



INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E  
TECNOLOGIA DO AMAZONAS  
CAMPUS MANAUS DISTRITO INDUSTRIAL  
ENGENHARIA DE CONTROLE E AUTOMAÇÃO



LUCAS PAIVA MONTEIRO

DETECÇÃO DE DEFEITOS EM PLACAS DE CIRCUITO  
IMPRESSO APLICANDO VISÃO COMPUTACIONAL

MANAUS - AM  
2023

LUCAS PAIVA MONTEIRO

DETECÇÃO DE DEFEITOS EM PLACAS DE CIRCUITO  
IMPRESSO APLICANDO VISÃO COMPUTACIONAL

Trabalho de Conclusão de Curso apresentado como requisito parcial à obtenção do título de Bacharel em Engenharia de Controle e Automação, do Instituto Federal de Educação, Ciência e Tecnologia do Amazonas, Campus Manaus Distrito Industrial – IFAM/CMDI.

Orientador: Prof. Dr. Alyson de Jesus dos Santos

MANAUS - AM  
2023

## ANEXO 7

### ATA DE DEFESA PÚBLICA DO TRABALHO DE CONCLUSÃO DE CURSO

Aos 17 dias do mês de Fevereiro, de 2023, às 11h, o(a) discente Lucas Paiva Monteiro apresentou o seu Trabalho de Conclusão de Curso para avaliação da Banca Examinadora constituída pelos seguintes integrantes: Prof(a). Dr. Alyson de Jesus dos Santos (docente-orientador), Prof(a).Msc. Jaidson Brandão da Costa (Membro 1) e Prof(a). Gabriel Pinheiro Compto (Membro 2). A sessão publica de defesa foi aberta pelo(a) presidente da banca, que apresentou a Banca Examinadora e deu continuidade aos trabalhos, fazendo uma breve referência ao TCC, que tem como titulodetecção de defeitos em placas de circuito impresso aplicando visão computacional. Na sequência, o(a) discente teve até 30 minutos para a comunicação oral de seu trabalho. Cada integrante da banca examinadora fez suas arguições após a defesa do mesmo. Ouvidas as explicações do(a) discente, a banca examinadora, reunida em caráter sigiloso, para proceder à avaliação final, deliberou e decidiu pela APROVAÇÃO com média final 9,0 (Nove)

do referido trabalho.

Foi dada ciência ao(à) discente que a versão final do trabalho deverá ser entregue até o dia 16/03/2023, com as devidas alterações sugeridas pela banca. Nada mais havendo a tratar, a sessão foi encerrada às 12h15 min, sendo lavrada a presente ata, que, uma vez aprovada, foi assinada por todos os membros da Banca Examinadora e pelo(a) discente.

Prof.(a) Orientador(a)/Presidente: Alyson de Jesus dos Santos  
Prof.(a) Avaliador 1: Jaidson Brandão da Costa  
Prof.(a) Avaliador 2: Gabriel Pinheiro Compto  
Discente: Lucas Paiva Monteiro

### Dados Internacionais de Catalogação na Publicação (CIP)

M757d Monteiro, Lucas Paiva.  
Detecção de defeitos em placas de circuitos impresso aplicando  
visão computacional / Lucas Paiva Monteiro. — Manaus, 2023.  
50f.: il. color.

Monografia (Graduação) — Instituto Federal de Educação,  
Ciência e Tecnologia do Amazonas, *Campus* Manaus Distrito  
Industrial, Curso de Engenharia de Controle e Automação, 2023.  
Orientador: Prof.º Alyson de Jesus dos Santos, Dr.

1. Visão computacional. 2. Identificação de padrões. 3.  
Qualidade na indústria 4.0. 4. Identificação de falha de solda. I.  
Santos, Alyson de Jesus. II. Instituto Federal de Educação, Ciência  
e Tecnologia do Amazonas. III. Título.

CDD 629.895

Elaborada por Oziane Romualdo de Souza (CRB11/ nº 734)



*Este trabalho é dedicado a minha família, e em memória ao RAPASTELSON...*

## **AGRADECIMENTOS**

Palavras não conseguem expressar os sentimentos de alegria pela conclusão de uma jornada tão importante quanto esta. Agradeço aos meus pais por todo apoio que me deram ao longo desta estrada, sem eles nada seria possível.

Agradeço a Deus por me dar força para superar cada um dos obstáculos que apareceram durante o caminho e sabedoria para completar cada desafio apresentado durante o curso.

Agradeço ao Polo de Inovação do IFAM por me dar suporte e a experiência necessária durante o curso, pelo acesso a materiais para realização desta pesquisa e por sempre confiarem no meu trabalho, com agradecimento especial ao professor Fabio Lima e Mitsuo Takeno.

Agradeço também aos meus amigos de faculdade Ícaro, Hylbert, Janderson, Pedro Neto, Rebecca, Yure, entre outros que me acompanharam durante toda esta caminhada e sempre estiveram ali para me auxiliar nas dificuldades.

A Secretaria do Curso, pela cooperação.

Agradeço ao meu orientador, Prof. Dr. Alyson de Jesus dos Santos, pela sabedoria e paciência com que me orientou, sendo bastante acessível para finalizar a jornada, e por sempre estar disposto a me ajudar.

*“Não vale a pena viver sonhando e se esquecer de viver.”  
(, Alvo Dumbledore)*

## RESUMO

Este trabalho consiste no desenvolvimento de um sistema de visão capaz de identificar falhas em placas de circuito impresso. O sistema é composto por uma câmera de alta velocidade de 2MP, um controlador da câmera para receber a imagem e aplicar ferramentas de visão computacional, um CLP para tratamento das informações da câmera e uma IHM para servir de interface para o operador do teste permitindo um acompanhamento dos resultados e o controle da trigger da câmera e reset das informações pós turno de trabalho. O sistema de visão é composto pela câmera e o controlador e trabalha aplicando ferramentas de visão computacional, neste trabalho utilizamos uma ferramenta de identificação de padrões com autoaprendizagem que nos permitiu a partir de um grupo de imagens obter um grau de qualidade maior nos resultados do teste.

Palavras-chave: Visão Computacional; Identificação de padrões; Qualidade na Indústria 4.0; Identificação de Falha de Solda

## **ABSTRACT**

This work consists of the development of a vision system capable of identifying faults in printed circuit boards. The system consists of a 2MP high-speed camera, a camera controller to receive the image and apply computational vision tools, a PLC to process camera information and an HMI to serve as an interface for the test operator, allowing a monitoring of results and control of the camera trigger and reset of post-shift information. The vision system is composed of the camera and the controller and works applying computer vision tools, in this work we use a self-learning pattern identification tool that allowed us to obtain a higher quality degree in the test results from a group of images.

Keywords: Computer vision; pattern identification; Quality in Industry 4.0; Solder Fault Identification

## LISTA DE ILUSTRAÇÕES

Figura 1 – Processo evolutivo da industria . . . . .	18
Figura 2 – Topologia para uma rede Ethernet industrial. . . . .	20
Figura 3 – Processo comparativo entre a visão humana e a visão computacional . . . . .	20
Figura 4 – Passos fundamentais em processamento de imagens digitais. . . . .	21
Figura 5 – Esquema básico de um CLP. . . . .	23
Figura 6 – Ciclo de Varredura de um CLP. . . . .	24
Figura 7 – Diagrama elétrico de uma partida direta. . . . .	25
Figura 8 – Diagrama elétrico de uma partida direta. . . . .	26
Figura 9 – Interface Homem-máquina. . . . .	27
Figura 10 – Posto de teste das placas . . . . .	28
Figura 11 – Amostra de teste . . . . .	29
Figura 12 – Diagrama de funcionamento do projeto . . . . .	30
Figura 13 – Fonte de alimentação . . . . .	31
Figura 14 – Tabela de características da fonte de alimentação . . . . .	31
Figura 15 – Controlador CV - X420F - KEYENCE. . . . .	32
Figura 16 – Tela de seleção das ferramentas do Software CV - X420F. . . . .	32
Figura 17 – Câmera CA - 200C - KEYENCE. . . . .	33
Figura 18 – Cabo de comunicação para câmera. . . . .	33
Figura 19 – Lente CA - LHR8 - KEYENCE. . . . .	34
Figura 20 – Câmera de testes completa. . . . .	34
Figura 21 – Catálogo câmera . . . . .	35
Figura 22 – Switch 8 portas . . . . .	35
Figura 23 – Cabo RJ45. . . . .	36
Figura 24 – CLP CompactLogix L30ERM. . . . .	36
Figura 25 – Interface inicial do Studio5000 . . . . .	37
Figura 26 – IHM Altus X2 Base 10 V2 . . . . .	37
Figura 27 – Tela inicial do IX Developer . . . . .	38
Figura 28 – Setup de testes . . . . .	39
Figura 29 – Seleção da ferramenta de visão . . . . .	39
Figura 30 – 1º Etapa da configuração . . . . .	40
Figura 31 – 2º Etapa da configuração . . . . .	40
Figura 32 – 3º Etapa da configuração . . . . .	41
Figura 33 – 4º Etapa da configuração . . . . .	41
Figura 34 – Seleção do protocolo de comunicação . . . . .	42
Figura 35 – Dados que serão enviados para o CLP via Ethernet IP . . . . .	42
Figura 36 – Monitor de dados recebidos . . . . .	43
Figura 37 – Monitor de dados enviados . . . . .	43

