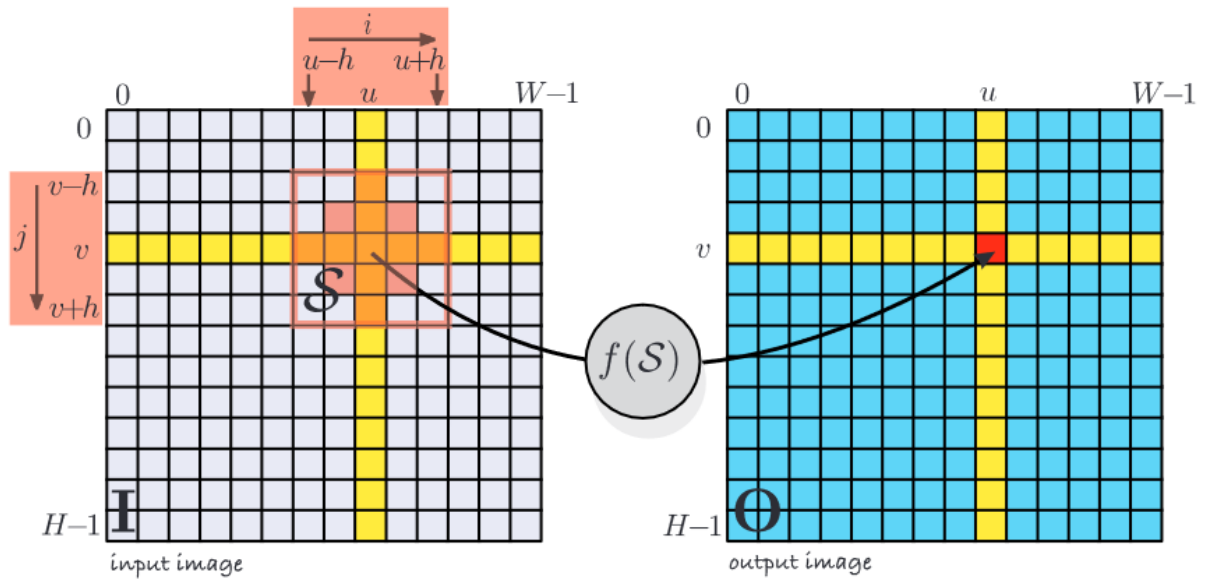


Figura 15 – Operação morfológica de processamento de imagem. Fonte: [CORKE; KHATIB, 2011].

3.1.1.6 Classificação, representação e descrição

A literatura trata comumente elementos conectados por *pixels* adjacentes em uma imagem binária como *blobs*, possibilitando sua classificação de forma que, ao percorrer a imagem, para cada conjunto de pontos conectados sem interrupção é atribuído um número de identificação. Formalmente a classe de um conjunto de *pixels* (*blob*) é dada por um número inteiro $\in \{0 \dots C - 1\}$ onde C é a quantidade de classes presentes na imagem [CORKE; KHATIB, 2011].

Perceba que se $C = 2$ em um problema de classificação, estamos tratando geralmente de uma classificação binária. Geralmente classificações binárias são utilizadas em processos de limiarização, comumente aplicado em imagens em escala de cinza. A Equação (3.2) representa o processo de classificação binária [CORKE; KHATIB, 2011], onde I é a imagem e t é um valor de limiar escolhido.

$$c[u, v] = \begin{cases} 0 & \mathbf{I}[u, v] < t \\ 1 & \mathbf{I}[u, v] \geq t \end{cases} \quad \forall (u, v) \in \mathbf{I} \quad (3.2)$$

Nesse ponto já é possível realizar uma simples segmentação. Apesar do fluxograma proposto na Figura 17 não representar a sequência de tratamentos utilizados na Seção 4.1.2.1, na segmentação proposta nesse trabalho, vemos nele uma sequência clássica de processamento para se obter uma segmentação de imagem. O processamento clássico envolve transformar uma figura colorida em uma figura de escala de cinza, permitindo que uma limiarização simples seja aplicada, segmentando assim uma porção da figura.

Uma análise bastante comum e muito relevante aqui é a análise de componentes conectados ou *blobs*. É razoável supor que em uma imagem binária seja de interesse que

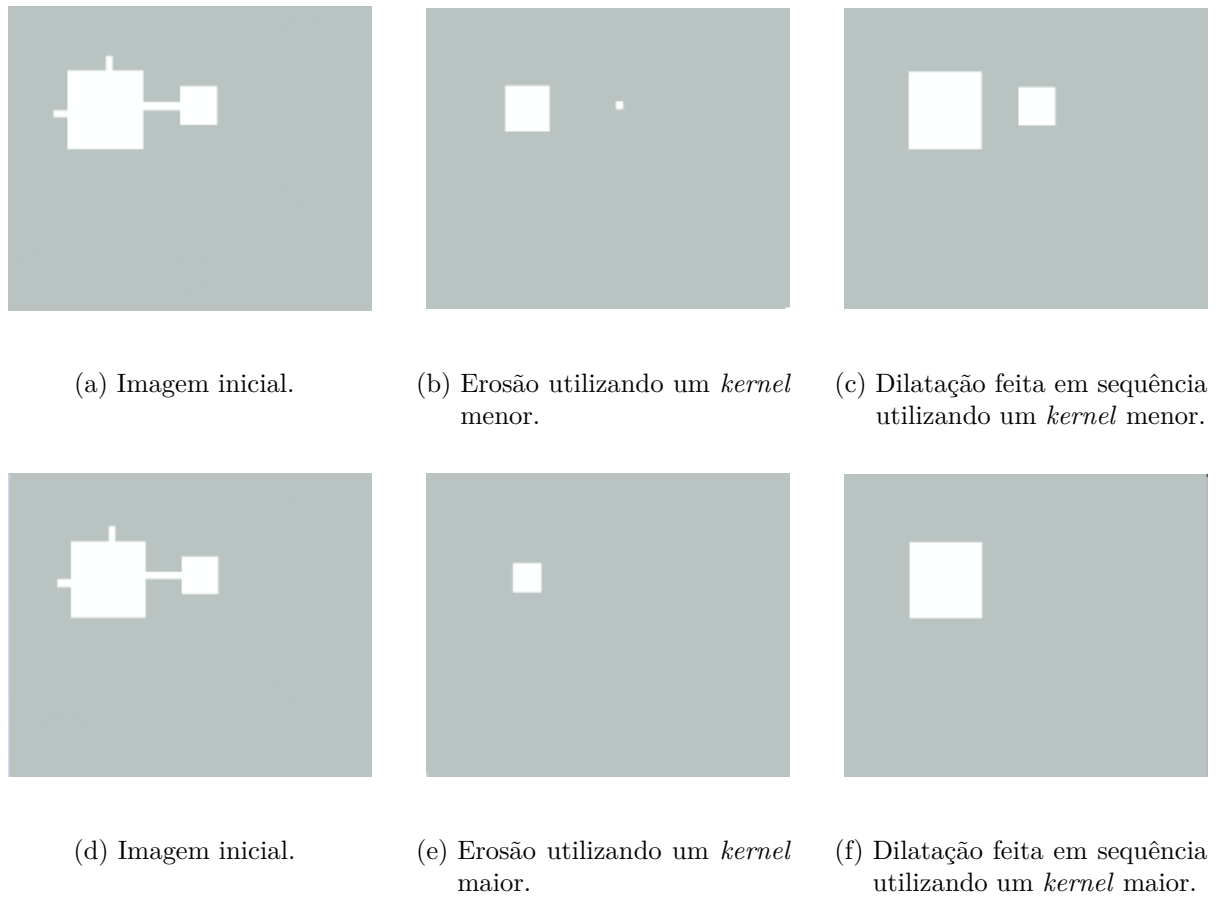


Figura 16 – Exemplo de operações de erosão e dilatação. Note que, caso o *kernel* não tenha um tamanho exagerado, um efeito de desconexão entre os objetos pode ser visto. Fonte: adaptado de [CORKE; KHATIB, 2011].

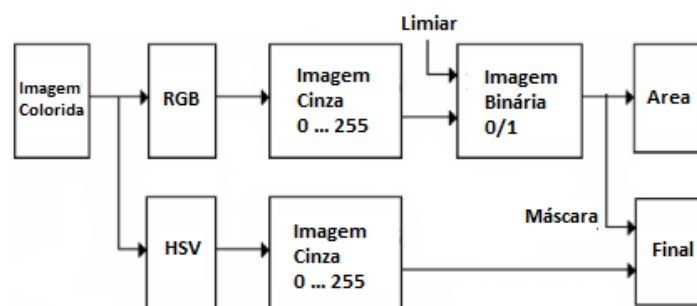
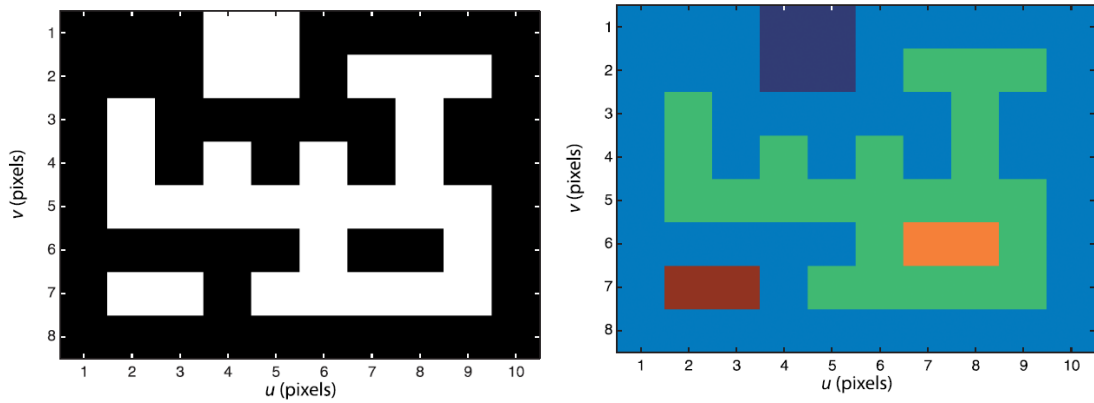


Figura 17 – Processo de segmentação clássica realizado em imagens coloridas. Fonte: adaptado de [MARTYNYENKO, 2006].

cada um dos *blobs* tenha sua própria classe. Perceba na Figura 18 que cada uma das cores representa um diferente valor de C , atribuindo para cada um dos elementos conectados sua própria classe. Determinar quais *pixels* pertencem a um mesmo elemento é um processo que consiste em verificar os arredores de um único *pixel* e atribuir o mesmo valor de classe para eles, criando um novo rótulo de classe quando um novo elemento é encontrado.

Vale ressaltar que é comum não considerar *pixels* de valor 0 como possíveis elementos, ao contrário do apresentado na imagem.



(a) Imagem binária.

(b) Classificação dos elementos.

Figura 18 – Classificação de elementos em uma imagem binária. Cada cor está relacionada com um diferente valor de C . Fonte: [CORKE; KHATIB, 2011].

A atribuição de um valor de classe para cada um dos *blobs* permite que seja possível selecionar os elementos individualmente. Na Figura 18b o maior elemento (verde) possui um valor de classe $C = 1$, permitindo a aplicação da Equação (3.3) de tal forma que apenas um *blob* permanece na imagem resultante, conforme ilustrado na Figura 19.

$$c[u, v] = \begin{cases} 0 & \mathbf{I}[u, v] \neq 1 \\ 1 & \mathbf{I}[u, v] = 1 \end{cases} \quad (3.3)$$

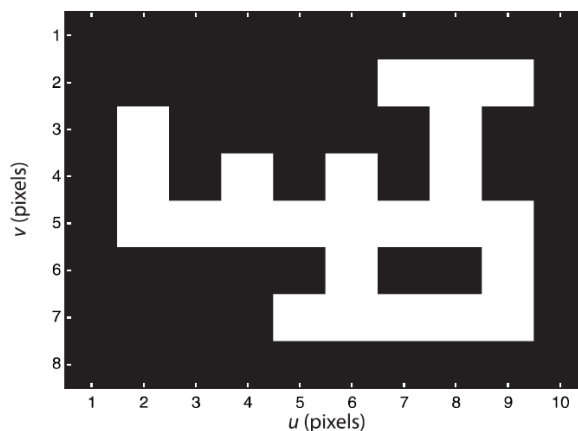


Figura 19 – Seleção de *blob* cuja classe seja $C = 1$. Fonte: [CORKE; KHATIB, 2011].

A descrição é responsável por apontar atributos aos *blobs* em uma imagem [FORSYTH; PONCE, 2011]. Dentre os atributos possíveis de se apontar os momentos se sobressaem,

esses que permitem por exemplo a estimativa de área de um *blob*. A área de um *blob* pode ser calculada através da análise de seus momentos, que são definidos pela Equação (3.4).

$$m_{pq} = \sum_{(u,v) \in \mathbf{I}} u^p v^q \mathbf{I}[u, v] \quad (3.4)$$

onde p e q são as variáveis que controlam a ordem do momento. Em específico o momento definido por $p = q = 0$ representa exatamente a área de um *blob* arbitrário, isto é, aplicando a Equação (3.4) para todos os conjuntos de elementos conectados (*blobs*) encontra-se um valor de área, em *pixels*, para cada um dos elementos na imagem

3.1.2 Utilização de sistemas de visão na indústria

A indústria cada vez mais busca por inspeções de qualidade capazes de ocorrer em alta velocidade e com elevada precisão [VILLALBA-DIEZ et al., 2019], contudo, existe o desejo de que esse tipo de inspeção não seja realizada por pessoas, mas sim através de sensoriamento. Em [AGIN, 1980] e [AMAT ALÍCIA CASALS (AUTH.), 1992] fica claro que o uso de um sistema de visão computacional para inspeção e montagem na indústria não é algo novo, tampouco desconhecido, possuindo diversos tipos de aplicações. As aplicações envolvendo visão computacional na indústria podem ser dos mais variados tipos, como em [KAKANI et al., 2020] que mostra recentes utilizações de *CVSs* em conjunto com técnicas de aprendizado de máquina para realizar inspeção de colheitas, controle de qualidade, embalagem e ordenação (Figura 20).



Figura 20 – Indústria de alimentos utilizando-se das tecnologias de visão computacional e *AI*. Fonte: adaptado de [KAKANI et al., 2020].

Ainda em [KAKANI et al., 2020] há um exemplo de sensoriamento para detecção de pestes, conforme visto na (Figura 21), expondo a importância da evolução do campo de

visão computacional para a agricultura. No geral esse tipo de solução de visão computacional é implementada em conjunto com técnicas de aprendizado de máquina, parecido com o exposto na Figura 20. Conforme os bancos de dados vão sendo criados e populados, podemos esperar pelo surgimento de diversos modelos preparados para realizar os mais variados tipos de detecções na indústria através do uso de visão computacional.