Figura 38 – Teste de placa padrão OK . . . . .	44
Figura 39 – Teste de placa padrão NG . . . . .	44
Figura 40 – Importando o modulo para Stuido5000 . . . . .	45
Figura 41 – DADOS RECEBIDOS PELO CLP . . . . .	46
Figura 42 – Programa de tratamento dos dados . . . . .	46
Figura 43 – DADOS RECEBIDOS PELO CLP . . . . .	47
Figura 44 – Tela inicial da IHM . . . . .	47
Figura 45 – Tela de testes da IHM . . . . .	48
Figura 46 – Tela de tags da IHM . . . . .	48
Figura 47 – Tela da IHM pós teste OK . . . . .	49
Figura 48 – Tela da IHM pós teste NG . . . . .	49

## LISTA DE TABELAS

Tabela 1 – Elementos Básicos em Ladder . . . . .	26
Tabela 2 – Lista de componentes do projeto . . . . .	28
Tabela 3 – Lista de endereços IP do sistema . . . . .	35



## LISTA DE ABREVIATURAS E SIGLAS

CLP	Controladores lógico programáveis
CPU	Unidade Central de Processamento (Central Processing Unit)
E/S	Entradas/Saídas
IHM	Interface homem máquina
IP	Internet protocol
NG	Not Good
OK	OK
PCB	Módulos de Circuito de Proteção (Protection Circuit Modules)
PCM	Módulos de Circuito de Proteção (Protection Circuit Modules)

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>15</b>
<b>1.1</b>	<b>JUSTIFICATIVA</b>	<b>16</b>
<b>1.2</b>	<b>OBJETIVOS</b>	<b>16</b>
<b>1.3</b>	<b>OBJETIVOS ESPECIFICOS</b>	<b>17</b>
<b>1.4</b>	<b>METODOLOGIA</b>	<b>17</b>
<b>1.5</b>	<b>ESTRUTURA DO TCC</b>	<b>17</b>
<b>2</b>	<b>REFERENCIAL TEÓRICO</b>	<b>18</b>
<b>2.1</b>	<b>INDUSTRIA 4.0</b>	<b>18</b>
<b>2.2</b>	<b>PROTOCOLO DE COMUNICAÇÃO ETHERNET</b>	<b>19</b>
<b>2.3</b>	<b>VISÃO COMPUTACIONAL</b>	<b>20</b>
<b>2.4</b>	<b>CONTROLADORES LÓGICO PROGRAMAVEIS (CLP)</b>	<b>22</b>
2.4.1	PRINCIPIO DE FUNCIONAMENTO	23
2.4.2	CLASSIFICAÇÃO DOS CLPs	25
2.4.3	LADDER	25
<b>2.5</b>	<b>INTERFACE HOMEM-MÁQUINA (IHM)</b>	<b>27</b>
<b>3</b>	<b>MATERIAIS E METODOS</b>	<b>28</b>
<b>3.1</b>	<b>COMPONENTES</b>	<b>28</b>
<b>3.2</b>	<b>POSTO DE TESTE</b>	<b>28</b>
<b>3.3</b>	<b>AMOSTRAS DE TESTE</b>	<b>29</b>
<b>3.4</b>	<b>DIAGRAMA DE FUNCIONAMENTO PROJETO</b>	<b>29</b>
<b>3.5</b>	<b>FONTE DE ALIMENTAÇÃO</b>	<b>31</b>
<b>3.6</b>	<b>CONTROLADOR CV - X420F</b>	<b>31</b>
3.6.1	CÂMERA KEYENCE CA - 200C	33
3.6.1.1	CABO DE COMUNICAÇÃO PARA CÂMERA	33
3.6.2	LENTE CA - LHR8	33
3.6.3	CÂMERA DE TESTES COMPLETA	34
<b>3.7</b>	<b>SWITCH 8 PORTAS</b>	<b>35</b>
3.7.1	CABO RJ 45	36
<b>3.8</b>	<b>CLP CompactLogix L30ERM</b>	<b>36</b>
3.8.1	Studio5000	37
<b>3.9</b>	<b>IHM X2 BASE 10 V2</b>	<b>37</b>
<b>3.10</b>	<b>IX - Developer</b>	<b>38</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÕES</b>	<b>39</b>
<b>4.1</b>	<b>SETUP DOS TESTES</b>	<b>39</b>

<b>4.2</b>	<b>ESCOLHA DA FERRAMENTA DE VISÃO</b>	<b>39</b>
4.2.1	CONFIGURAÇÃO DA FERRAMENTA DE VISÃO	40
4.2.2	FAZENDO SETUP DOS DADOS DE SAÍDA	41
4.2.3	MONITORANDO OS DADOS DE SAÍDA	42
4.2.4	VALIDAÇÃO DO SISTEMA DE VISÃO	44
<b>4.3</b>	<b>CONFIGURANDO CLP</b>	<b>45</b>
4.3.1	DADOS RECEBIDOS PELO CLP	45
4.3.2	PROGRAMA LADDER PARA TRATAMENTO DOS DADOS	46
4.3.3	DADOS DE SAIDA PARA IHM	46
<b>4.4</b>	<b>TELAS DA IHM</b>	<b>47</b>
4.4.1	TAGS DA IHM	48
4.4.2	TELAS DA IHM COM RESULTADOS ALCANÇADOS	48
<b>5</b>	<b>CONCLUSÃO</b>	<b>50</b>
	<b>REFERÊNCIAS</b>	<b>51</b>

## 1 INTRODUÇÃO

A visão computacional é uma forma de inteligência artificial, na qual podemos programar sistemas para reconhecer e extrair dados do ambiente. As imagens capturadas por estes sistemas são submetidas a ferramentas desenvolvidas para extrair as informações mais relevantes do ambiente, com isso podemos observar o meio em que estes sistemas estão integrados de uma maneira artificial, porém com olhos mais atentos por submetermos este ambiente a ferramentas computacionais extremamente precisas e específicas.

Construir algoritmos para que computadores possam enxergar o ambiente em que ele está inserido e extrair informações de objetos e locais, a princípio pode parecer tarefas complexas que necessitam de profundos conhecimentos de programação e de intensos conhecimentos matemáticos. Porém os sistemas de visão computacional estão cada vez mais acessíveis para todos os níveis de conhecimento, podendo ser usado em aplicações, construídos desde entusiastas até as gigantes da indústria (XAVIER, 2022).

As ferramentas de visão computacional estão se tornando cada vez mais essenciais na indústria, pois com o avanço das tecnologias a qualidade de produtos tem se tornado imprescindível, as câmeras podem fazer inspeções muito mais rápidas e precisas do que pessoas. Além da vantagem de precisão e velocidade os sistemas de visão tornaram-se bem mais compactos ao longo dos anos isso faz com que possam ser instalados diversos postos de trabalho necessitando apenas de alguém para fazer a manutenção destes sistemas, diminuindo assim o número de pessoas em uma equipe.

A detecção de falhas em placa de circuitos impresso tem se tornado um ponto muito importante no crescimento da indústria 4.0, pois, os componentes estão ficando cada vez menores, e isso vai adicionando falhas ao fator humano em testes deste tipo, mesmo com microscópios como atualmente estes testes são realizados em diversos lugares da indústria, mesmo com feramente específicas a tarefa de encontrar falhas nestes componentes é bastante maçante e seria bem melhor executada por um programa de computador tanto pela ergonomia do operador quanto pela qualidade e velocidade de execução.

Este Trabalho de Conclusão de Curso (TCC) delimitou-se em colher informações sobre colher as informações de como um sistema de detecção de falhas em placas de circuito impresso poderia ser aplicável na indústria 4.0 atualmente, visando buscar resultados em melhoria de qualidade e melhoria em ambientes de trabalho com atividades maçantes para pessoas.

## 1.1 JUSTIFICATIVA

Durante um período de trabalho em uma fábrica de baterias foi verificado a ocorrência de diversas peças aprovadas com defeitos de solda nos conectores das placas PCM. Ao fim da linha existia um posto responsável pela verificação manual com auxílio de um microscópio digital destes conectores, porém mesmo com esta verificação o número de peças aprovadas com defeito era discrepante.

Com isso verificou-se a necessidade de automatizar esta verificação, pois concluiu-se que a atividade se tornava maçante ao decorrer do turno de trabalho, com isso o operador responsável pela verificação destes conectores acabava por permitir a passagem de peças defeituosas.

Neste trabalho foi desenvolvido um sistema de verificação da integridade de placas PCB por meio de visão computacional, a aquisição da imagem das placas foi feita por meio de uma câmera de alta velocidade de 2MP integrada a um software de visão computacional da KEYENCE, por meio de uma ferramenta desenvolvida no software CVX – 400 foi possível verificar falhas de integridade e assim retornar valores de OK para placas dentro do padrão e NG para placas defeituosas, através dos dados obtidos utilizamos um CLP CompactLogix L30ERM para tratar os dados e envia-los para uma IHM que será destinada ao operador, para que ele possa acompanhar a validação das placas e caso haja falha possa retirar este produto e direciona-lo a bancada técnica para reparo.

Este trabalho tem como proposta capturar imagens das placas e por meio de visão computacional identificar as possíveis falhas existentes, evitando assim que produtos defeituosos possam ser aprovados por falha humana.

Neste contexto, a pergunta que norteou a pesquisa foi “É possível por meio de visão computacional criar um posto de testes que retire operados de atividades maçantes para um ser humano e torne mais efetivo a processo de análise deste produto?”.

## 1.2 OBJETIVOS

Sendo assim, o presente trabalho de conclusão de curso (TCC) ao apresentar um sistema de inspeção que busca melhorar a qualidade do processo identificando com mais eficiência as amostras defeituosas, e permitindo ao operador deste sistema visualizar um histórico de produção, e com isso utilizar os dados do obtidos para acompanhar o índice de produtos defeituosos e buscar soluções para este problema.

Neste contexto, o objetivo deste trabalho é projetar e desenvolver um sistema capaz de identificar os produtos com falhas e informar ao operador este status, e permitir ao usuário do sistema acompanhar o histórico de sua produção.

### 1.3 OBJETIVOS ESPECIFICOS

Para obter um resultado satisfatório listamos os seguintes objetivos específicos:

- a) Modelar uma ferramenta capaz de identificar falhas nas placas a serem testadas.
- b) Realizar a captura das imagens das placas.
- c) Enviar os resultados para o CLP para tratamento dos dados retirados do sistema de visão computacional.
- d) Realizar o tratamento dos dados no Studio5000.
- e) Enviar os dados tratados para a IHM.
- f) Criar uma interface que receba os dados do teste e mostre de maneira clara para o operador do sistema.

### 1.4 METODOLOGIA

A metodologia respeita as características da pesquisa aplicada, onde os dados foram coletados em ambiente laboratorial, e os testes com hardware foram pensados em uma implementação industrial. Sendo assim foram utilizados todos os conhecimentos de visão computacional para desenvolvimento do sistema e validação da melhor forma possível, todos os resultados foram testados inúmeras vezes para que as falhas tendessem a zero, e toda pesquisa foi catalogada ao longo deste trabalho.

Referindo-se ainda a pesquisa aplicada, (GERHARDT; SILVEIRA, 2009) definem aquela cujo objetivo é o de “gerar conhecimentos para aplicação prática, dirigidos à solução de problemas específicos. Envolve verdades e interesses locais.” O que se aplica a nossa pesquisa, pois toda informação catalogada aqui tem por objetivo solucionar um problema pratico proposto anteriormente e de grande relevância para a indústria no seu estado atual.

### 1.5 ESTRUTURA DO TCC

O restante deste trabalho irá apresentar alguns elementos de grande importância para a descrição da proposta e dos resultados obtidos. No segundo capítulo serão apresentados elementos de referência para validação da proposta, elementos que servem de base para criarmos uma boa fundamentação. No terceiro capítulo serão abordados os materiais e métodos utilizados para o desenvolvimento e aplicação do sistema. No quarto capítulo serão apresentados os resultados obtidos com o sistema proposto, e por fim no capítulo cinco será apresentada uma discussão final dos resultados e do projeto como um todo, como forma de concluir a ideia inicialmente proposta.

## 2 REFERENCIAL TEÓRICO

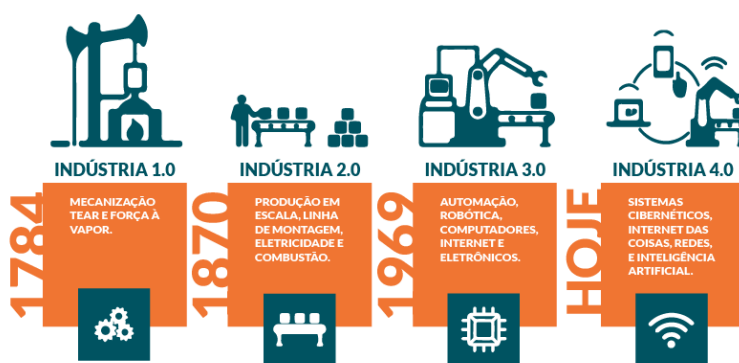
Este capítulo tem por objetivo apresentar conceitos teóricos necessários para as discussões práticas deste TCC.

### 2.1 INDÚSTRIA 4.0

A quarta revolução industrial ou indústria 4.0 está alterando a forma como trabalhamos e nos relacionamos com o meio em que vivemos. Esta era da digitalização está nos desafiando a experimentar coisas que nunca experimentamos antes, seus impactos dependerão de muitos fatores que ainda não foram plenamente mensurados. No campo organizacional, estamos vivendo profundas mudanças em escala, alcance e complexidade do processo produtivo, com base nas tecnologias cada vez mais incorporadas ao trabalho. Os desafios desta nova fase da revolução industrial irão fomentar o desenvolvimento de soluções que envolvam diversos stakeholders incluindo: a política global, setores públicos e privados, academia e sociedade em geral (SCHWAB, 2017).

Assim, destaca-se que a valorização da aprendizagem contínua precisa estar contida na cultura organizacional (SENGE, 2012), ainda mais se for considerado que a verdadeira vantagem competitiva está na capacidade e velocidade de aprendizagem das pessoas (DAVENPORT; PRUSAK, 1998) e que na sociedade da informação sobrevivem as organizações que conseguem melhor gerir seus conhecimentos (FILHO, 2000).

Figura 1 – Processo evolutivo da industria



Fonte:(SILVEIRA, 2016b)

Com isto podemos observar que com o evoluir da revolução industrial cada vez mais se fez necessária a mão de obra especializada, isso acabou se tornando bastante exigente, pois com o avanço das tecnologias esta cada vez mais difícil se manter especializado para o mercado de trabalho na indústria. Os polos industriais vêm cada vez mais buscando maneiras de capacitar seus técnicos e engenheiros para que não precisem buscar novos

funcionários que entendam das novas tecnologias, isto é essencial principalmente quando falamos de automação que é uma área onde não existe estagnação e constantemente estão aparecendo novos dispositivos ou técnicas para tornar os sistemas cada vez mais completos e menos dependentes das pessoas.