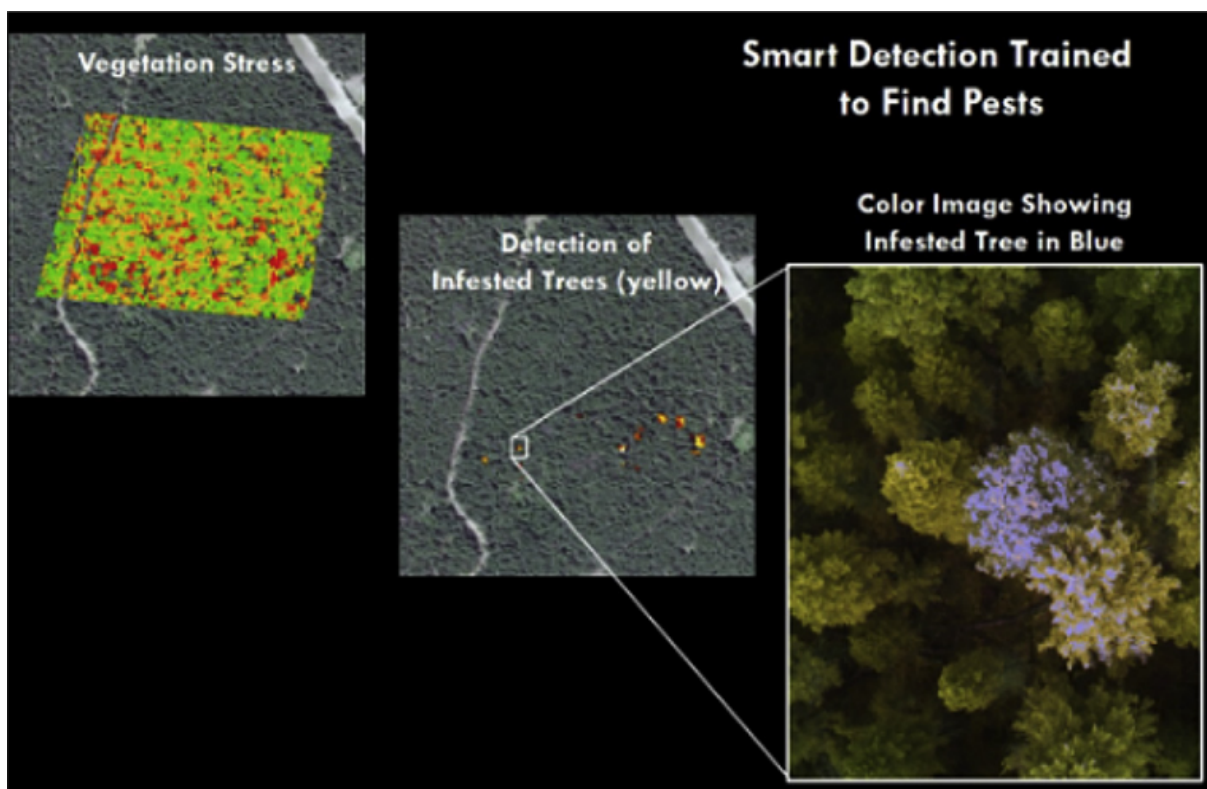


Figura 21 – *Smart Detection* treinado para encontrar pestes em um campo. Fonte: [KAKANI et al., 2020].

Outro tipo de aplicação que pode ser vista em [STIVANELLO; MARCELLINO, 2019], trata de um sistema capaz de identificar defeitos em rebites de latas de metal. Esse tipo de defeito é bastante relevante para latas que tem como destino a indústria alimentícia, podendo causar mal funcionamento na sua abertura, vazamento do produto ou até contaminação do conteúdo. Esses tipos de soluções permitem concluir que, de fato, as aplicações envolvendo visão computacional podem ser extremamente variadas e tendem a assumir a função de diversos tipos de sensores conforme as pesquisas e os desenvolvimentos na área avançam.

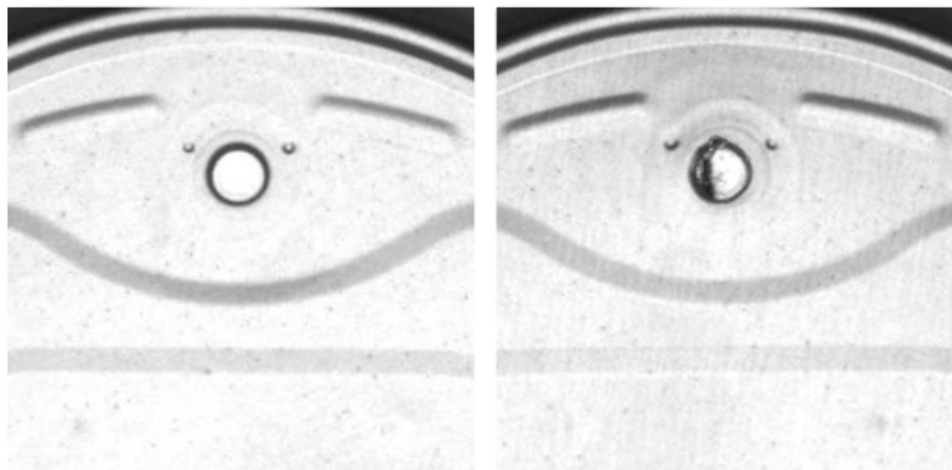


Figura 22 – Rebites presentes em latas metálicas. Percebe-se na imagem que o rebite da esquerda aparenta boa qualidade ao passo que o rebite da direita apresenta defeito. Fonte: [STIVANELLO; MARCELLINO, 2019].

3.2 Outras ferramentas utilizadas

3.2.1 Sensor de distância a laser

A invenção dos sensores de medição de distância a laser é datada da década de 60, logo após a invenção do laser rubi [MORGAN, 2020]. Apesar de existirem dois principais métodos de atuação para o sensor de distância a laser, o método utilizado no laser do projeto se aproveita da medição por *ToF* (*Time of Flight*), isto é, a medição de distância é baseado no tempo que alguma onda demora para percorrer determinada distância [KILPELÄ; PENNALA; KOSTAMOVARA, 2001]. Para entender melhor o funcionamento desse tipo de sensor considere a Figura 23.

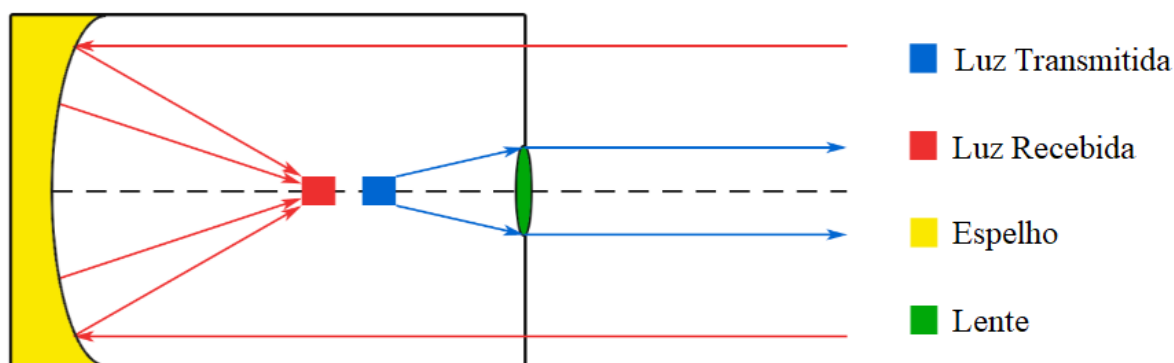


Figura 23 – Exemplo de sensor de distância a laser coaxial. Fonte: adaptado de [MORGAN, 2020].

Observe que um pulso que tem um determinado comprimento de onda do espectro eletromagnético é enviado através do transmissor e, então, o circuito interno do laser realiza a contagem de tempo até que esse mesmo pulso atinja um objeto e retorne ao

receptor. Conhecendo-se o tempo decorrido entre tais eventos e também a velocidade da luz no meio de propagação, o circuito interno do sensor calcula uma estimativa de distância entre o próprio sensor e o objeto que refletiu o pulso.

Sensores de distância a laser são amplamente conhecidos por sua precisão e exatidão. Por conta disso são excelentes para o controle de qualidade e monitoramento de processos [APPLICATIONS. . . , 2021]. No processo de produção de automóveis, por exemplo, algumas peças possuem formatos bastante incomuns, dificultando sua detecção através de sensores de contato, situação a qual favorece o uso de sensor a laser. Outra aplicação que emprega bastante esse tipo de sensor é a medida de nível em tanques cujo líquido interno é opaco, permitindo que o sensor realize a medida da altura do líquido. No contexto desse trabalho o laser é utilizado para realizar a medida de distância até uma face de tora de madeira.

3.2.2 Protocolos de comunicação

No contexto atual das redes de comunicação o protocolo é entendido como um conjunto de regras que governam o formato das mensagens que são trocadas entre dispositivos [POPOVIC, 2018]. Os protocolos no geral são estabelecidos levando em consideração o modelo ISO OSI, que cria um modelo de referência com sete camadas, sendo elas: aplicação, apresentação, sessão, transporte, rede, enlace e física. Cada uma das camadas do modelo ISO OSI tem uma função específica, que são apresentadas com mais detalhes em [KUROSE; ROSS, 2010]. Os protocolos utilizados nesse projeto, conhecidos como *MQTT* (*Message Queuing Telemetry Transport*) e *I²C* (*Inter-Integrated Circuit*), são exemplos de protocolos baseados no modelo ISO OSI.

3.2.2.1 Protocolo MQTT

O protocolo *MQTT* surgiu em meados da década de 90 através do esforço conjunto da IBM e Eurotech, projetado para ser um protocolo de comunicação *M2M* (*Machine to Machine*) que trata do envio de mensagens (camada de aplicação). Sua criação o fez se tornar um protocolo voltado para os problemas de IoT, adquirindo características como:

- Fácil transmissão de dados de um cliente para vários clientes;
- Arquitetura orientada a eventos;
- Baixo consumo de energia;
- Envio de pequenos pacotes de dados em grandes quantidades.

Em [HILLAR, 2017] apresentada ver uma extensa discussão a respeito do protocolo *MQTT* e suas aplicações, contudo, o ponto mais importante é seu modo de funciona-

mento: subscrição e publicação. O princípio de funcionamento por subscrição e publicação é muito parecido com o funcionamento de um jornal convencional, isto é, usuários inscritos em um tópico são sempre avisado de novos dados (notícias) e os usuários que fazem as publicações em um tópico têm a garantia de que todos os inscritos recebem as informações. A Figura 24 mostra um exemplo simples do funcionamento de uma rede *MQTT*. Perceba na Figura 24 que o *broker* funciona como uma espécie de servidor temporário de mensagens, mantendo a mensagem publicada durante um tempo para que os usuários possam ser avisados e façam a coleta dessa informação.

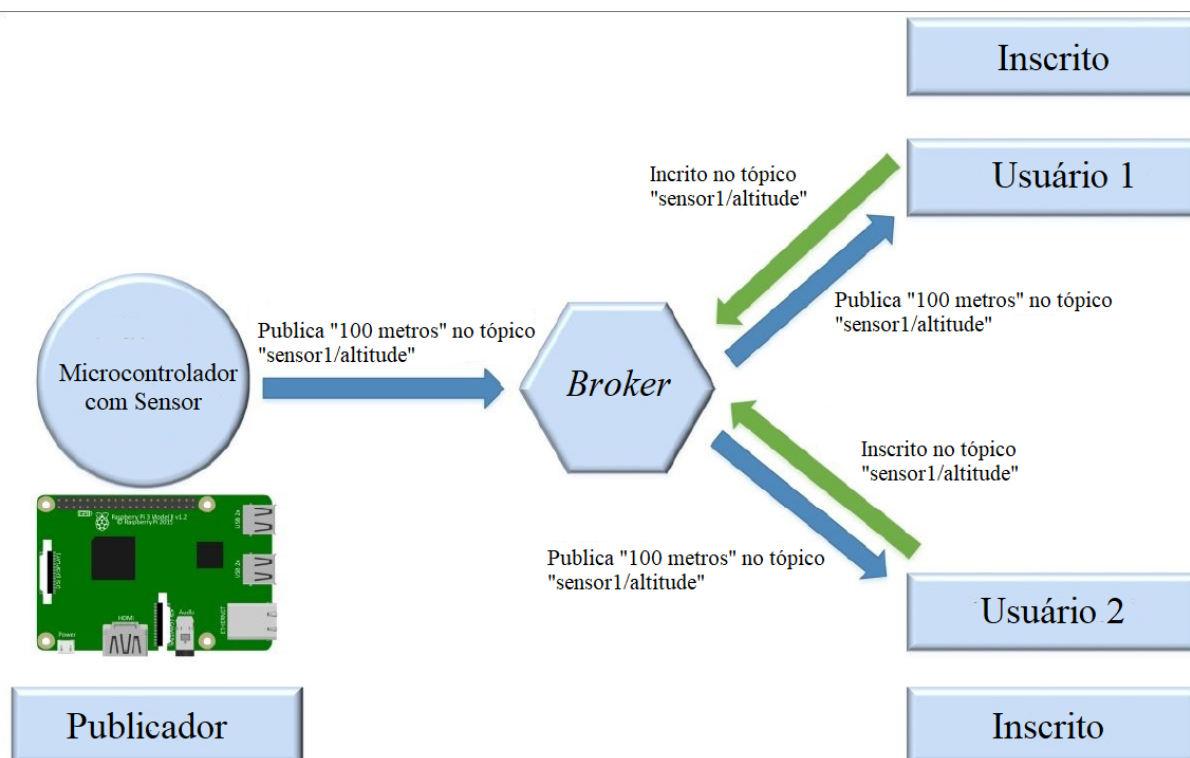


Figura 24 – Exemplo de uma rede *MQTT*. Fonte: adaptado de [HILLAR, 2017].

3.2.2.2 Protocolo I²C

O protocolo I²C segue o princípio de funcionamento de hierarquia (mestre e escravo), bastante comum em protocolos industriais como ModBUS RTU/ASCII. Nesse tipo de configuração o mestre deve coordenar o funcionamento de toda a rede, enviando e solicitando informações para cada um dos elementos conectados no barramento [MADEIRA, 2021], conforme sugere a Figura 25. Apesar de possuir algumas limitações, como a quantidade de dispositivos na rede e tamanho total do cabo, o protocolo I²C se mostra suficiente para conexões simples como aquela proposta aqui, entre microcontrolador e sensor. Perceba ainda na Figura 25 que o protocolo conta com dois fios, sendo eles o **SDL** que é responsável pela transmissão do *clock* para o barramento e o **SDA** que é responsável pela

transmissão de dados. Os elementos em um barramento I²C precisam possuir endereço único.

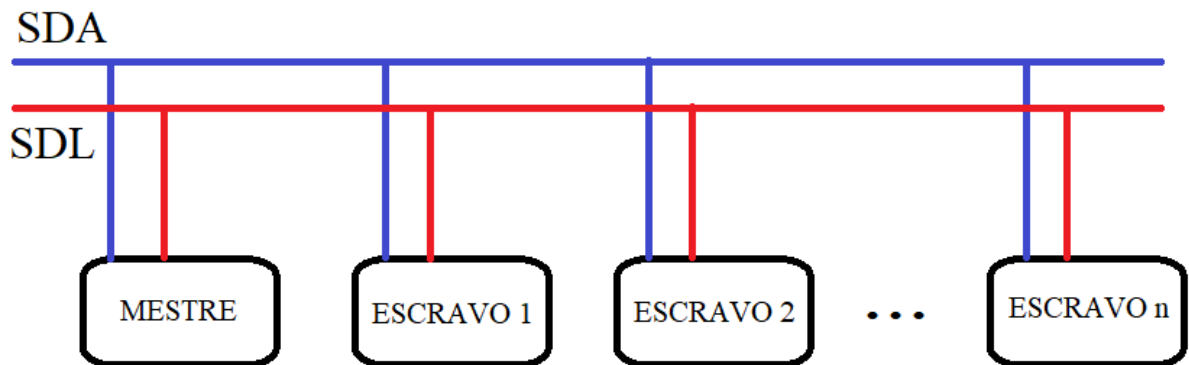


Figura 25 – Barramento I²C com um mestre e n escravos. O valor n é definido pelo tamanho do endereço, geralmente sendo 7, 10 ou 16 bits. Fonte: o autor.

O protocolo I²C é bastante utilizado em aplicações que envolvem microcontroladores, como o Arduino UNO e a ESP32. Tal protocolo permite que nessas aplicações seja possível fazer a conexão com diversos sensores e módulos complementares, para rápida prototipagem e testes de conceito. Nesse sentido ele se mostra um bom protocolo para utilizar no âmbito desse trabalho, que tem o objetivo de criar um protótipo de um sistema.