## **2.2 PROTOCOLO DE COMUNICAÇÃO ETHERNET**

A Ethernet/IP foi desenvolvida a partir do Common Industrial Protocol (CIP), que é um conjunto de padrões abertos que todas as empresas de automação devem manter. As tecnologias Ethernet/IP e CIP são gerenciadas pela ODVA (Open DeviceNet Vendors Association), uma organização global de comércio e desenvolvimento de padrões fundada em 1995 com mais de 300 membros corporativos. A rede ethernet/IP é um dos principais protocolos industriais nos Estados Unidos e é amplamente utilizado em diversos setores, incluindo fábrica, híbrido e processos. (VIEIRA, 2022)

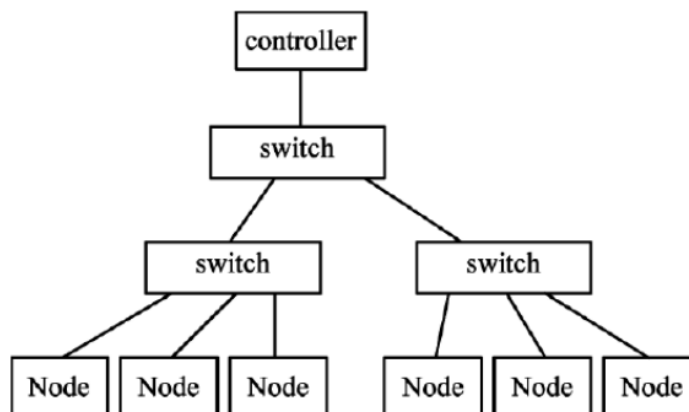
Ethernet é a rede local de tecnologia mais popular e utilizada nos ambientes tanto residenciais quanto comerciais. Alguns dos motivos são a rapidez, facilidade de implantação e adaptadores economicamente viáveis sendo que a maioria se não todos os computadores têm uma interface Ethernet incorporada. Sendo que a rede Ethernet consegue entregar a indústria uma tecnologia de rede uniforme, que deriva desde o chão de fábrica ao sistema gerencial do processo produtivo. Dessa forma os dados das áreas de produção e maquinário podem ser visualizados e monitorados por toda a rede da empresa. (DECOTIGNIE, 2019)

Um grande exemplo do desenvolvimento das redes Ethernet como tecnologia de comunicação e controle é seu meio físico, onde tem-se aplicações específicas a depender do uso e necessidade. Ambientes industriais necessitam de uma maior resistência física classificadas no IP (Ingress Protection), em português classe de proteção ou proteções de entrada dos componentes, a medida em que deve ter o maior nível de confiabilidade possível. Ao contrário do ambiente residencial onde o valor elevado devido a especificação técnica dos componentes tornaria inviável a aplicação. (PROFINET, 2018)

O protocolo de comunicação Ethernet IP como visto é um dos tipos de protocolo que vem ganhando bastante espaço nas indústrias e residências, e atualmente já é o principal protocolo de comunicação utilizado na integração de sistemas de automação, isso se dá por sua simplicidade de implementação e manutenção. A quantidade de materiais sobre Ethernet IP na internet faz com que ele se torne um dos grandes favoritos para escolha de um projeto, pois já existem inúmeras aplicações feitas com este protocolo o que torna o desenvolvimento de projeto mais dinâmico, tanto para desenvolvedores experientes quanto para os novatos na área de redes.



Figura 2 – Topologia para uma rede Ethernet industrial.

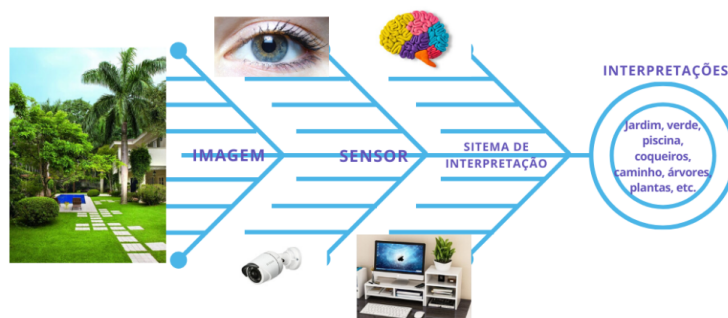


Fonte:(COUTO, 2010)

### 2.3 VISÃO COMPUTACIONAL

A visão computacional tem se tornado cada vez mais importante na indústria 4.0, pois com o avanço da tecnologia a eletrônica está ficando cada vez mais compacta, o que dificulta uma análise visual de componentes e solda eficiente por seres humanos. Com isso os sensores de visão conseguem a partir dos padrões definidos pela empresa manter uma qualidade impecável para que seu produto esteja apto a competir no mercado e gere assim o menor número de descartes por falha humana num processo tão delicado como a análise final de placas eletrônicas.

Figura 3 – Processo comparativo entre a visão humana e a visão computacional



Fonte:(NIEBLES; LI., 2016)

Um sistema de visão industrial é um conjunto de técnicas visuais de controle de qualidade, geralmente constituídos por uma câmera, iluminação, filtros e um software. Todo o conjunto desses equipamentos são utilizados para verificar se determinado produto está dentro dos padrões necessários para a aprovação e coletar dados da produção. Os sistemas

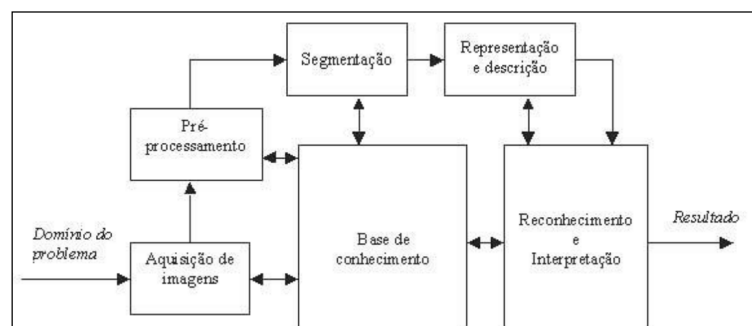
de visão são excelentes para assegurar uma inspeção detalhada e com um alto padrão para controle de falhas e erros de fabricação. (BAUMGARTEN, 2018)

Para (BARELLI, 2018), a visão computacional enquanto Ciência, pode ser entendida como aquela que estuda e desenvolve tecnologias que permitem que máquinas sejam capazes de extrair dados do ambiente ao redor, através de imagens capturadas por sensores de aquisição.

Os sistemas de visão são muito importantes para garantir a qualidade do produto e a credibilidade da marca. Com eles, não só a inspeção, mas toda a cadeia produtiva pode ficar mais eficiente, competitiva e confiável. (BAUMGARTEN, 2018)

Basicamente, um sistema de visão computacional envolve as etapas de aquisição de imagens, pré-processamento, segmentação, representação e descrição e reconhecimento (GONZALEZ; WOODS et al., 2002), como ilustrado na Figura 3.

Figura 4 – Passos fundamentais em processamento de imagens digitais.



Fonte:(GONZALEZ; WOODS et al., 2002)

Segundo (GONZALEZ; WOODS et al., 2002), os passos mostrados a Figura 3 podem ser descritos da seguinte forma:

- **Aquisição de imagens:** refere-se à forma em que a imagem é adquirida, seja por meio de scanner ou uma câmera, trabalhando on-line ou off-line. Nos experimentos realizados neste trabalho a aquisição é feita on-line por uma webcam. (GONZALEZ; WOODS et al., 2002)

- **Pré-processamento:** tem como objetivo melhorar a qualidade da imagem de forma a aumentar as chances de sucesso nas próximas etapas de processamento. Pode-se citar como exemplo a transformação de uma imagem RGB contendo uma amostra de grãos em uma outra em níveis de cinzas, com base nas informações das cores dos grãos. (GONZALEZ; WOODS et al., 2002)

- **Segmentação:** é realizada para separar da imagem apenas os fragmentos representando os objetos de interesse para a análise, como por exemplo, a segmentação de grãos. Neste trabalho, a função da segmentação é separar apenas os grãos contidos na imagem, para posterior classificação e detecção de defeitos. Na maioria das abordagens de segmentação descritas nos trabalhos encontrados na literatura inicialmente executa-se

a simplificação das imagens, normalmente por meio de binarização, para depois fazer a extração das características. O principal problema dessa forma de segmentação é que caso ocorra alguma falha na binarização, todas as etapas restantes podem ser comprometidas. (GONZALEZ; WOODS et al., 2002)

- Representação e descrição: envolve a extração de características dos objetos, formando um descritor, utilizado na etapa reconhecimento e interpretação. Cita-se como exemplos a extração de informações de cor de um grão para determinação da sua classe ou a extração de características geométricas para determinar se o grão está inteiro ou quebrado. (GONZALEZ; WOODS et al., 2002)

- Reconhecimento e interpretação: o reconhecimento é o processo que atribui um rótulo ao objeto, com base em um conjunto de informações previamente fornecidas pelo descritor. A interpretação envolve também a atribuição de um significado a um conjunto de objetos reconhecidos. Podese citar como exemplo a determinação da classe de grão (carioca, mulato ou preto), com base nas informações de cor previamente adquiridas e armazenadas. (GONZALEZ; WOODS et al., 2002)

- Base de conhecimento: contém o conhecimento adquirido e armazenado sobre o problema. Por exemplo, a informação do tamanho médio dos grãos, utilizada tanto na etapa de segmentação quanto na detecção do defeito “quebrado”. (GONZALEZ; WOODS et al., 2002)

Apesar de serem necessárias diversas etapas para implementação de um sistema de visão computacional, a indústria 4.0 está caminhando com bastante foco neste rumo, alguns exemplos são as empresas do ramo automobilístico e de autopeças, as empresas farmacêuticas, entre muitas outras empresas que vem investindo em tecnologias para melhoria da qualidade de seus produtos.