4 Sistema de medição do volume de toras

No Capítulo 2 foi descrito e delimitado o problema de estimação do volume de toras de madeira já arredondas que são a matéria prima para produção de lâminas de madeira.

Neste capítulo é proposta uma solução baseada em visão computacional e em medição de distância a laser. Considere o modelo tridimensional da tora mostrada na Figura 26, a ideia para determinar o volume V é o cálculo do produto entre as dimensões de área A e comprimento H da tora, ou seja:

$$V = A.H \quad (4.1)$$

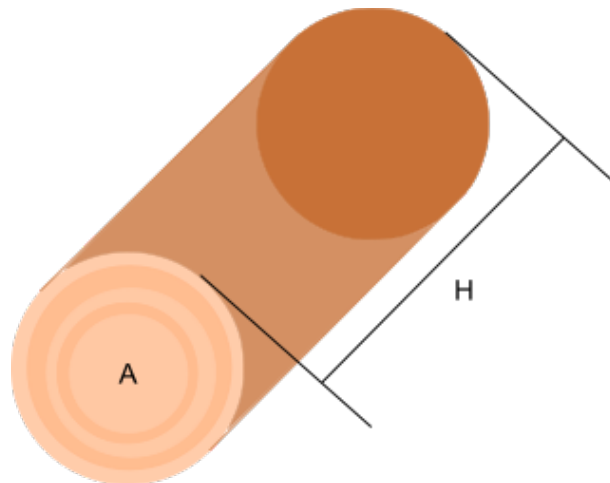


Figura 26 – Esboço de uma tora de madeira. Fonte: o autor.

A área é estimada com uma solução envolvendo visão computacional e o comprimento por meio medidas de distância com sensores laser, conforme é visto na sequência deste capítulo.

4.1 Descrição geral da solução desenvolvida

A arquitetura geral do sistema está disposta na Figura 27, e é dividida em dois módulos, o módulo 1 e o módulo 2. O módulo 1, o principal, contém os seguintes equipamentos:

- Servidor;
- Câmera;
- Sensor a laser.

Ao passo que o módulo 2 contém:

- Sensor de presença;
- Sensor a laser;
- Microcontrolador.

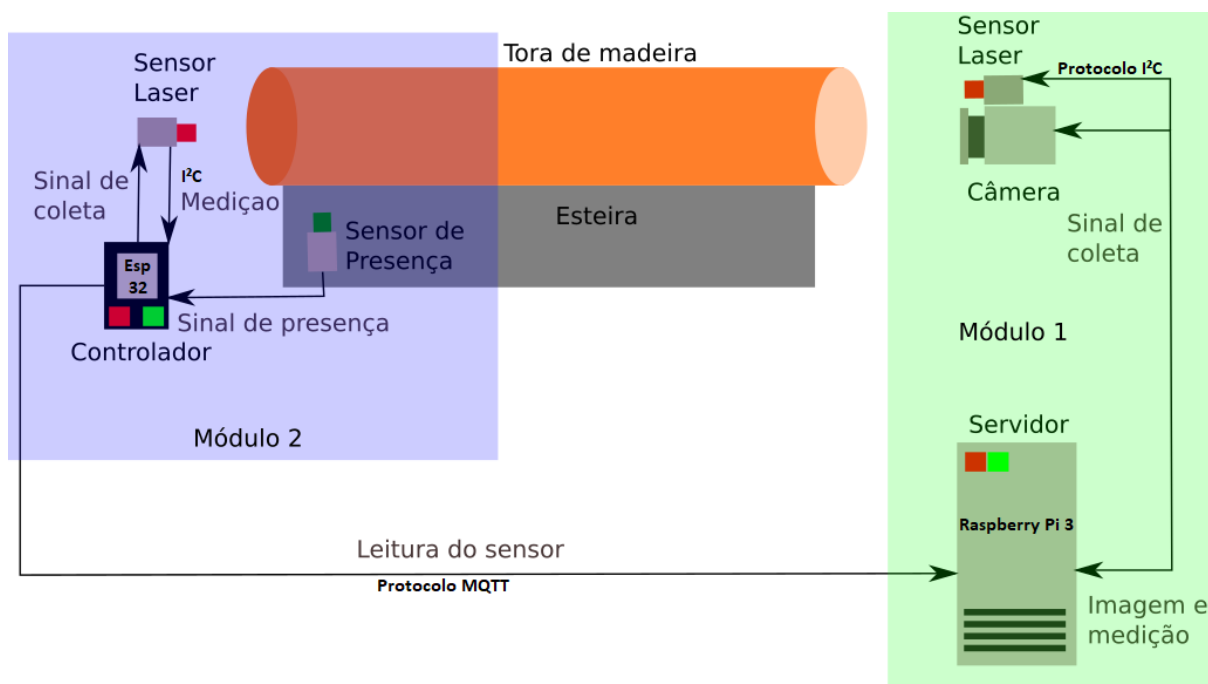


Figura 27 – Arquitetura geral do sistema. Fonte: o autor.

O funcionamento do sistema inicia pelo módulo 2 quando o seu sensor de presença detecta a existência de uma tora na posição para efetuar todo o processamento, essa informação é lida pelo microcontrolador que aciona o sensor a laser, que retorna sua distância à tora, denominada d_2 , e então o microcontrolador envia as informações para servidor presente no módulo 1.

O servidor, ao receber os dados do microcontrolador, aciona o sensor a laser e a câmera presentes no módulo 1. O sensor retorna sua distância à tora, denominada d_1 e a câmera uma foto da face da seção transversal da tora. Concluída a coleta de todas informações, um processo computacional é realizado no servidor para determinar a estimativa de volume da tora.

O processo é sintetizado no digrama mostrado na Figura 28 e, conforme já dito anteriormente o tempo de espera para o início da laminação de uma tora de madeira é da ordem de uma dezena segundos. Tal informação é importante para definição dos requisitos técnicos que cada componente do sistema deve atender. Assim, o requisito principal da solução é a exatidão.

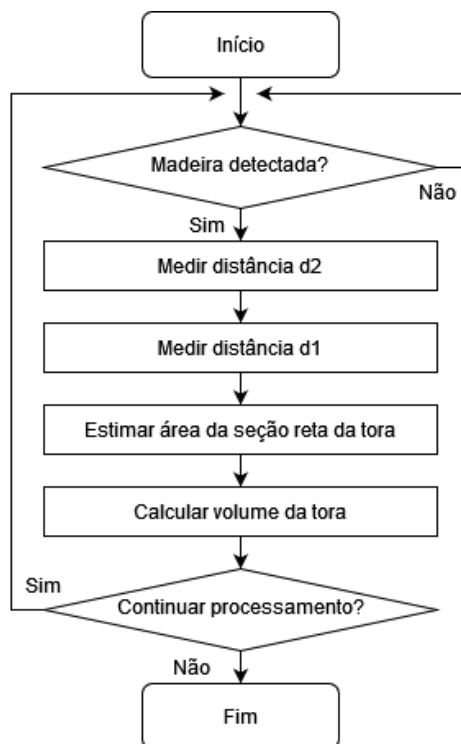


Figura 28 – Sequência de aquisição e processamento. Fonte: o autor.

Nas próximas seções são descritos com detalhes primeiro o princípio para medição do comprimento das toras, na sequência os passos de processamento de imagem para determinar a área da seção reta da tora, e por fim, a estrutura computacional.

4.1.1 Estimativa de comprimento

A estimativa de comprimento da tora de madeira, denominada na Figura 26 por H , faz o uso de sensores de distância apontados para a face da tora de madeira. Como visto a seguir, conhecendo as medidas dos sensores e a distância entre eles é possível estimar o tamanho da tora.

Na Figura 29 encontra-se um esquemático que demonstra uma forma de estimar o comprimento de um objeto, utilizando um par de sensores de distância a laser. Note que a Figura 29 indica que é possível medir o comprimento da tora de madeira caso sejam conhecidas as medidas dos sensores (d_1 e d_2) e também a distância entre os sensores (L). Esse formato de medição faz uso de ambos os módulos propostos na Seção 4.1 e foi concebido no pressuposto de que as toras de madeira estão livres em cima da esteira. Naturalmente, o comprimento H da tora pode ser obtido através da Equação (4.2).

$$H = L - (d_1 + d_2) \quad (4.2)$$

O sistema é projetado utilizando dois módulos justamente porque é comum que a tora de madeira não esteja livre sobre a esteira. Em casos como esse o sistema se assemelha

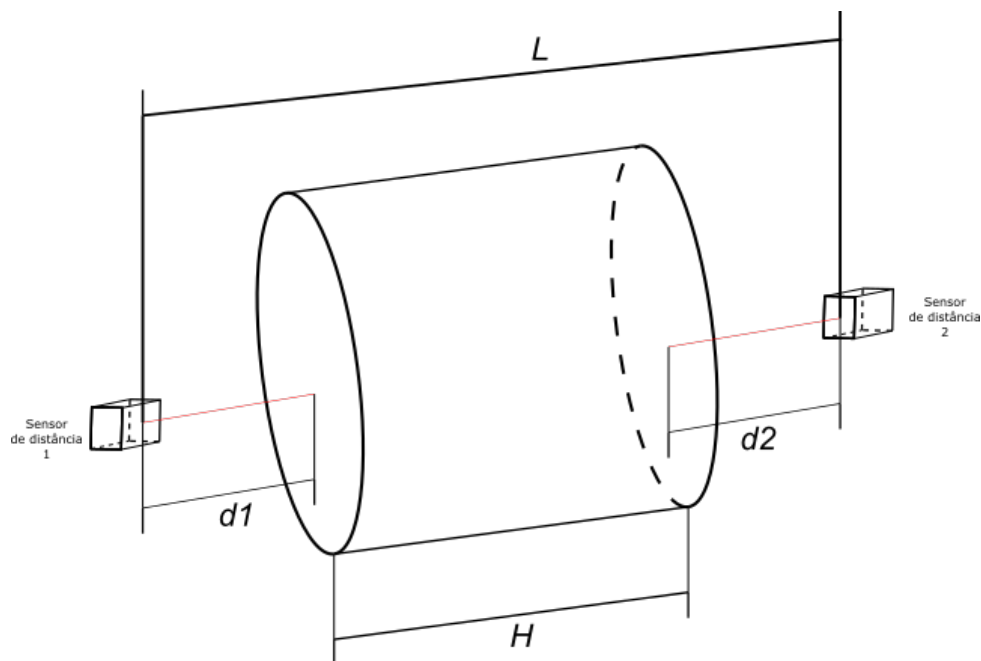


Figura 29 – Cálculo de comprimento com dois sensores. Fonte: o autor.

muito com aquele apresentado na Figura 30. Perceba que nessa situação é possível eliminar o módulo secundário e conectar o sensor de presença diretamente no servidor. Nesse ponto a distância L não é mais a distância entre os sensores, mas sim uma distância entre o sensor principal e a outra extremidade da tora de madeira, valor esse que deve ser uma constante. Nessa situação o comprimento da tora de madeira é dado através da Equação (4.3), uma simplificação da Equação (4.2). Ainda assim, para fins de projeto, esse trabalho utiliza no seu desenvolvimento o caso genérico, com dois módulos.

$$H = L - d \quad (4.3)$$

É importante que os sensores estejam posicionados próximos do eixo longitudinal da tora de madeira (z), definido na Figura 8, garantindo que seja feita uma medida correta da face do objeto. Levando em consideração que as toras de madeira podem possuir um tamanho bastante variado, é impossível que o sensor esteja sempre posicionado exatamente sobre o eixo z , contudo, para as medidas aqui necessárias, é apenas importante que o sensor esteja paralelo ao eixo z e esteja apontando diretamente para a face do objeto.

A estimativa de comprimento é, de fato, o primeiro passo para que seja possível calcular o volume da tora de madeira conforme indica a Equação (4.1). A sequência de funcionamento do sistema é bastante simples e pode ser vista na Figura 28, porém, note que a interconexão entre cada um dos processos é detalhada nas seções seguintes.

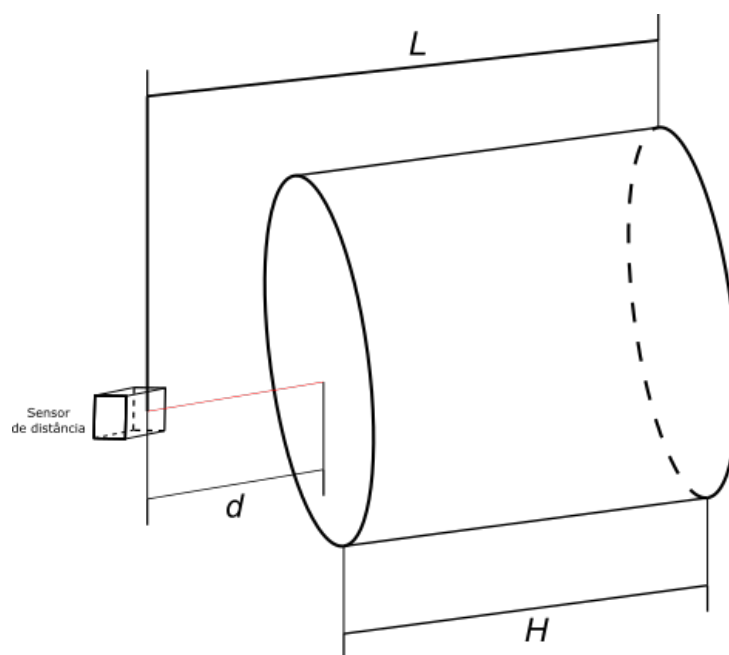


Figura 30 – Cálculo de comprimento com um sensor. Fonte: o autor.

4.1.2 Estimativa de área

Estimar a área aparente de um objeto pode ser uma tarefa difícil, principalmente se a proposta é analisar objetos cuja área aparente tem livre formato, especialmente se o formato não possui uma repetição. Apesar de existirem outras abordagens para estimar a área de objetos, aqui é empregado o uso de visão computacional para buscar uma segmentação da face da madeira e, através dessa segmentação, estimar sua área. Ambos, segmentação e cálculo de área, devem apresentar boa repetibilidade quanto aos seus resultados, portanto, os testes devem ser realizados em uma série de imagens, isto é, um banco de imagens. Inicialmente são utilizadas seis imagens com o objetivo de determinar a sequência de processamento necessária para fazer a segmentação da área de seção reta de tora de madeira, conforme visto na Figura 31.

Ainda que as fotografias da Figura 31 sejam utilizadas como base para o desenvolvimento da solução de segmentação, outra câmera é utilizada para a aquisição de fato das imagens (Figura 32) de forma a fazer atuação em conjunto com o sistema de medição de comprimento de tora (H). Algumas informações pertinentes da câmera são apresentadas na Tabela 1.