## **2.4 CONTROLADORES LÓGICO PROGRAMÁVEIS (CLP)**

Os CLPs ou Controladores Lógico Programáveis podem ser definidos, segundo a norma ABNT, como um equipamento eletrônico-digital compatível com aplicações industriais. Os CPLs também são conhecidos como PLCs, do inglês: Programmable Logic Controller. O primeiro CLP data de 1968 na divisão de hidráulicos da General Motors. Surgiu como evolução aos antigos painéis elétricos, cuja lógica fixa tornava impraticável qualquer mudança extra do processo. A tecnologia dos CLPs só foi possível com o advento dos chamados Circuitos Integrados e da evolução da lógica digital. (SILVA, 2007)

O Controlador Lógico Programável, CLP, é um dispositivo que possui uma função de computador industrial, nele são armazenadas instruções no qual serão utilizadas para um processo de controle, como por exemplo: instruções de sequência lógica, temporização, contagem. O CLP manipula dados, realiza operações lógicas e aritméticas e comunicação em rede. O CLP também pode ser definido como um equipamento eletrônico digital no qual

é compatível com processos industriais. Outra definição é que o CLP é um aparelho no qual utiliza uma memória programável para armazenamento de instruções, implementando funções específicas por meio de módulos de entradas e saídas. (SILVA, 2020)

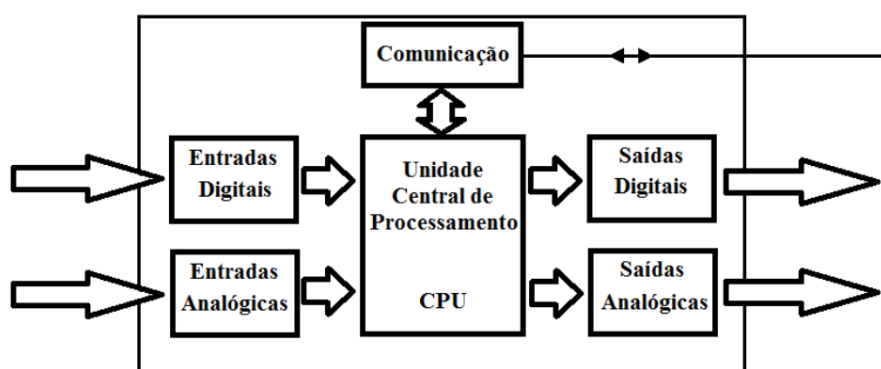
Os CLP'S são elementos robustos geralmente utilizados para aplicações industriais e trazem como vantagem para sua implementação os seguintes pontos:

- Facilidade de diagnóstico durante o projeto.
- Economia de espaço devido ao seu tamanho geralmente compacto.
- Podem ser programados sem a necessidade de interromper a produção.
- Necessita apenas de alguém especializado para sua manutenção
- Pode ser expandido com novos blocos de E/S, Comunicação, Fontes, entre outros elementos que podem ser adicionados como módulos de expansão.
- Capacidade de se comunicar com diversos outros equipamentos e com diversas técnicas de comunicação.

Os CLPs possuem características ideais para a indústria, principalmente a facilidade que apresentam em sua implementação, enquanto para montagem de outros sistemas é necessário a presença de uma equipe grande, para implementar um sistema com um CLP basta uma pessoa especializada que entenda tanto de CLP quanto de linguagem Ladder que ela será capaz de desenvolver um grande sistema. Para a indústria 4.0 isso se torna fundamental tanto pelo gasto em mão de obra quanto em tecnologia.

Para os sistemas de visão computacional geralmente podemos utilizar os CLPs pela quantidade de protocolos de comunicação que eles podem trabalhar e por sua capacidade de processamento, que os permite trabalhar com a quantidade de dados que são enviados de um sistema de visão computacional.

Figura 5 – Esquema básico de um CLP.



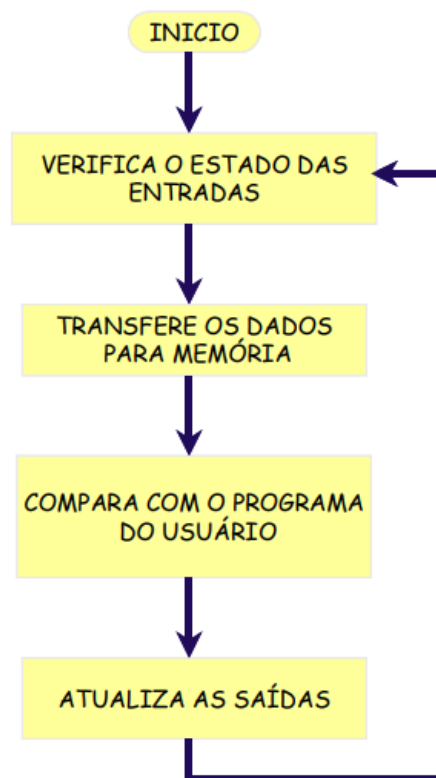
Fonte:(MARQUES, 2017)

#### 2.4.1 PRINCIPIO DE FUNCIONAMENTO

Conforme a Figura 6 abaixo, o CLP funciona de forma seqüencial, fazendo um ciclo de varredura em algumas etapas. É importante observar que quando cada etapa do ciclo é

executada, as outras etapas ficam inativas. O tempo total para realizar o ciclo é denominado CLOCK. Isso justifica a exigência de processadores com velocidades cada vez mais altas. (SILVA, 2007)

Figura 6 – Ciclo de Varredura de um CLP.



Fonte:(SILVA, 2007)

**Início:** Verifica o funcionamento da C.P.U, memórias, circuitos auxiliares, estado das chaves, existência de um programa de usuário, emite aviso de erro em caso de falha. Desativa todas as as saídas. (SILVA, 2007)

**Verifica o estado das entradas:** Lê cada uma das entradas, verificando se houve acionamento. O processo é chamado de ciclo de varredura. (SILVA, 2007)

**Compara com o programa do usuário:** Através das instruções do usuário sobre qual ação tomar em caso de acionamento das entradas o CLP atualiza a memória imagem das saídas. (SILVA, 2007)

**Atualiza as saídas:** As saídas são acionadas ou desativadas conforme a determinação da CPU. Um novo ciclo é iniciado. (SILVA, 2007)

## 2.4.2 CLASSIFICAÇÃO DOS CLPs

Os CLPs podem ser divididos em algumas categorias sendo elas:

Nano e micro CLPs: possuem até 16 entradas e a saídas. Normalmente são compostos por um único módulo com capacidade de memória máxima de 512 passos. (SILVA, 2007)

CLPs de médio porte: capacidade de entrada e saída em até 256 pontos, digitais e analógicas. Permitem até 2048 passos de memória. (SILVA, 2007)

CLPs de grande porte: construção modular com CPU principal e auxiliares. Módulos de entrada e saída digitais e analógicas, módulos especializados, módulos para redes locais. Permitem a utilização de até 4096 pontos. A memória pode ser otimizada para o tamanho requerido pelo usuário. (SILVA, 2007)

## 2.4.3 LADDER

A linguagem Ladder foi a primeira que surgiu na programação dos Controladores Lógico Programáveis (CLPs), pois sua funcionalidade procurava imitar os antigos diagramas elétricos, utilizados pelos Técnicos e Engenheiros da época. O objetivo era o de evitar uma quebra de paradigmas muito grande, permitindo assim a melhor aceitação do produto no mercado. (SILVA, 2007)

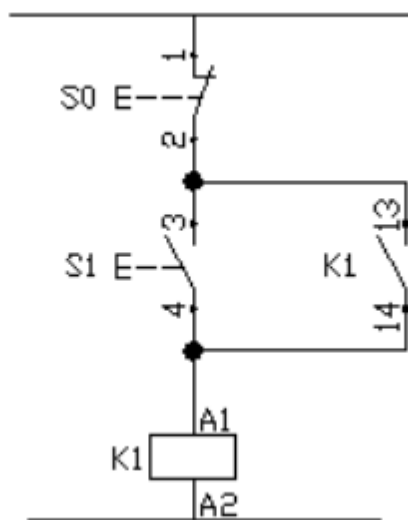
O diagrama de contatos (Ladder) consiste em um desenho formado por duas linhas verticais, que representam os pólos positivo e negativo de uma bateria, ou fonte de alimentação genérica. Entre as duas linhas verticais são desenhados ramais horizontais que possuem chaves. Estas podem ser normalmente abertas, ou fechadas e representam os estados das entradas do CLP. Dessa forma fica muito fácil passar um diagrama elétrico para linguagem Ladder. (SILVA, 2007)

Figura 7 – Diagrama elétrico de uma partida direta.



Fonte:(SILVA, 2007)

Figura 8 – Diagrama elétrico de uma partida direta.



Fonte:(SILVA, 2007)

A partir do exemplo dado podemos notar alguns exemplos que acabam sendo bastante comuns nos diagramas Ladder em geral, na Tabela 1 estão representados alguns destes elementos:

Tabela 1 – Elementos Básicos em Ladder

Nomeclatura	Abreviação	Símbolo
Contato Normalmente Aberto	NA	
Contato Normalmente Fechado	NF	
Bobina ou Saída	-	

Fonte:(SILVA, 2007)

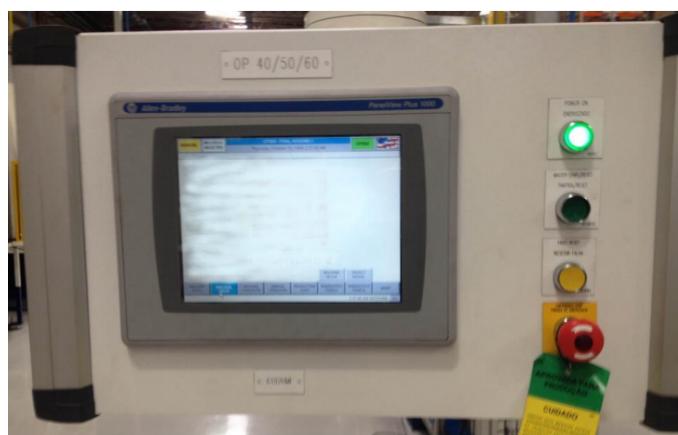
A linguagem Ladder é fundamental para quem deseja desenvolver projetos de automação com CLPs, com ela é possível transformar circuitos que fisicamente seriam gigantes com quadros de comando em simples lógicas de programação para desempenhar as mesmas atividades. Em ambientes de desenvolvimentos de CLPs podemos encontrar lógicas muito comuns como “Lógica AND”, “Lógica OR”, “Lógica XOR”, entre muitas outras, isso também torna o Ladder uma linguagem bem mais simplificada, pois utilizamos de elementos já bastante conhecidos da eletrônica digital.

A linguagem Ladder conhecida como diagrama de blocos torna a programação de CLPs bastante compacta, pois podemos desenvolver sistemas inteiros e transformá-los em blocos configuráveis por E/S.

## 2.5 INTERFACE HOMEM-MÁQUINA (IHM)

Uma IHM (Interface Homem Máquina) é a tradução do termo HMI (Human Machine Interface) e como o próprio nome diz é uma interface gráfica que permite o homem interagir com a máquina. As interfaces IHM de mercado variam bastante, podendo ser aplicadas desde no controle de plantas nucleares até na tela de um Iphone (Isso mesmo, um APP no Iphone que permite a interface com a máquina é uma HMI) (SILVEIRA, 2016a).

Figura 9 – Interface Homem-máquina.



Fonte:(SILVEIRA, 2016a)

As IHMs para controle de processos de manufatura, são aplicadas a um controle centralizado aplicado à linhas de produção e máquinas industriais. Neste caso, tais interfaces normalmente são equipadas por receitas de modelos e processos, registros de eventos, alarmes, e outras informações de máquina, processo ou produção que necessite de acesso instantâneo (SILVEIRA, 2016a).

Para que uma linha de manufatura possa ser equipada com uma IHM, primeiramente é necessário que exista um CLP que coleta a informação dos sensores, a transforma em álgebra booleana, para que então a HMI possa conectar no CLP e “ler” a informação a fim de mostrar a mesma de forma visual e também possibilitando uma melhor tomada de decisões (SILVEIRA, 2016a).

As IHMs são muito utilizadas nas máquinas da indústria para setup e supervisão, mas também podem ser utilizadas em testes automatizados para que o usuário faça um acompanhamento do teste ou mude alguma configuração para rodar novos produtos. Na indústria o controle de qualidade da produção é algo fundamental então registrar os dados dos testes em IHMs é algo que pode ser fundamental para melhoria das linhas de produção, pois assim a engenharia de produto pode acompanhar diariamente o status da produção e buscar solução para os produtos defeituosos caso estejam em grande quantidade.



### 3 MATERIAIS E METODOS

Este capítulo tem por objetivo apresentar os materiais que foram necessários para o desenvolvimento e implementação sistema, assim como as técnicas que foram adotadas para solucionar todos os problemas propostos e atingir o resultado desejados.

#### 3.1 COMPONENTES

Na Tabela 2 estão listados todos os componentes que compõem o sistema proposto.

Tabela 2 – Lista de componentes do projeto

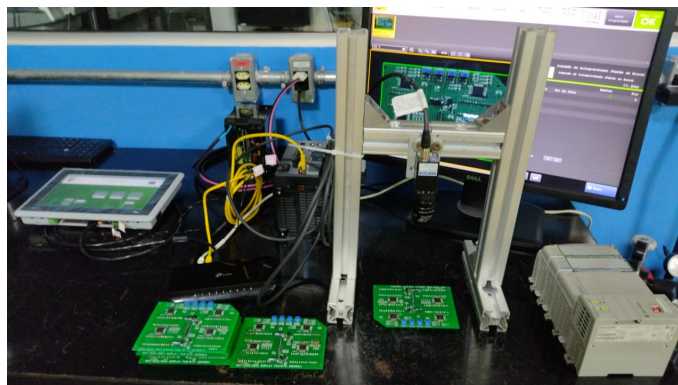
<b>COMPONENTES</b>	<b>QTD</b>
<b>Fonte de alimentação</b>	1
<b>Controlador CV - X420F</b>	1
<b>Câmera KEYENCE 2MP CV - X 200C</b>	1
<b>Lente KEYENCE 8mm LHR8</b>	1
<b>Cabo de comunicação para câmera</b>	1
<b>Switch 8 portas D - link</b>	1
<b>Cabo RJ45</b>	4
<b>CLP CompactLogix L30ERM</b>	1
<b>IHM Altus X2 Base 10 V2</b>	1

Fonte: O Próprio Autor (2023)

#### 3.2 POSTO DE TESTE

Na Figura 10 está mostrado o posto de teste das placas que foi pensado em uma bancada para um operador receber as amostras a serem testadas de uma esteira, no caso de ambientes industriais, e realizar os testes nas amostras com o sistema de visão.

Figura 10 – Posto de teste das placas



Fonte: O Próprio Autor (2023)

A estrutura de suporte da câmera foi pensada em perfil BOSH, para que o sistema de visão possa ser modificado de altura em futura necessidade, o sistema de visão foi pensado em uma altura fixa de 10 cm, com iluminação ambiente que estava adequada para a aplicação, porém a iluminação muda de acordo com o ambiente em que se integra um sistema de visão computacional, então isto se torna algo a ser trabalhado para outros ambientes de integração do sistema.

### 3.3 AMOSTRAS DE TESTE

A amostra utilizada como padrão para validação do projeto está apresentada na Figura 11.

Figura 11 – Amostra de teste



Fonte: O Próprio Autor (2023)

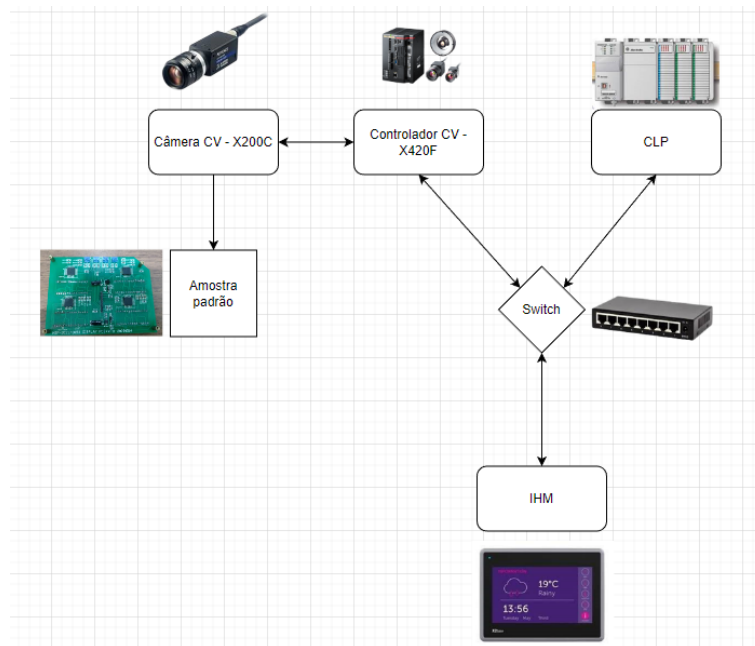
A placa escolhida para teste é um circuito de driver para displays de sete segmentos, após uma análise foi escolhido apenas um CI presente na amostra para que houvesse uma simplificação na hora de realizar os testes de visão computacional, tendo em vista que ao montar a ferramenta seria possível aumentar o número de perfis a serem avaliados. A ferramenta escolhida neste trabalho foi um identificador de padrões com autoaprendizagem, para utilizar esta ferramenta foi necessária a criação de um banco de imagens padrões, pois, isso aumenta a confiabilidade da ferramenta quanto está realizando os testes.