4.1.2.1 Segmentação

A segmentação tem como o principal objetivo separar a área da seção reta da tora de madeira do resto da imagem, possibilitando que na sequência seja realizada uma conversão de tal forma que a área real (em mm^2) seja calculada. É comum que um processo de segmentação passe por várias etapas, por exemplo: coleta de imagem, definição de uma

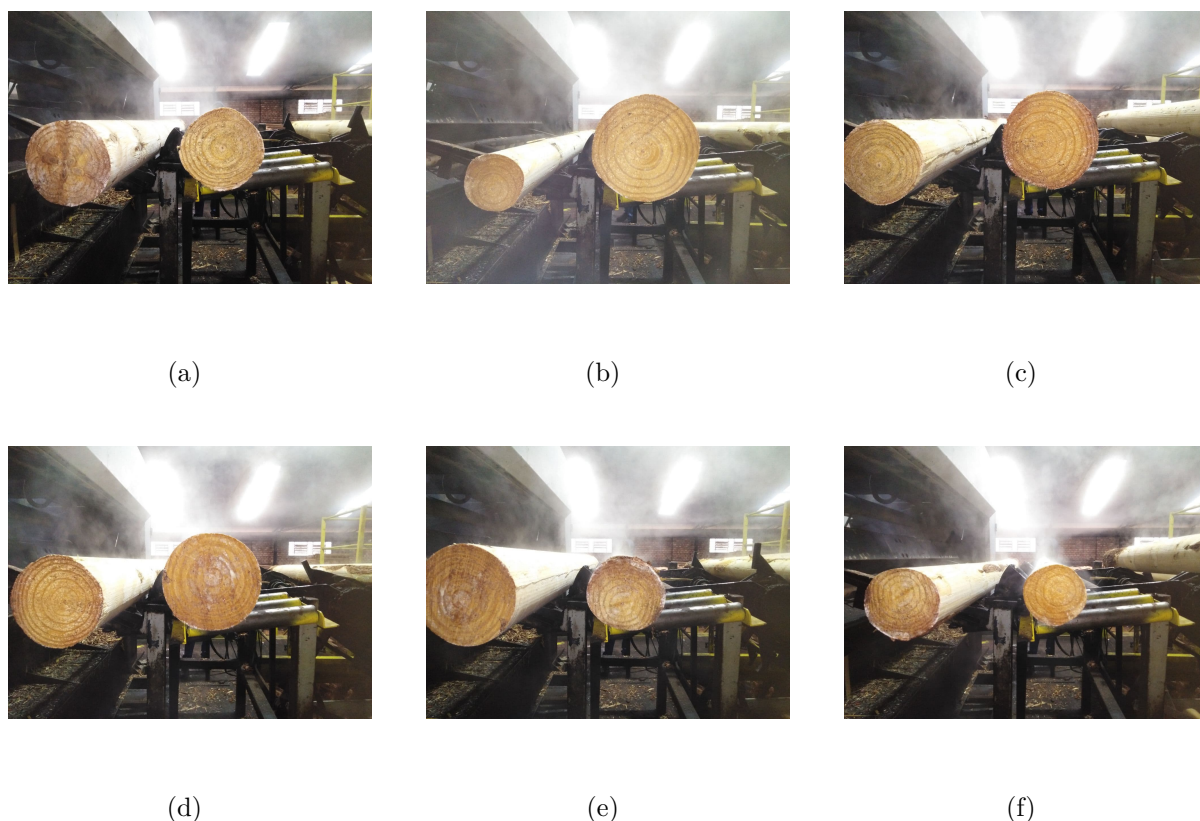


Figura 31 – Conjunto de imagens utilizadas para os testes de segmentação e cálculo de área. Fonte: o autor.

Tabela 1 – Características da câmera.

Característica	Detalhe
Sensor	OV5647
Tamanho do pixel	$1.4\mu\text{m} \times 1.4\mu\text{m}$
Área da imagem	$3673.6\mu\text{m} \times 2738.4\mu\text{m}$
Resolução	5MP (2592 × 1944)
Comprimento focal	3.6mm (ajustável)
Ângulo de visão	75.7°

região de interesse (*ROI*), conversão do espaço de cor, binarização por conjunto de cores, operações morfológicas, análise de componentes conexos e reconstrução.

As imagens coletadas para análise podem estar representadas de várias formas, dependendo da câmera. Em alguns casos são utilizadas câmeras monocromáticas, como aquelas oferecidas pela empresa Basler, resultando em imagens com apenas uma dimensão de cor, isto é, comumente imagens na escala de cor cinza. Apesar de imagens na escala de cor cinza apresentarem maior facilidade ao manipular quando comparadas à aquelas no padrão **RGB**, uma segmentação por cor se torna limitada por trata-se de uma escala unidimensional. Tendo em vista que imagens na escala de cinza podem não conter informação suficiente para executar uma segmentação em ambientes extremamente ruidosos,



Figura 32 – Câmera utilizada em conjunto com a Raspberry. Fonte: [CÂMERA... , 2021]

devido ao mapeamento permitir que duas cores distintas do espectro **RGB** sejam iguais na escala de cinza, optou-se pelo uso de imagens coloridas, captadas já no espectro **RGB**. Na Figura 33 é apresentada uma versão ampliada da Figura 31c, utilizada aqui como exemplo no desenvolvimento do algoritmo.

Note que a imagem capturada pela câmera (apresentada na Figura 33) contém bastante informação, inclusive elementos que não fazem parte da tora de madeira, o que pode dificultar a análise, indicando que é interessante determinar uma *ROI*. A *ROI* não tem um tamanho exato definido, contudo, precisa garantir que a tora de madeira esteja nela e, nessa aplicação, a região tem duas grandes funções: limitar o tamanho da imagem para facilitar a análise e também reduzir o tamanho da imagem para reduzir o custo computacional do processamento. Uma possível *ROI* para a Figura 33 pode ser vista na Figura 34, isto é, a Figura 34 representa apenas um recorte da Figura 33 com o objetivo de facilitar o processamento.

Seguindo a linha de processamento proposta, o próximo passo é converter a imagem já recortada para o espaço **HSV**, pelos motivos apresentados na Seção 3.1.1.4. O espaço de cor **HSV** vai permitir que uma segmentação por cor seja feita de forma mais simples, através de seleção de um *pixel* de cor da madeira e suas variações.

Suponha um *pixel* retirado da face da tora de madeira, cujos valores – no espectro **HSV** – são H' , S' e V' , valores os quais permitem a criação de uma imagem equivalente, de mesmo tamanho, no formato binário, onde *pixels* de mesmo valor (H' , S' e V') são representados por 1 ao passo que *pixels* diferentes são representados por 0. É natural pensar que o resultado seria uma imagem com apenas alguns *pixels* (conforme ilustrado



Figura 33 – Imagem de teste C. Fonte: o autor.



Figura 34 – ROI da Figura 33. Fonte: o autor.

na Figura 35), tendo em vista que existe uma grande variação de cor ao longo da face da tora de madeira, problema o qual pode ser contornado se inserirmos alguns valores de tolerância para o conjunto $[H', S', V']$.

Sejam ht , st e vt as tolerâncias, respectivamente, para H' , S' e V' , torna-se fácil verificar através da Figura 36 que a adição das tolerâncias nos permite selecionar um subespaço de cor pertencente ao espectro **HSV**, aumentando assim a abrangência da segmentação. Nesse ponto entra a vantagem do espectro **HSV**, note que ao escolher uma pequena

tolerância para *ht* e grandes tolerâncias para *st* e *vt* estamos na verdade escolhendo vários tons da mesma cor.

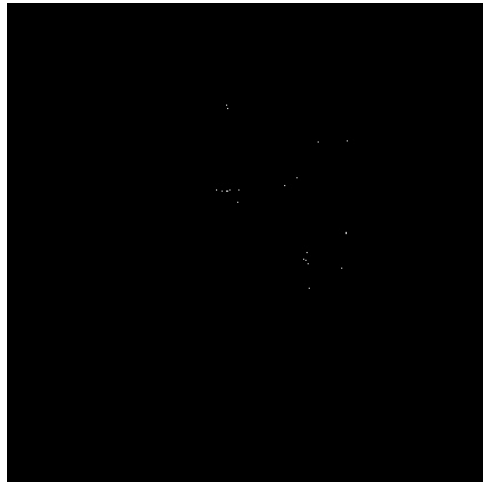


Figura 35 – Segmentação da Figura 34, por cor, sem tolerância. Fonte: o autor.

Seguindo os parâmetros da Tabela 2 é possível selecionar um subespaço de cor, dentro do espectro **HSV**, capaz de segmentar parcialmente a face da tora de madeira, como visto na Figura 37. Perceba que, apesar do resultado ser muito superior quando comparado ao da Figura 35, ainda existem várias imperfeições e, também, elementos que não pertencem a face da madeira e, dessa forma, essa informação dá origem a necessidade do próximo passo no processo de segmentação. Apesar do resultado da Figura 37 demonstrar avanços, algumas falhas devem ser corrigidas, sendo que as principais falhas são: eliminar elementos que não pertencem a face da tora de madeira e preencher a face que até então apresenta buracos. A primeira falha pode ser resolvida ao aplicar dois tipos de operações, conhecidas como operação morfológica e análise de componentes conectados, aplicadas nessa respectiva ordem.

Tabela 2 – Parâmetros da segmentação. O autor.

Parâmetro	Valor
H'	18
S'	146
V'	217
ht	10
st	57
vt	57

A primeira operação a ser feita, conhecida como operação morfológica, está definida na Seção 3.1.1.5. No total são aplicadas duas operações morfológicas na imagem da Figura 37, conhecidas como erosão e dilatação, com mesmo tamanho de *kernel*, utilizadas em conjunto com o objetivo de eliminar possíveis ruídos presentes na imagem, além de tentar desconectar elementos que não pertençam a tora de madeira. Note que o efeito

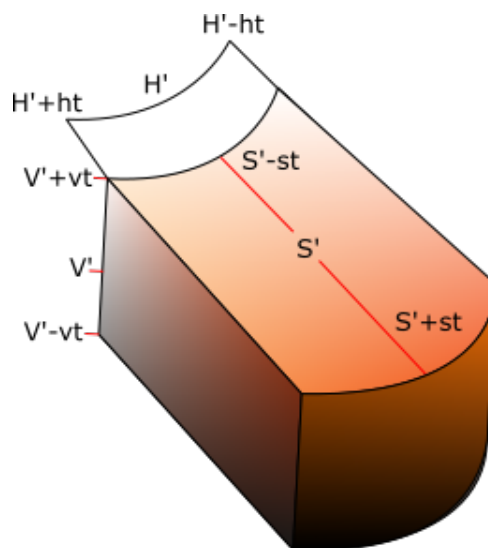


Figura 36 – Subespaço de cor **HSV**. Fonte: o autor.

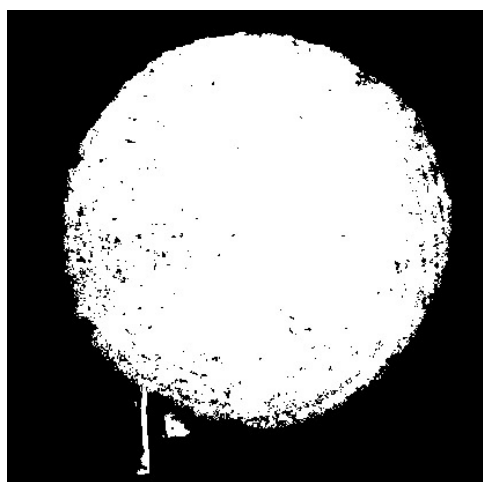
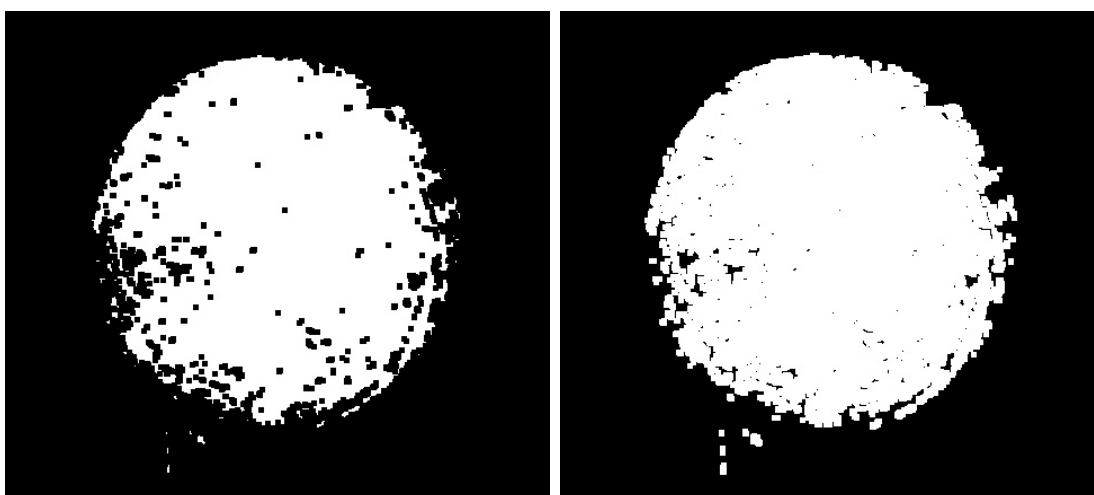


Figura 37 – Segmentação de cor utilizando um subespaço de cores. Fonte: o autor.

gerado pela aplicação das operações de erosão e dilatação é justamente a desconexão de elementos, caso estes estejam conectados por uma fina trilha de *pixels*, permitindo que posteriormente sejam identificados como elementos distintos. É importante ressaltar que as operações devem ser feitas na ordem mencionada (erosão e dilatação), do contrário o efeito obtido pode ser exatamente a conexão de objetos ao invés da desconexão.

Aplicando as operações de erosão e dilatação na Figura 37 com um *kernel* retangular definido por $h = 2$, obtemos os resultados apresentados na Figura 38. Note que o efeito desejado foi obtido, separando o objeto que não pertencia a face da tora de madeira, contudo, vale ressaltar que esse tipo de operação pode ser melhorado ao aplicarmos as operações mais de uma vez, realizando fechamentos (erosão e então dilatação) e aberturas (dilatação e então erosão), conforme apresentado na Figura 39.

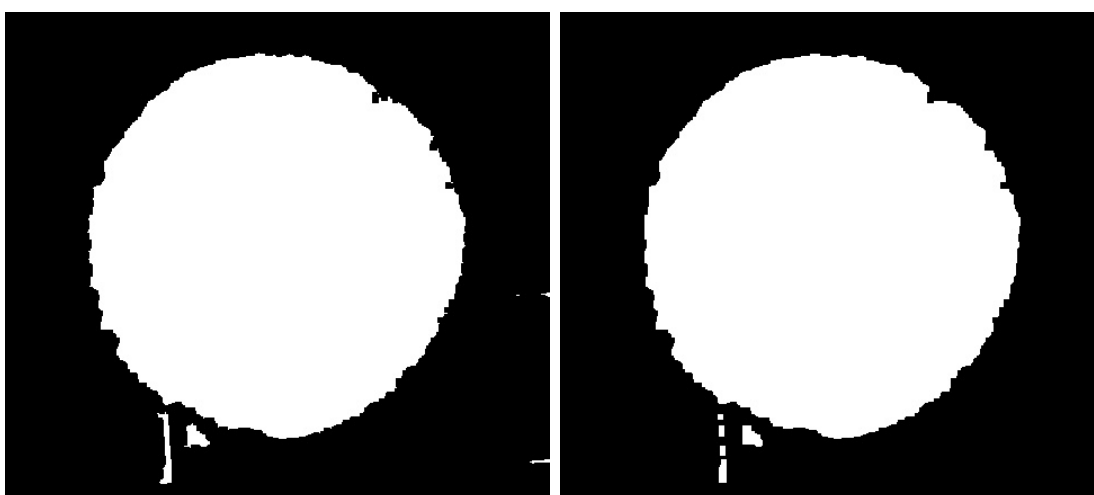
Perceba que a partir desse ponto já é possível verificar claramente que existem alguns elementos na imagem e, naturalmente, o maior deles é a face da tora de madeira, essa que



(a) Operação de erosão.

(b) Operação de dilatação.

Figura 38 – Resultados das operações de erosão e dilatação aplicadas na Figura 37.



(a) Operação de fechamento.

(b) Operação de abertura.