### 3.4 DIAGRAMA DE FUNCIONAMENTO PROJETO

A composição do sistema de visão computacional proposto neste TCC está mostrada na Figura 12. A câmera irá capturar a imagem das placas e enviar para seu software que vai aplicar uma ferramenta de verificação de padrões para detectar diferenças nas amostras, com isso os dados obtidos serão enviados para um CLP via Ethernet IP, após o recebimento dos dados, o programa em Ladder irá trabalhar o que foi recebido e transformar

em informações relevantes para enviar a IHM, após os dados serem tratados e enviados existirá uma tela de interação do operador que permitirá que o mesmo realize o trigger da câmera para tirar a foto e o sistema de visão computacional faça a análise e retorne os resultados de status do teste e contagem das peças, também será permitido ao operador na tela de testes a opção de fazer o RESET da contagem após o fim do turno de trabalho.

Figura 12 – Diagrama de funcionamento do projeto



Fonte: O Próprio Autor (2023)

- Câmera KEYENCE CV – X200C: Foi a câmera escolhida para este projeto principalmente pela quantidade de ferramentas disponíveis para análise, é uma câmera bastante utilizada nas empresas do distrito, possui 2 Megapixels e é uma câmera de alta velocidade.

- Controlador CV – X420F: É o controlador da câmera que faz a integração da hardware com os softwares, nele podemos integrar até duas câmeras e programar diversas ferramentas, o sistema possui diversos meios de comunicação e podemos enviar e receber dados de diferentes fontes.

- Controlador CompactLogix L30ERM: É o CLP escolhido para o desenvolvimento do projeto, ele permite expansões por módulos, então conseguimos adicionar E/S, módulos de rede, entre diversos outros. Assim como o controlador da câmera podemos comunicar este dispositivo com diferentes outros por muitos protocolos de comunicação, e assim trabalhar os dados recebidos para gerar informações relevantes para quem irá operar o teste.

- IMH Altus X2 BASE 10 V2: É a IHM escolhida para o desenvolvimento deste sistema por conta de sua simplicidade de integração e por conta de sua qualidade. Nas telas desenvolvidas serão exibidos dados relevantes de status do teste e contagem das placas que passaram por ele, mostrando a quantidade de placas que estão OK e a quantidade de

placas NG.

- Switch: Foi um dispositivo utilizado para integrar os diferentes dispositivos em rede via Ethernet, como o CLP não possuía uma expansão de portas ethernet foi necessária a adição deste componente ao projeto.

### 3.5 FONTE DE ALIMENTAÇÃO

Figura 13 – Fonte de alimentação



Fonte: O Próprio Autor (2023)

A fonte de alimentação foi escolhida de tal maneira que fosse capaz de atender a todos os componentes do projeto, as características da fonte estão mostradas na Figura 14.

Figura 14 – Tabela de características da fonte de alimentação

Power ratings	Input voltage	Output Voltage (VDC)	Output current *1	Model number
60 W	Single phase	24 V	2.5 A	S8VK-C06024
120 W	100 to 240 VAC	24 V	5 A	S8VK-C12024
240 W	90 to 350 VDC	24 V	10 A	S8VK-C24024
480 W		24 V	20 A	S8VK-C48024

Fonte: O Próprio Autor (2023)

### 3.6 CONTROLADOR CV - X420F

Controlador que serve de plataforma para o software de criação de ferramentas de visão computacional da KEYENCE, através dele podemos criar diversas ferramentas, executar as

mesmas na câmera, acompanhar e validar o funcionamento delas com a modo de execução e transferir os dados obtidos da câmera para diferentes plataformas por vários protocolos de comunicação.

Figura 15 – Controlador CV - X420F - KEYENCE.



Fonte: O Próprio Autor (2023)

Na Figura 16 esta mostrada a tela de seleção de ferramentas do software, podemos desenvolver inúmeros tipos de aplicações aqui, tais como, presença e ausência, ferramentas de dimensionamento de objetos, leitura de códigos 1D e 2D, verificação de macha, entre muitos outros tipos de ferramentas disponíveis.

A ferramenta que melhor atendeu a proposta do projeto foi a de identificação de padrões, com ela conseguimos determinar na placa pontos em que o software deveria trabalhar e buscar falhas, nesta ferramenta foi preciso adicionar uma quantidade de imagens para aprendizado, e a partir dai melhorar a busca por falhas do sistema.

Figura 16 – Tela de seleção das ferramentas do Software CV - X420F.



Fonte: O Próprio Autor (2023)

### 3.6.1 CÂMERA KEYENCE CA - 200C

Na Figura 17 está mostrada a base câmera selecionada para o projeto do sistema de visão, escolhemos esta câmera por sua simplicidade de implementação, principalmente por conta de seu tamanho compacto e por sua gama de ferramentas disponíveis.

Figura 17 – Câmera CA - 200C - KEYENCE.



Fonte: O Próprio Autor (2023)

#### 3.6.1.1 CABO DE COMUNICAÇÃO PARA CÂMERA

Na Figura 18 está mostrado o cabo de comunicação da câmera com o controlador.

Figura 18 – Cabo de comunicação para câmera.



Fonte: O Próprio Autor (2023)

### 3.6.2 LENTE CA - LHR8

Na Figura 18 está mostrada a lente da câmera para o sistema de visão, a lente possui uma abertura de 8mm e foi escolhida por conta desta característica que nos permitiu trabalhar com amostras pequenas como componentes eletrônicos e trilhas de solda.



Figura 19 – Lente CA - LHR8 - KEYENCE.



Fonte: O Próprio Autor (2023)

### 3.6.3 CÂMERA DE TESTES COMPLETA

A câmera completa possui um tamanho bastante compacto que se adaptou bem ao projeto, foi desenvolvido um suporte para que ela fosse fixada no posto de testes. A câmera irá ficar em posto fixo a uma altura de 10 cm das amostras, as amostras serão recebidas pelo operador e testadas manualmente para validação, o operador irá posicionar as placas no local de teste e apertar o botão de iniciar teste na IHM para que a câmera capture a imagem e envie para ferramenta no software.





Figura 20 – Câmera de testes completa.



Fonte: O Próprio Autor (2023)

Na Figura 21 está mostrada uma tabela com algumas características da câmera utilizada, tais como, resolução, tempo de transferência e coloração.

Figura 21 – Catálogo câmera

Câmera de 2 megapixels				
				
Modelo	CA-H200MX / CA-H200CX	CA-H200M / CA-H200C	CA-200M / CA-200C	CA-HS200M / CA-HS200C
Especificações	Monocromática de alto desempenho com velocidade de 16x / colorida de alto desempenho com velocidade de 16x * <sup>1</sup>	Monocromática resistente ao ambiente com velocidade de 16x / colorida resistente ao ambiente com velocidade de 16x * <sup>2</sup>	Monocromática resistente ao ambiente / colorida resistente ao ambiente * <sup>2</sup>	Monocromática compacta com velocidade de 16x / colorida compacta com velocidade de 16x
Intervalo de captura	1600 × 1200 pixels	1600 × 1200 pixels	1600 × 1200 pixels	1600 × 1200 pixels
Tempo de transferência	11,7 ms	11,8 ms	56,5 ms	14,2 ms

Fonte: Catálogo KEYENCE

### 3.7 SWITCH 8 PORTAS

O switch utilizado foi escolhido com o intuito de expandir as portas Ethernet do sistema, como não possuíamos uma expansão para o próprio CLP tivemos que adicionar um switch ao projeto, junto a isso foi escolhido o protocolo de comunicação Ethernet IP para os componentes do sistema, principalmente por sua simplicidade de implementação.