Figura 39 – Resultados das operações de fechamento e abertura, feitas em sequência, aplicadas na Figura 37. Fonte: o autor.

agora precisa ser reconstruída e ter seu tamanho estimado. Tomando a Figura 39b como ponto de partida é notório que bastam executar duas tarefas para que a segmentação esteja completa: excluir os menores elementos não conectados ao maior e suavizar o contorno do maior elemento. Os elementos da imagem são classificados através de uma análise de componentes conectados. Essa definição está proposta na Seção 3.1.1.6.

Agora que a análise individual de cada *blob* é uma possibilidade, é necessário definir um parâmetro que possa dizer qual dos *blobs* é, de fato, aquele desejado. Apesar de existirem diversos parâmetros possíveis para se analisar, aqui é realizada uma comparação da área

em *pixels*, afinal, em imagens como a Figura 39b é notável que a face da tora de madeira é, sem dúvidas, o maior *blob*.

A área de cada *blob* é calculada através da Equação (3.4), pertencente ao processo de descrição demonstrado na Seção 3.1.1.6. Naturalmente, o *blob* que possuir o maior valor de área deve ser a face da tora de madeira, permitindo escolher apenas os *pixels* que pertencem a essa classe como *pixels* que compõem a face da madeira. Essa operação elimina a dificuldade de entender quais *pixels* filtrados pelo subespaço **HSV** pertencem a face da tora de madeira. Nesse ponto resta apenas o problema da suavização da borda do *blob* restante.

A operação escolhida para realizar a suavização está no escopo dos problemas da geometria computacional, conhecida como envoltório convexo. Ao supor que a face da tora de madeira deve ser um conjunto convexo, é possível aplicar uma operação de envoltório convexo com o principal objetivo de encontrar o menor envoltório convexo que consiga abranger todos os pontos pertencentes ao *blob* restante. A Definição 1 mostra formalmente os requisitos para que um conjunto seja convexo.

Definição 1. Um conjunto $P \subseteq \mathbb{R}^2$ é convexo se $\overline{pq} \subseteq P$, para todo $p, q \in P$

Onde P é o conjunto que define um polígono convexo e p e q são pontos arbitrários contidos no conjunto P . Ao admitir que o resultado esperado é o de um conjunto convexo, é possível aplicar uma operação de envoltório convexo na Figura 39b, cuja saída esperada é o menor envoltório possível de tal forma que o *blob* responsável por representar a face da tora de madeira seja uma superfície convexa. A saída obtida ao aplicar as operações de eliminação de *blobs* pequenos e de envoltório convexo na Figura 39b pode ser vista na Figura 40b. Ainda, a fim de comparação, na Figura 40 vemos lado-a-lado o *ROI*, resultado final da segmentação e segmentação sobreposta com o *ROI*. Por fim, visando demonstrar os resultados do algoritmo nas imagens propostas no início da seção, vemos na Figura 41 os resultados para a segmentação aplicada em cada uma das imagens utilizadas nos testes.

Note que a sequência de operações propostas de fato consegue segmentar a face da tora de madeira de maneira satisfatória, possibilitando a continuidade do processo para estimar o volume da tora de madeira através da Equação (4.1).

4.1.2.2 Conversão de unidade de área

Como visto na Seção 4.1.2.1, o algoritmo proposto é capaz de criar um *blob* que representa a face da tora de madeira, segmentando a área desejada para o cálculo. Ao utilizar o *blob* criado em conjunto com a Equação (3.4), seguindo $p = q = 0$, têm-se como resultado exatamente uma estimativa da área da seção reta da tora, porém, em *pixels*. Para que seja possível aplicar a Equação (4.1) surge a necessidade de converter a área da seção reta da tora, agora conhecida em *pixels*, para uma unidade de medida convencional como, por exemplo, milímetros quadrados (mm^2). Para estabelecer uma relação entre os

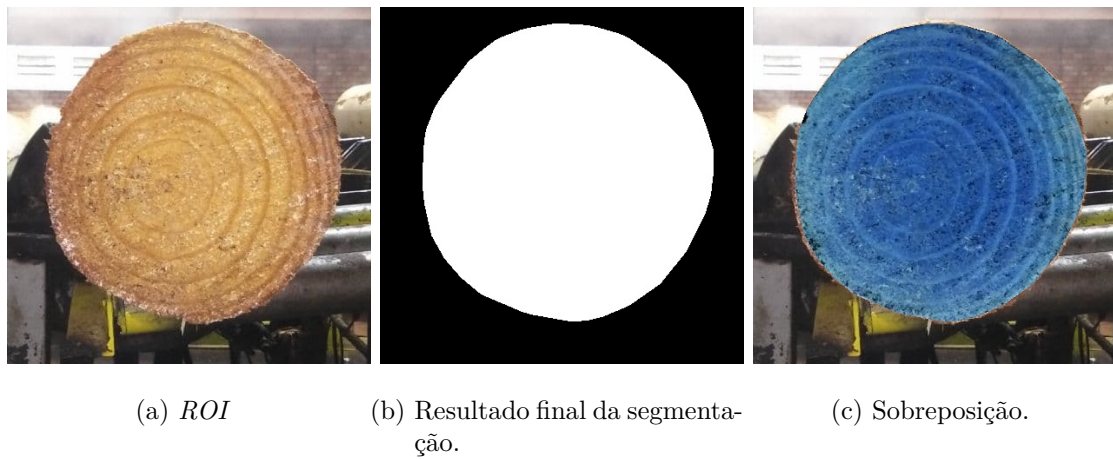


Figura 40 – Resultado final da segmentação. Note que na imagem **c** o resultado da operação de envoltório convexo está sobreposto na imagem *ROI*. Fonte: o autor.

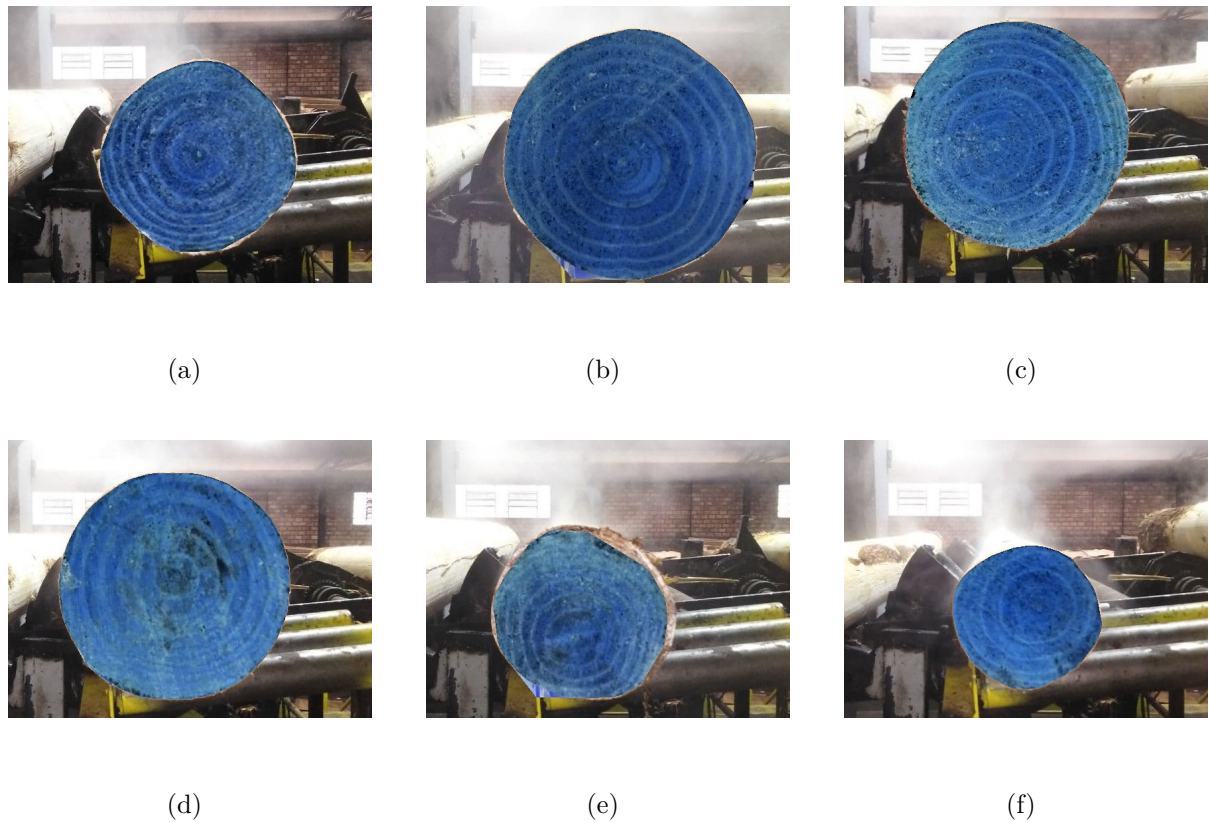


Figura 41 – Resultados da operação de segmentação proposta. Fonte: o autor.

valores de *pixel* e medidas reais toma-se como exemplo o esquema visto na Figura 42, tal qual representa um modelo clássico de lentes para uma câmera qualquer [CORKE; KHATIB, 2011][SIEGWART; NOURBAKHS; SCARAMUZZA, 2011].

Perceba que através do esquema de lente proposto cria-se a possibilidade de estabelecer uma relação entre os tamanhos do objeto real (na posição z_o) e sua respectiva representação no sensor (na posição $-f$), relação a qual é baseado na fato de que existe

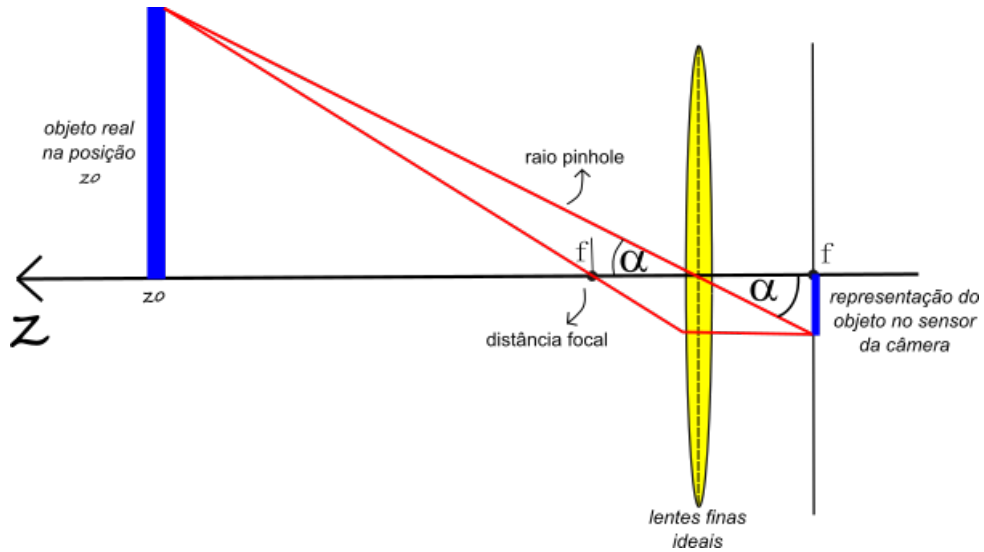


Figura 42 – Modelo utilizado para elaborar as equações de conversão de unidade. Fonte: o autor.

uma igualdade trigonométrica através do ângulo α , seguindo a Equação (4.4).

$$\tan(\alpha) = \frac{H'_{obj}}{z_o} = \frac{h'_{obj}}{f} \quad (4.4)$$

Onde H'_{obj} representa a altura real do objeto, h'_{obj} representa a altura do objeto no sensor de imagem da câmera, f é a distância focal da lente da câmera e z_o é a distância da lente da câmera até o objeto real. Rearranjando os termos da Equação (4.4) fica claro que, se o objetivo é calcular o tamanho real de um objeto qualquer, existe uma relação linear entre o tamanho real do objeto (H'_{obj}) e o tamanho do objeto no sensor (h'_{obj}), considerando que tanto f quanto z_o são constantes. Note que a relação estabelecida na Equação (4.4), apesar de ser criada pensando na altura do objeto, vale também para sua largura, conforme indica a Equação (4.5).

$$\tan(\beta) = \frac{W'_{obj}}{z_o} = \frac{w'_{obj}}{f} \quad (4.5)$$

onde W'_{obj} representa a largura real do objeto, w'_{obj} representa a largura do objeto no sensor e β é o ângulo entre o raio *pinhole* e o eixo de imagem. Note que a relação necessita conhecer duas informações para que seja possível estimar o tamanho real do objeto: distância até o objeto (z_o) e o tamanho do objeto no sensor (mm). Através da segmentação proposta na Seção 4.1.2.1 conseguimos descobrir o tamanho do objeto em *pixels*, entretanto existe outra informação bastante relevante levantada, essa que pode ser vista na Tabela 1: tamanho do *pixel*. Perceba que cada *pixel* da nossa imagem possui um tamanho real no sensor, permitindo que uma vez que tenhamos o tamanho de um objeto arbitrário em *pixels* possamos calcular seu tamanho no sensor. Tanto a altura de

um objeto no sensor (h'_{obj}) quanto sua largura (w'_{obj}) podem ser calculadas através da Equação (4.6) e Equação (4.7) respectivamente.

$$h'_{obj} = \psi_h h^*_{obj} \quad (4.6)$$

$$w'_{obj} = \psi_w w^*_{obj} \quad (4.7)$$

Sendo que ψ_h é a altura do *pixel* em *mm*, ψ_w é a largura do *pixel* em *mm*, h^*_{obj} é a altura do objeto em *pixels* e w^*_{obj} é a largura do objeto em *pixels*. Ao substituir a Equação (4.6) na Equação (4.4) e substituir a Equação (4.7) na Equação (4.5) obtemos, respectivamente, a Equação (4.8) e Equação (4.9) que, se multiplicadas, dão origem a Equação (4.10).

$$H'_{obj} = h^*_{obj} \frac{z_o \psi_h}{f} \quad (4.8)$$

$$W'_{obj} = w^*_{obj} \frac{z_o \psi_w}{f} \quad (4.9)$$

$$A'_{obj} = a^*_{obj} \frac{z_o^2 \psi_h \psi_w}{f^2} \quad (4.10)$$

Note que A'_{obj} é na verdade a área real aparente do objeto, $a^*_{obj} = h^*_{obj} w^*_{obj}$ é a área do objeto em *pixels* e, ainda, caso o *pixel* seja quadrado como o da câmera proposta na Tabela 1, temos que $\psi_h = \psi_w \implies \psi_h \psi_w = \psi^2$. A estimativa de área da tora de madeira pode ser concluída então através da Equação (4.10), uma vez que se a câmera for posicionada ao lado do sensor de distância 1 (Figura 29), o valor de medida $d1$ é exatamente o o valor de distância entre objeto e lente (z_o). Basta então que o processamento de imagem seja feito logo após a coleta do valor de $d1$ no funcionamento do sistema.

4.1.3 Estrutura Computacional

4.1.3.1 Microcontrolador

Os dois primeiros passos da sequência de funcionamento, apresentada na Figura 28, podem ser realizados utilizando um microcontrolador que seja capaz de receber um sinal indicando que a tora de madeira está na posição desejada e que consiga fazer a leitura de um sensor de distância a laser. O microcontrolador utilizado para realizar as aquisições mencionadas é conhecido como Esp32-CAM (Figura 43), tal placa contempla uma série de GPIOs (*General Purpose Input/Output*), conexão bluetooth, Wi-Fi, periférico para comunicação *I²C* e, também, suporte para o acoplamento de uma câmera que não é utilizada nesse projeto.

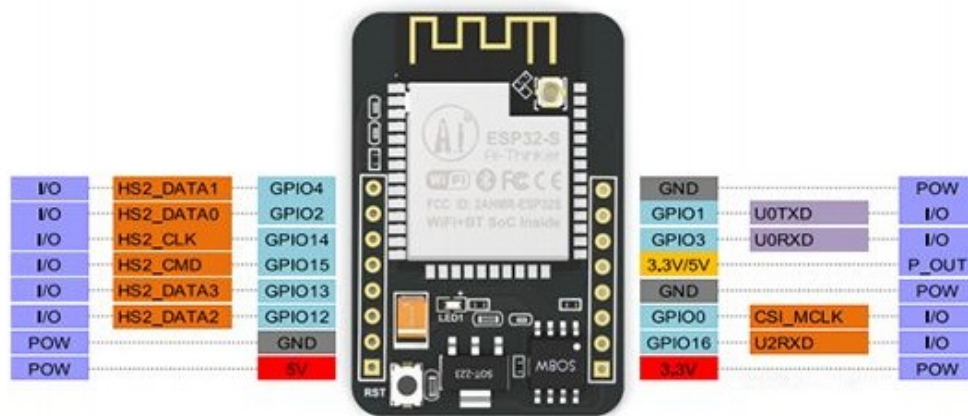


Figura 43 – Esp32-CAM. Fonte: [CâMERA..., 2021].

Uma das atribuições do microcontrolador é detectar a presença da tora de madeira na posição correta. Existem diversas formas de detectar a presença de um objeto em uma posição com sensoriamento industrial. Neste projeto, especificamente é utilizado um sensor capacitivo que é comumente encontrado na detecção de objetos não metálicos. A Esp32-CAM é programada através da IDE Arduino, permitindo fazer-se uso da grande maioria das bibliotecas desenvolvidas para as placas Arduino, possibilitando que suas GPIOs sejam configuradas como entradas de sensores industriais. Uma vez que alguma GPIO do microcontrolador é configurada para receber um sinal indicando a presença da tora de madeira, essa mesma informação lida pela GPIO deve ser utilizada como um gatilho dentro do algoritmo indicando que o processamento deve iniciar. O fragmento de código abaixo seleciona a GPIO 16, apresentada na Figura 43, como entrada e demonstra o mínimo necessário para que seja efetuado o gatilho. Note que tratamentos do sinal de entrada podem ser necessários, afinal, alguns sensores apresentam o fenômeno de *switch bouncing*, que poderá acionar o gatilho em momentos indesejados.

```
// Pré configurações do algoritmo
void setup(){
    ...
    pinMode(16, INPUT_PULLUP);
    ...
}
void loop(){
    int sensorSignal = digitalRead(16);
    if (sensorSignal == HIGH){
        //Ação após detectar presença de tora
        ...
    }
}
```

Tabela 3 – Características do sensor VL53L0X.

Característica	Detalhe
Tamanho	4.40 x 2.40 x 1.00 mm
Tensão de operação	2.6 até 3.5 V
Temperatura de operação	-20 até 70 °C
Emissor infravermelho	940 nm

}

O sinal que indica a presença da tora de madeira tem a função de iniciar as coletas de dados que dão origem ao processamento e, por fim, permitem executar o cálculo da estimativa de volume. Seguindo a sequência estabelecida na Figura 28 vemos que o próximo passo é estimar a distância d_2 (Figura 29), que aqui é medida através de um sensor de distância a laser acoplado ao microcontrolador, utilizando como meio de comunicação o protocolo I^2C . O sensor utilizado para medir a distância d_2 é conhecido por VL53L0X [VL53L0X..., 2021] e algumas informações pertinentes à esse sensor são apresentadas na Tabela 3. O sensor VL53L0X deve ser posicionado conforme o sensor de distância 2, presente na Figura 29, de forma que seu raio emissor infravermelho seja paralelo ao eixo longitudinal da tora de madeira (z), garantindo boa medida.

Naturalmente, o microcontrolador não detém processamento suficiente de forma que seja possível estimar todas as dimensões da tora de madeira através dele, contudo, a Esp32-CAM é capaz de utilizar comunicação via rede sem fio (Wi-Fi) e enviar os dados coletados para um servidor capaz de finalizar o processamento, concedendo ao microcontrolador apenas três funções: receber sinal de presença de objeto, ler a distância d_2 no registrador do sensor e enviar os dados para um servidor. O servidor em questão é discutido com mais detalhes na Seção 4.1.3.2.

4.1.3.2 Servidor e processamento

O microcontrolador apresentado na Seção 4.1.3.1 é responsável por executar as duas primeiras operações apresentadas na Figura 28, deixando apenas dois grandes problemas a serem resolvidos: medir a distância d_1 e estimar a área da seção reta da tora de madeira. Ambos os problemas são aqui tratados no servidor, este que pode ser qualquer computador capaz de rodar um sistema operacional que consiga executar códigos na linguagem Python. Inicialmente o servidor deve ser capaz de receber informações enviadas pelo microcontrolador, logo, conclui-se que o servidor obviamente deve possuir uma forma de se comunicar, seja através de uma comunicação sem fio ou através de um cabo ethernet.

Um computador que atende os requisitos citados anteriormente é a Raspberry Pi 3 Model B [RASPBERRY..., 2021] vista na Figura 44. A Raspberry é um computador que possui um processador baseado na arquitetura ARM, suportando sistemas operacionais baseados em Linux como por exemplo o Raspbian, além de ter comunicação via cabo

ethernet e sem fio (Wi-Fi), uma série de GPIOs, portas USB e uma porta para conexão com câmera, utiliza para comunicação com uma câmera compatível com a Raspberry Pi 3 Model B.

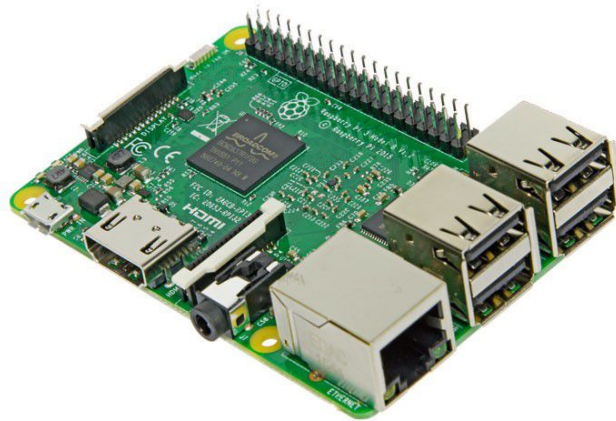


Figura 44 – Raspberry Pi 3 Model B. Fonte: [RASPBERRY. . . , 2021].

A Raspberry não é apenas responsável por receber os dados do microcontrolador, mas também por realizar a coleta da distância d_1 e processar a imagem. Inicialmente, para receber os dados do microcontrolador é necessário que um protocolo de comunicação seja escolhido, garantindo a comunicação entre Esp32-CAM e Raspberry. Note que o microcontrolador de fato só precisa enviar a informação de distância d_2 para que a Raspberry termine o processamento, assim, pode-se utilizar o envio da informação de distância como um gatilho para continuar o processamento.

O protocolo *MQTT* é o escolhido para esse projeto, visto que se enquadra bem com a situação descrita anteriormente, uma vez que este segue o conceito de *publish* e *subscribe*, conforme descrito na Seção 3.2.2.1, permitindo que a publicação da distância d_2 em um tópico específico funcione como um gatilho para a Raspberry, dando início a coleta do valor de distância d_1 e, na sequência, continuar o fluxo de processos descrito na Figura 28. Outra característica interessante do protocolo *MQTT* é sua facilidade de uso, permitindo criar com pouco esforço um servidor local na Raspberry, que serve de destino para as mensagens enviadas pelo microcontrolador. É importante perceber que outro protocolo poderia ser utilizado, como por exemplo o HTTP, apesar do escolhido ser o *MQTT*.

O cliente aqui escolhido para funcionar como servidor de dados é conhecido como Eclipse Mosquitto, um *message broker* de código aberto com licenças EPL/EDL imple-

mentando o protocolo *MQTT*, ideal para enviar pequenos pacotes de dados consumindo pouco processamento, característica que corrobora com a arquitetura proposta, baseada em equipamentos com baixa capacidade computacional.

A arquitetura já é capaz de detectar a presença de uma tora de madeira na posição desejada, calcular a distância d_2 através de um sensor a laser, enviar os dados coletados para um servidor *MQTT* e armazenar os dados no servidor (Raspberry). O próximo passo lógico é medir a distância d_1 , que é feito ao conectar um sensor de distância VL53L0X nas GPIOs presentes na Raspberry, similar ao realizado no microcontrolador, ainda utilizando o protocolo *I²C*. A decisão para medir a distância d_1 é tomada após o servidor receber o valor da distância d_2 enviada pelo microcontrolador.

Note que a Raspberry, fazendo o trabalho de servidor, deve ser capaz de se conectar com o *broker MQTT*, que neste trabalho é um serviço interno, além de se conectar com um sensor VL53L0X para medição de distância e, por fim, também fazer o processamento para estimativa de área. Todos esses processos precisam estar concatenados em um único algoritmo de forma que o sistema não fique extremamente lento e, para atingir esse feito, o algoritmo é desenvolvido na linguagem de programação Python, essa que disponibiliza de bibliotecas capazes de conectar em um *broker MQTT*, fazer a leitura do sensor VL53L0X, além de ter suporte para câmeras acopladas na Raspberry e fácil acesso a biblioteca OpenCV, utilizada na seção 4.1.2.

O algoritmo desenvolvido em Python é tratado como um serviço por parte da Raspberry, onde este se conecta na porta do *broker MQTT* (1833 por padrão) e se inscreve no tópico cujo microcontrolador irá publicar, tópico este definido pela *string Esp32CAM / sensores / d2*. Toda vez que alguma informação é publicada no tópico mencionado o algoritmo recebe um sinal do *broker*, indicando que existem dados a serem coletados, isto é, o valor d_2 foi enviado ao *broker* pelo microcontrolador. Na sequência, logo após coletar a informação de distância d_2 , o algoritmo deve utilizar o sensor VL53L0X acoplado a Raspberry para medir a distância d_1 , através do protocolo *I²C*, garantindo que todas as informações necessárias para o cálculo do comprimento da tora (H) estejam prontas, possibilitando que o algoritmo agora utilize-se da Equação (4.2), dado que a distância entre os sensores L é uma distância fixa, determinada durante a montagem física do projeto.

Em síntese, o sistema já é capaz de solucionar o problema da estimativa de comprimento da tora de madeira (H), uma vez que os sensores de distância 1 e 2 são, respectivamente, lidos através da Raspberry e do microcontrolador Esp32-CAM, interligados através do protocolo de comunicação *MQTT*, permitindo a transferência de informação.

5 Resultados

Este capítulo apresenta os resultados obtidos pela arquitetura proposta anteriormente. Os corpos de prova são testados em uma bancada de testes, que é equivalente ao processo produtivo real, descrita na Seção 5.1. Os resultados obtidos através do uso dessa bancada podem ser vistos na Seção 5.2, onde é feita uma discussão a respeito dos mesmos. Finalmente, na Seção 5.3, são levantadas algumas considerações a cerca do sistema.

5.1 Metodologia de testes e protótipo

Com o objetivo de testar a arquitetura proposta, uma versão equivalente do sistema foi montada em uma bancada de testes. Uma visão geral do sistema equivalente é apresentada na Figura 45. Na bancada apresentada podemos ver uma mesa redonda, capaz de girar tanto no sentido horário quanto no sentido anti horário. Essa mesa tem a função de simular o movimento causado pela esteira, trazendo as toras de madeira para frente da câmera. Já a câmera e o sensor de distância a laser, que são conectados diretamente ao servidor, estão posicionados de madeira perpendicular a mesa. A Figura 46 ilustra como estão montados câmera e sensor.

Buscando simular a presença de madeira na mesa giratória é feito o uso de fotos impressas. Essas fotos ficam apoiadas sobre de suportes de plástico com diferentes alturas, ao passo que os suportes de plástico ficam apoiados na mesa giratória. Os suportes são utilizados para dar o efeito de toras de madeiras mais longas e mais curtas, isto é, mais próximas ou mais distantes da câmera. Logo abaixo dos suportes de plástico existe um pequeno elemento metálico grudado à mesa, que dispara o sensor de presença quando o suporte estiver na posição. O sensor de presença utilizado é do tipo capacitivo e está posicionado abaixo da mesa, conectado ao microcontrolador.

Como os sensores de distância a laser são exatamente iguais, para fins de bancada de teste não há a necessidade de usar ambos. Dessa forma, o valor medido pelo sensor conectado ao servidor é utilizado tanto para a medida $d1$ quanto para a medida $d2$, isto é, $d1 = d2$. Um atraso de $300ms$ é adicionado ao microcontrolador, logo após a presença do suporte ser detectada, simulando o tempo que ele poderia levar para medir o sensor. Já para distância entre sensores, definida na Equação (4.2) como L , é tomado o valor de $3000mm$. Então o comprimento da tora (H) deve ser calculado através da Equação (5.1), onde $d1$ é a distância medida pelo sensor posicionado junto com a câmera na Figura 46.

$$H = 3000 - 2d1 \tag{5.1}$$

Naturalmente, para efeito de comparação, uma medida de maior precisão deve ser

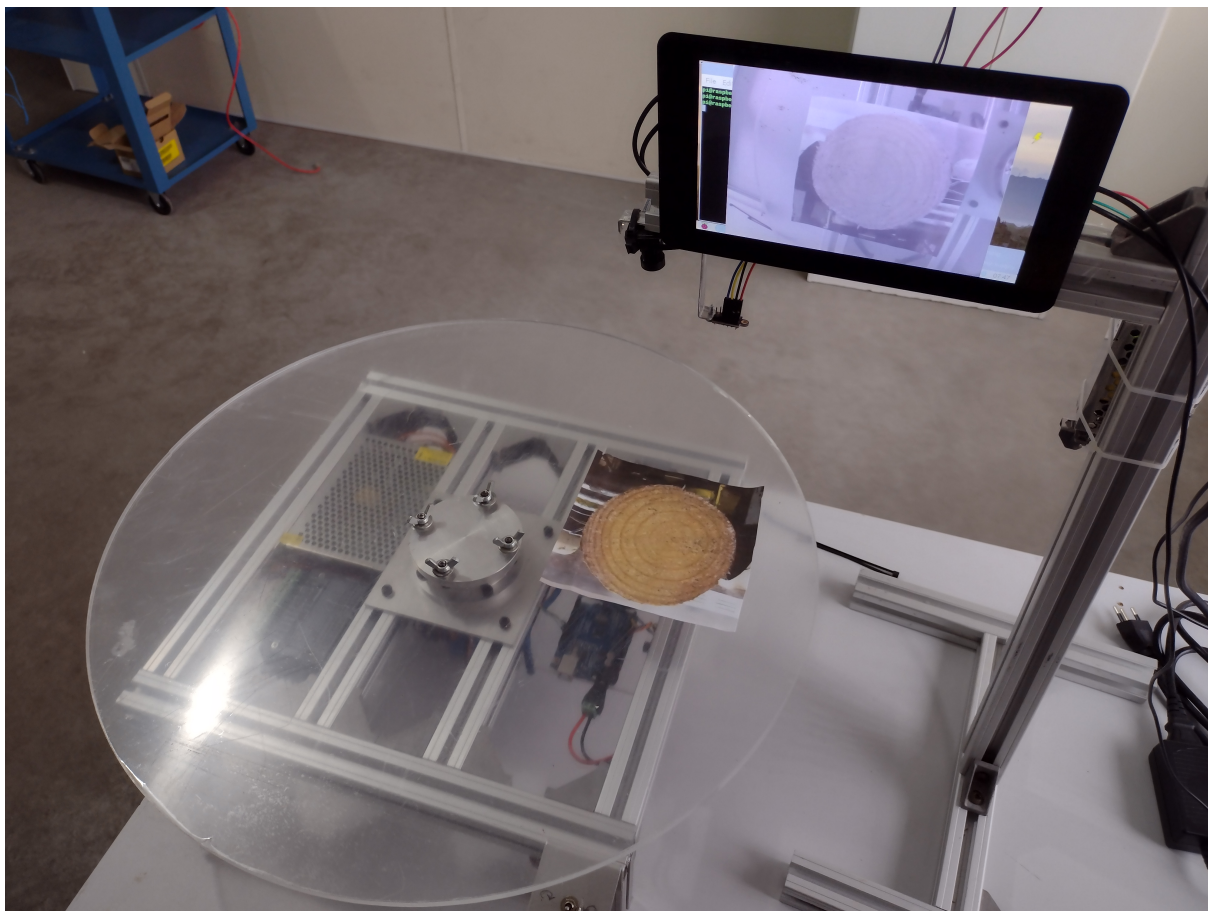


Figura 45 – Vista geral da bancada de testes. Fonte: o autor.

Tabela 4 – Medidas da bancada de testes.

Característica	Medida
Distância sensor-mesa	$450 \pm 1 \text{ mm}$
Distância sensor-câmera	$30 \pm 1 \text{ mm}$
Tamanho suporte 1	$100 \pm 1 \text{ mm}$
Tamanho suporte 2	$110 \pm 1 \text{ mm}$
Tamanho suporte 3	$120 \pm 1 \text{ mm}$
Tamanho suporte 4	$130 \pm 1 \text{ mm}$

tomada, permitindo que seja possível averiguar a qualidade da solução proposta. A medida $d1$, definida na Equação (4.2), é conhecida, uma vez que o tamanho do suporte e a distância entre mesa giratória e sensor são conhecidas. Perceba também que existe um deslocamento entre o sensor de distância laser e a câmera na Figura 46, também conhecido. Todas as medidas são apresentadas na Tabela 4.

A medida de área é realizada de forma indireta através de um paquímetro que mede o diâmetro da tora. As fotos impressas colocadas sobre os suportes são ajustadas digitalmente, através de um programa de edição de imagens, para garantir que as toras sejam circulares. Através da medida de diâmetro da tora de madeira na foto e do uso da Equação (5.2) é feito o cálculo de área. Perceba que na Equação (5.2) a variável D

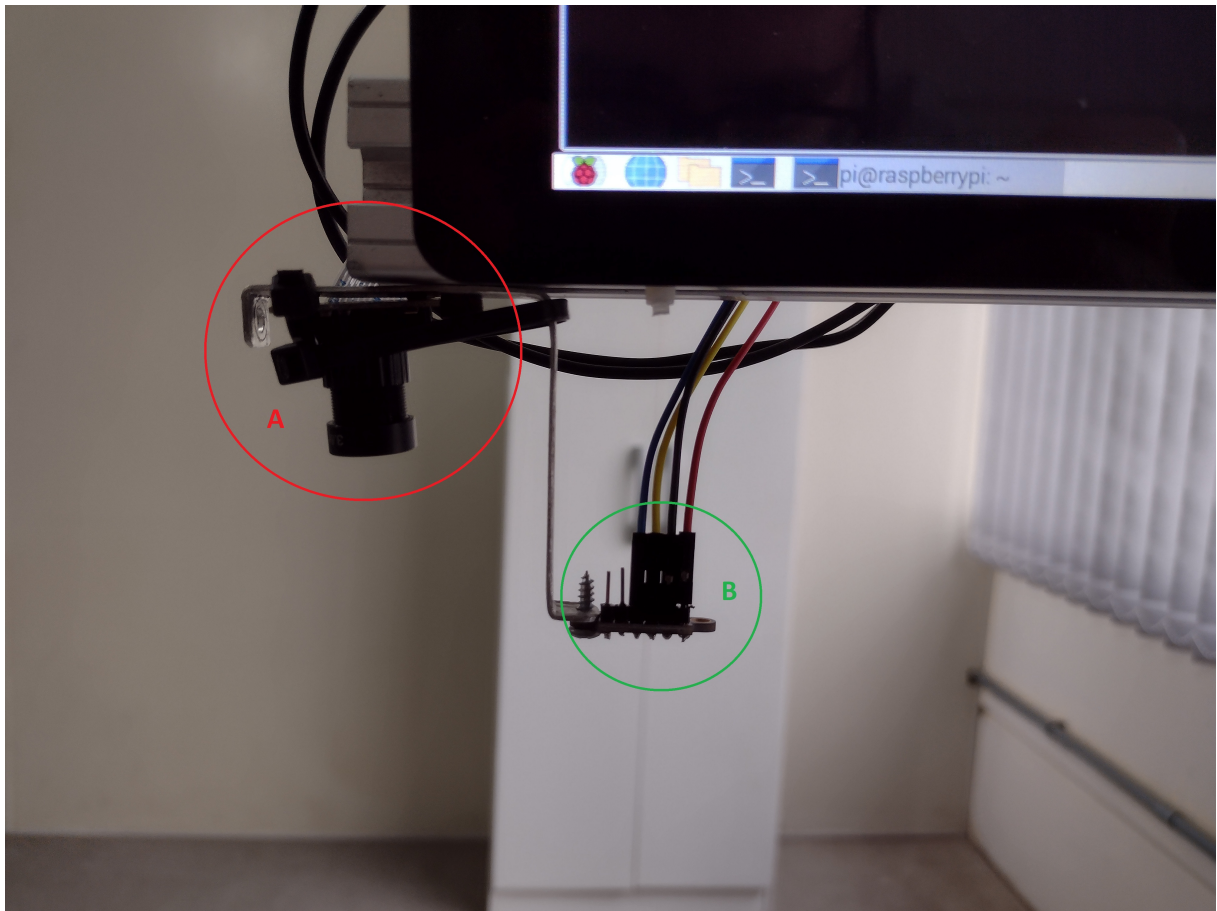


Figura 46 – Vista do posicionamento da (A) câmera e do (B) sensor de distância a laser.
Fonte: o autor.

representa o diâmetro real da tora de madeira e A é a área real. É importante ressaltar que a Equação (5.2) é válida porque as imagens escolhidas para teste possuem faces de madeira com geometrias muito próximas a de uma circunferência, como já mencionado. Note que o teste é válido, uma vez que as técnicas de segmentação e conversão de unidade empregadas no sistema não possuem nenhuma dependência com a circularidade, isto é, o sistema não sabe que está tratando de toras perfeitamente circulares.

$$A = \frac{\pi D^2}{4} \quad (5.2)$$

A metodologia do teste aplicado consiste em posicionar quatro suportes na mesa giratória, cada um contendo uma foto impressa de uma tora de madeira, ao passo que a mesa efetua uma lenta rotação, seja no sentido horário ou anti-horário. A rotação da mesa faz com que o sistema execute as medidas sempre que um suporte passar próximo ao sensor de presença.

5.2 Validação do sistema

Os resultados preliminares obtidos através do teste proposto, utilizando uma foto cuja tora possui aproximadamente 96mm de diâmetro, podem ser vistos na Tabela 5.

Tabela 5 – Resultados preliminares para 4 objetos de teste.

Característica	$d1$ real	erro $d1$ medido	A real	erro A medido
Suporte 1	350 mm	1.1 %	7238 mm ²	2.1 %
Suporte 2	340 mm	2.6 %	7238 mm ²	5.7 %
Suporte 3	330 mm	4.2 %	7238 mm ²	0.2 %
Suporte 4	320 mm	3.1 %	7238 mm ²	5.8 %

É importante executar testes longos e com várias amostras para averiguar o comportamento do sistema. Partindo do mesmo princípio abordado na Seção 5.1 os testes aqui são feitos através de fotos apoiadas nos suportes. No total foram realizados 20 processamentos com as amostras, cujas grandezas estimadas são armazenadas. Posteriormente cada uma das medidas feitas pelo sistemas é comparada com sua equivalente de referência.

Dentre as medidas realizadas é preciso analisar principalmente os valores de $d1$, A , V e o tempo que o sistema demorou para fazer todo o processamento, bem como os erros relacionados a cada uma das grandezas exceto o tempo. Vale salientar que para fins de teste, conforme já discutido, o valor de $d2$ é exatamente igual ao valor de $d1$. Na Figura 48, Figura 49 e Figura 50 estão apresentados, respectivamente, os valores de $d1$, A , V medidos pelo sistema proposto em comparação com os valores reais. Já na Figura 51 existe uma curva de tempo de processamento do sistema, onde cada ponto da curva representa um ciclo de processamento para uma tora de madeira.

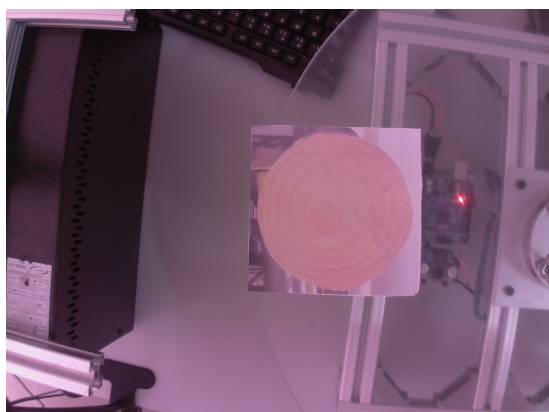
Também foram criadas curvas de erro para as medidas $d1$, A e V , apresentadas de forma sobreposta na Figura 52. Através dos dados obtidos nas curvas criaram-se médias dos erros associados a cada grandeza dimensional das toras. Os valores de erro médio, maior erro e desvio padrão podem ser vistos na Tabela 6.

5.3 Considerações

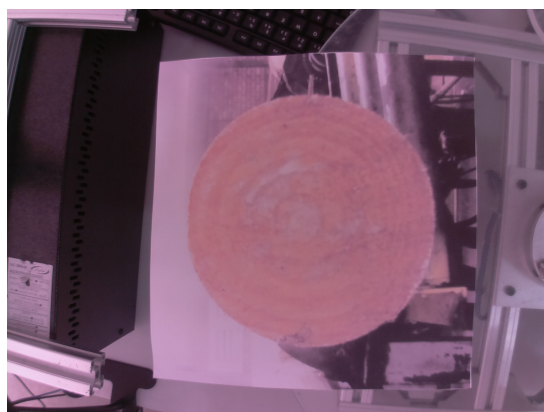
Os principais requisitos que o sistema precisa atender, que também estão apresentados no Capítulo 2 são: medir o volume da tora de madeira, processar as informações em sincronia com a linha produtiva, realizar o processamento completo em até 5 segundos

Tabela 6 – Medidas da bancada de testes.

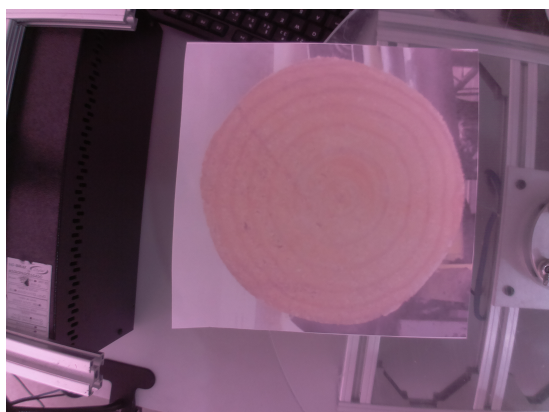
Medida	Erro Médio	Maior Erro	Desvio Padrão
$d1$	1.6 %	4.2 %	14 mm
A	4.6 %	8.7 %	2322 mm ²
V	4.4 %	8.5 %	5.6x10 ⁶ mm ²



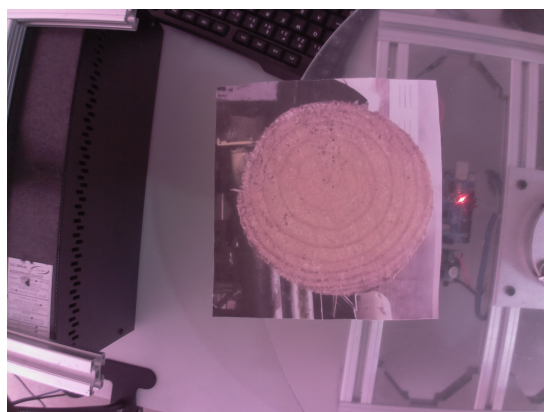
(a)



(b)



(c)



(d)

Figura 47 – Exemplos de imagens coletadas para os testes finais. Fonte: o autor.

e obter um erro médio menor do que 10%. Medir o volume das toras de madeira e processar as informações em sincronia com a máquina são requisitos que foram atendidos com sucesso pelo sistema proposto, contudo, resta analisar os erros de medida e o tempo de processamento.

Supondo um cenário em que a segmentação é sempre bem sucedida, conclui-se que $d1$ é o elemento cujo erro afeta mais o sistema. Tanto a Equação (4.2) quanto a Equação (4.10) indicam que ambos, comprimento e área, dependem de $d1$. Ainda, a Tabela 6 indica que o erro associado ao sensor de distância a laser é razoavelmente alto, considerando que esse tipo de sensor geralmente apresenta boa precisão. Esse comportamento é presente principalmente pelo fato de que o modelo de sensor escolhido é de baixo custo, não conferindo as melhores características de precisão possíveis. Idealmente, para um produto comercial,

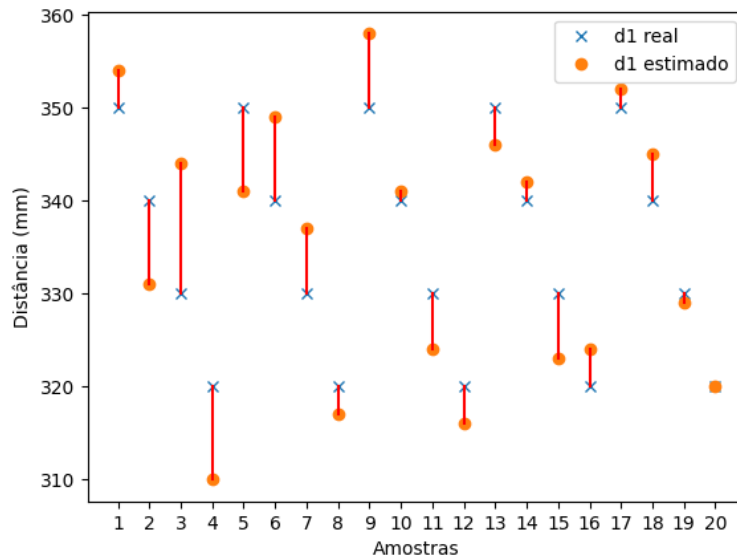


Figura 48 – Comparação entre os valores medidos e reais de $d1$. Fonte: o autor.

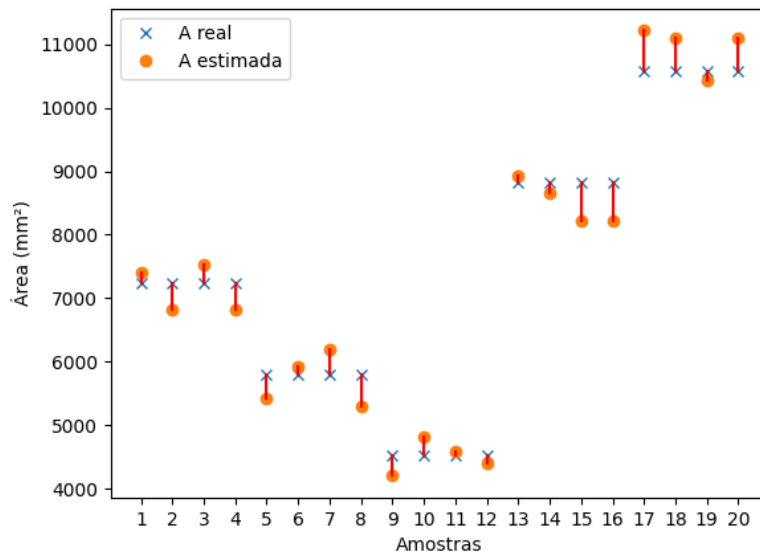


Figura 49 – Comparação entre os valores medidos e reais de A . Fonte: o autor.

o sensor deve ser substituído por uma versão industrial que apresente uma precisão de ± 1 mm.

Agora para um cenário qualquer, os resultados indicam que existem momentos nos quais a segmentação não obteve sucesso. A amostra de número 20 é um exemplo claro, que na Figura 48 indica que a medida de $d1$ foi exata e precisa, contudo, a Figura 49 indica que a estimativa de área obteve erros. Nesse exemplo discutido o problema está na segmentação. Analisando a Figura 47 fica nítido que a câmera proposta para o sistema não tem boa qualidade de cor, fator que deve ser repensado, afinal a base da segmentação

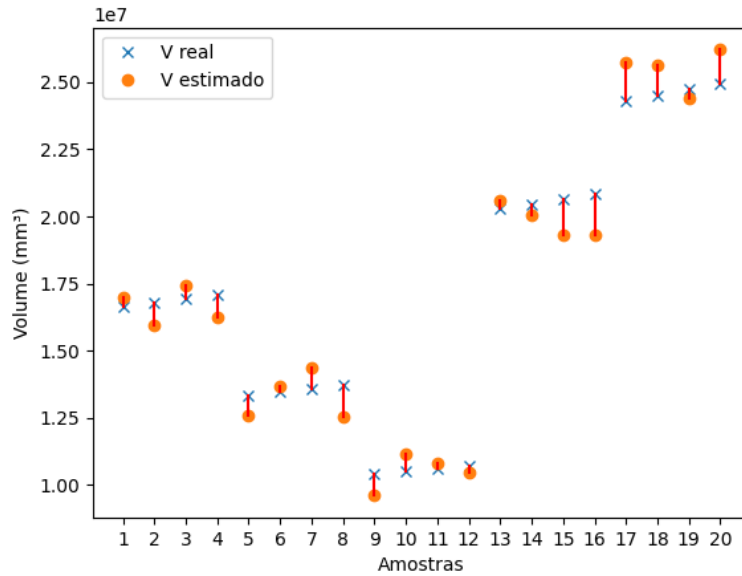


Figura 50 – Comparação entre os valores medidos e reais de V . Fonte: o autor.

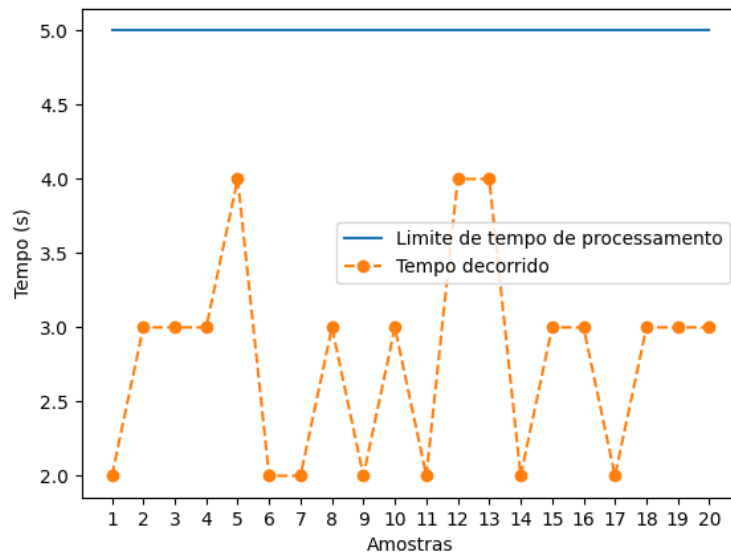


Figura 51 – Curva de tempo de processamento do sistema. Note que o processamento do sistema é menor do que o limite proposto. Fonte: o autor.

proposta é a utilização de um subespaço de cores.

Ainda que o sensor e a câmera não tenham apresentado a consistência desejada, vemos que tanto o tempo de processamento como a estimativa de volume mantiveram-se dentro do esperado. O tempo de processamento deve sofrer grande redução caso o servidor não utilize uma GUI ou caso seja substituído por outro com maior capacidade de processamento. Mesmo assim é possível detectar que no tempo de processamento existe uma falta de consistência, apresentando variações de até 2 segundos. Essa inconsistência está

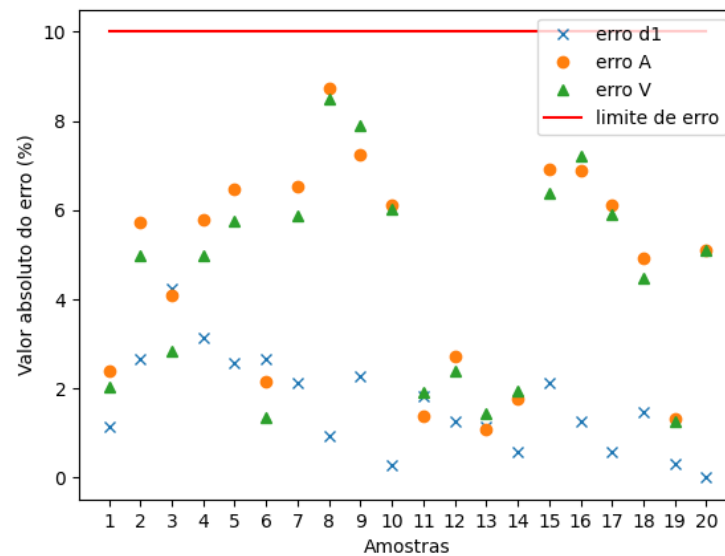


Figura 52 – Erro associado as medidas de $d1$, A e V feitas pelo sistema. Fonte: o autor.

atrelada a comunicação entre módulos secundário e primário, sugerindo que o servidor apresenta dificuldades em fazer todo o processamento e ainda manter uma comunicação externa. Em momentos o servidor responde rapidamente a comunicação *MQTT*, já em outros essa espera pode ser de até 2 segundos. Por outro lado as técnicas empregadas no processo de segmentação se mostraram suficientemente boas, mesmo que o ambiente seja ruidoso e pouco controlado.

A conclusão que se chega ao analisar as informações apresentadas é de que o sistema está pronto para uma primeira implantação em ambiente industrial. Mesmo que o uso de equipamentos próprios para a indústria muito provavelmente melhore a precisão do sistema, nesse instante ele já apresenta boa velocidade e boa precisão, podendo substituir o trabalho manual de estimação de volume de toras de madeira. Ademais, é esperado que o desempenho do sistema na indústria seja substancialmente inferior, considerando os pontos negativos dos equipamentos.

6 Considerações finais

O principal objetivo desse trabalho foi desenvolver e testar uma arquitetura capaz de realizar a medição das dimensões de toras de madeira. Os resultados do trabalho foram obtidos ao comparar as medidas feitas pelo sistema com medidas tomadas como reais, ambas realizadas em uma bancada de testes equivalente ao processo real na indústria de laminação. Para a análise de resultados foram consideradas as medidas de $d1$, A , V e o tempo de processamento realizadas pelo sistema. Os resultados envolvem os erros das medidas que não devem passar de um limiar (10%) e o tempo de processamento que não deve exceder 5 segundos.

O sistema proposto se mostrou apto a resolver a problemática descrita no Capítulo 2. Os erros médios e máximos obtidos ficaram abaixo do limite estipulado de 10%, sendo que o maior erro registrado foi de 8.7%. O tempo de processamento máximo também ficou abaixo dos 5 segundos, indicando que o sistema consegue atuar em linha com o processo produtivo sem grandes problemas. As medidas de desvio padrão, por outro lado, sugerem que o sistema possui uma substancial variação em seus resultados, favorecendo a conclusão de que a precisão precisa ser melhorada.

Os resultados se mostram muito promissores e encorajadores, considerando os limites estabelecidos e o *hardware* empregado. Uma eventual troca do sensor de distância, para uma versão industrial, deve impactar muito positivamente o sistema em todas as medidas. Quanto ao processo que envolve a visão computacional, o uso de uma luz própria para o sistema, somado ao emprego de uma nova câmera com maior precisão de cor, provavelmente podem solucionar a inconsistência apresentada pela segmentação, que sofre bastante com a variação da luz ambiente.

A solução se mostra então pronta para uma implementação em ambiente industrial, permitindo validar o sistema como um potencial produto comercial. Esta implementação é importante, visto que o ambiente industrial apresenta geralmente diversos desafios que não são possíveis de se prever em uma bancada, como por exemplo o excesso de vapor que pode acumular gotículas de água na lente da câmera.

6.1 Trabalhos futuros

Estão listados a seguir alguns itens que podem complementar o sistema, bem como algumas sugestões para a continuação e trabalhos relacionados:

1. Algumas toras de madeira, mesmo após o processo de arredondamento, podem apresentar uma geometria parecida com a de um segmento de cone. Realizar um estudo

da implementação de uma segunda câmara do outro lado da tora para melhorar a precisão pode ser importante.

2. Neste trabalho os efeitos causados pela distorção das lentes foi ignorado, contudo, uma estudo sobre a melhoria da precisão e exatidão ao se considerar esses efeitos pode ser interessante.
3. Estudar a aplicação de outros protocolos de comunicação, como o HTTP, para verificar a melhora ou piora do tempo de resposta do sistema.
4. O sistema proposto assume que todas as toras de madeira devem possuir uma seção reta convexa. Seria importante realizar abordagens que sejam capaz de considerar superfícies não convexas.
5. Estudar a implementação de outras abordagens que não sejam a visão computacional clássica, como por exemplo o uso de redes neurais convolucionais, a fim de detectar possíveis melhoras no desempenho da segmentação e impactos no tempo de processamento.
6. Avaliar a melhora do sistema em um caso onde o *hardware* é preparado para uma aplicação industrial.
7. Construir uma interface intuitiva que permita fácil configuração do sistema.
8. Integrar os resultados gerados pelo sistema com um banco de dados para facilitar a consulta e análise da produção.

Referências Bibliográficas

- AGIN, G. J. Computer vision systems for industrial inspection and assembly. *Computer*, IEEE Computer Society, v. 13, n. 05, p. 11–20, 1980. 32
- AMAT ALÍCIA CASALS (AUTH.), P. C. T. e. J. *Computer Vision: Theory and Industrial Applications*. 1. ed. [S.l.]: Springer-Verlag Berlin Heidelberg, 1992. ISBN 978-3-642-48677-7,978-3-642-48675-3. 32
- APPLICATIONS of Laser Sensors. 2021. <<https://www.waycon.biz/products/laser-sensors/applications-laser-sensors/>>. 35
- BENECKE Máquinas. 2021. <<https://www.benecke.com.br/>>. 18, 19
- BOCALON, B. A. S. et al. Utilização da madeira desde os primórdios da humanidade até a contemporânedade. *Anais do Simpósio de Sustentabilidade e Contemporaneidade nas Ciências Sociais*, v. 2, n. 1, 2014. 14
- CIPOLLA, R.; BATTIATO, S.; FARINELLA, G. M. *Computer Vision: Detection, recognition and reconstruction*. [S.l.]: Springer, 2010. v. 285. 23
- CORKE, P. I.; KHATIB, O. *Robotics, vision and control: fundamental algorithms in MATLAB*. [S.l.]: Springer, 2011. v. 73. 23, 25, 28, 29, 30, 31, 50
- CÂMERA IP: Cuide do seu bebê com o ESP32-CAM. 2021. <<https://www.filipeflop.com/blog/esp-32-camera-ip/>>. 44, 53
- FORSYTH, D.; PONCE, J. *Computer vision: A modern approach*. [S.l.]: Prentice hall, 2011. 23, 24, 25, 31
- HILLAR, G. C. *MQTT Essentials-A lightweight IoT protocol*. [S.l.]: Packt Publishing Ltd, 2017. 35, 36
- HUNTER Torno Laminador. 2021. <<https://www.hunterprojetos.com.br/principal>>. 15
- KAKANI, V. et al. A critical review on computer vision and artificial intelligence in food industry. *Journal of Agriculture and Food Research*, Elsevier, v. 2, p. 100033, 2020. 32, 33
- KILPELÄ, A.; PENNALA, R.; KOSTAMOVAARA, J. Precise pulsed time-of-flight laser range finder for industrial distance measurements. *Review of Scientific Instruments*, American Institute of Physics, v. 72, n. 4, p. 2197–2202, 2001. 34
- KUROSE, J.; ROSS, K. *Computer networks: A top down approach featuring the internet*. *Peorsoim Addison Wesley*, 2010. 35
- MADEIRA, D. *Protocolo I²C - Comunicação entre Arduinos*. 2021. <<https://portal.vidadesilicio.com.br/i2c-comunicacao-entre-arduinos/>>. 36
- MARTYNENKO, A. Computer-vision system for control of drying processes. *Drying Technology*, Taylor & Francis, v. 24, n. 7, p. 879–888, 2006. 23, 30

- MATTOS, R. L. G.; GONÇALVES, R. M.; LACERDA, F. d. C. Painéis de madeira no brasil: panorama e perspectivas. Banco Nacional de Desenvolvimento Econômico e Social, 2008. 14
- MORGAN, M. *Development of a novel time-of-flight laser rangefinder for opto-mechatronic applications*. Tese (Doutorado) — Memorial University of Newfoundland, 2020. 34
- NUSSENZVEIG, H. M. *Curso de física básica: Ótica, relatividade, física quântica (vol. 4)*. [S.l.]: Editora Blucher, 2014. 23, 25
- PAREDES, G. M. Study of mooring systems for offshore wave energy converters. 2016. 26
- POPOV, V.; OSTAREK, M.; TENISON, C. Practices and pitfalls in inferring neural representations. *NeuroImage*, Elsevier, v. 174, p. 340–351, 2018. 28
- POPOVIC, M. *Communication protocol engineering*. [S.l.]: CRC press, 2018. 35
- RASPBERRY Pi 3 Model B Documentation. 2021. <<https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>>. 54, 55
- SANQUETTA, C. R. et al. Produção, importação, exportação e consumo aparente de painéis de madeira no brasil entre 1961 e 2016. *BIOFIX Scientific Journal*, v. 5, n. 1, p. 44–49, 2019. 14
- SHERLOCK. 2021. <<https://www.infaimon.com/pt-pt/producto/sherlock/>>. 15
- SIEGWART, R.; NOURBAKHSI, I. R.; SCARAMUZZA, D. *Introduction to autonomous mobile robots*. [S.l.]: MIT press, 2011. 23, 50
- STIVANELLO, M. E.; MARCELLINO, K. J. A machine vision system for online metal can-end rivet inspection. In: IOP PUBLISHING. *Journal of Physics: Conference Series*. [S.l.], 2019. v. 1335, n. 1, p. 012002. 33, 34
- SZELISKI, R. *Computer vision: algorithms and applications*. [S.l.]: Springer Science & Business Media, 2010. 23, 24, 25, 27
- (US), F. P. L. *Wood handbook: wood as an engineering material*. [S.l.]: The Laboratory, 1987. 14
- VILLALBA-DIEZ, J. et al. Deep learning for industrial computer vision quality control in the printing industry 4.0. *Sensors*, Multidisciplinary Digital Publishing Institute, v. 19, n. 18, p. 3987, 2019. 32
- VL53L0X Documentation. 2021. <<https://www.st.com/en/imaging-and-photonics-solutions/vl53l0x.html#documentation>>. 54
- WÖHLER, C. *3D computer vision: efficient methods and applications*. [S.l.]: Springer Science & Business Media, 2012. 23, 25