Figura 22 – Switch 8 portas



Fonte: O Próprio Autor (2023)

Tabela 3 – Lista de endereços IP do sistema

COMPONENTES	Endereço IP
PC	192.168.0.35
Controlador CV - X420F	192.168.0.17
CLP CompactLogix L30ERM	192.168.0.50
IHM Altus X2 Base 10 V2	192.168.0.37

Fonte: O Próprio Autor (2023)



### 3.7.1 CABO RJ 45

Um cabo Ethernet, ou cabo RJ-45, como também é chamado, é um cabo de rede usado para conexões com fio. Pode ser usado em diversos lugares.

Figura 23 – Cabo RJ45.



Fonte: O Próprio Autor (2023)

## 3.8 CLP CompactLogix L30ERM

O CLP utilizado neste TCC é um CompactLogix L30ERM, é um controlador da Rockwell que permite expansões de diversas tecnologias, tais como, E/S, ethernet, fonte, entre várias outras.

Este controlador foi utilizado para receber as informações da câmera e trabalhar elas para serem transformadas em dados relevantes para o operador, ao receber as informações o programa desenvolvido irá buscar dentro do modulo da câmera importado no software o código que descreve um teste como OK ou NG, ao identificar este código ele inicia a contagem do determinado tipo de resultado, e em ambos os casos soma ao total testado, a logica de RESET irá zerar estas contagens ao receberem o sinal enviado pela IHM e ao receber o sinal de iniciar testes as informações de contagem são enviadas para a IHM.

Figura 24 – CLP CompactLogix L30ERM.



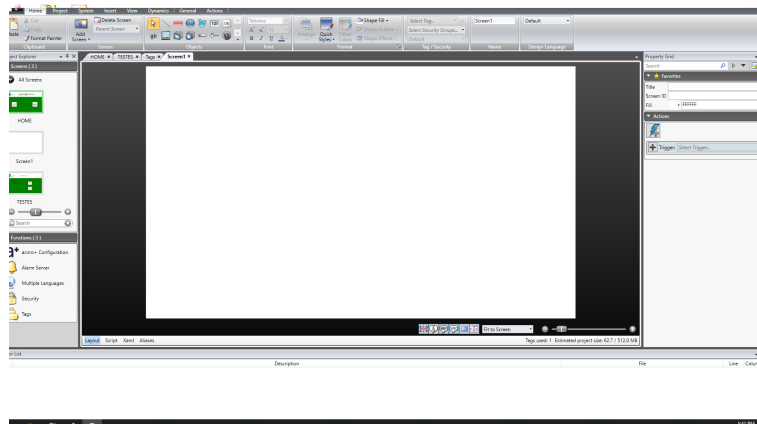
Fonte: O Próprio Autor (2023)



### 3.10 IX - Developer

O IX Developer é o software utilizado para desenvolvimento das telas de IHM, neste trabalho ele foi utilizado para desenvolvimento das telas que irão mostrar para o operador os dados do teste. A partir deste software foi possível desenvolver e configurar tudo referente ao Front end do sistema, tais como, transições entre botões, acionamento dos blocos de texto e mensagens necessárias para quem irá acompanhar os dados exibidos.

Figura 27 – Tela inicial do IX Developer



Fonte: O Próprio Autor (2023)

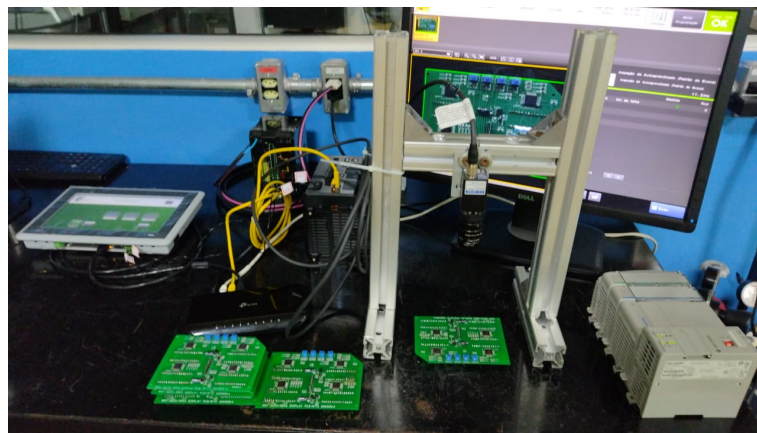
## 4 RESULTADOS E DISCUSSÕES

O objetivo deste trabalho foi o desenvolvimento de um sistema de visão computacional capaz de a partir de uma amostra conseguir identificar falhas em placas de circuito impresso em pontos planejados.

### 4.1 SETUP DOS TESTES

Na Figura 28 esta apresentada a montagem do setup de testes.

Figura 28 – Setup de testes

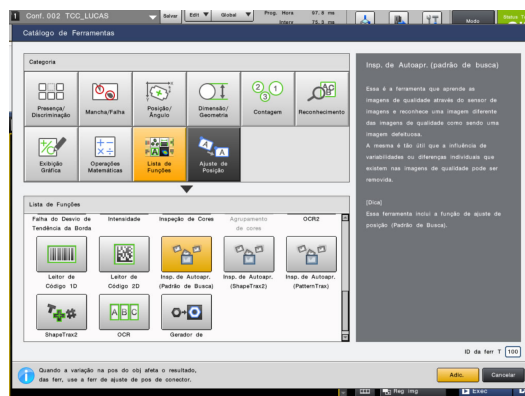


Fonte: O Próprio Autor (2023)

### 4.2 ESCOLHA DA FERRAMENTA DE VISÃO

Primeiramente foi realizado um estudo das ferramentas presentes no software e selecionada a que melhor atendesse a proposta do sistema de visão computacional, na Figura 29 está mostrada a ferramenta selecionada.

Figura 29 – Seleção da ferramenta de visão

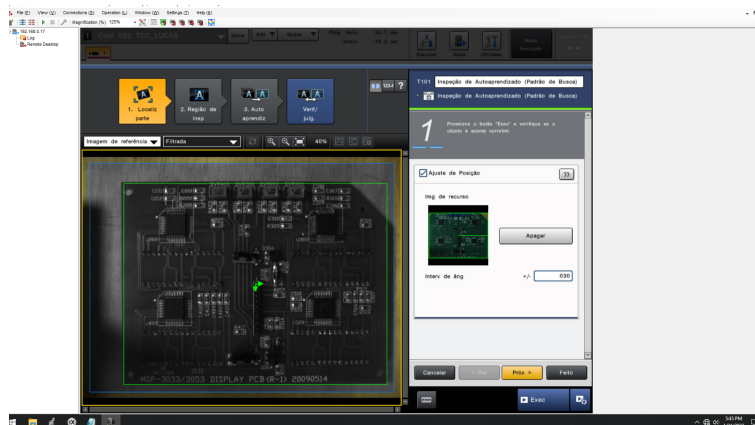


Fonte: O Próprio Autor (2023)

#### 4.2.1 CONFIGURAÇÃO DA FERRAMENTA DE VISÃO

A ferramenta escolhida para identificação das falhas no sistema de visão foi uma ferramenta que faz a detecção de padrões utilizando uma base de dados obtidas a partir das amostras das placas como referência de padrões Ok, na Figura 30 está mostrada a primeira etapa da configuração da ferramenta que é a seleção do perfil das placas, aqui selecionamos a placa inteira para que o sistema faça uma varredura completa na amostra.

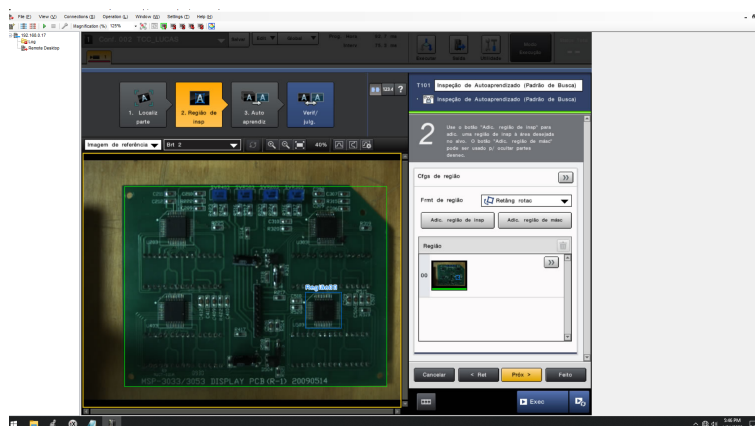
Figura 30 – 1º Etapa da configuração



Fonte: O Próprio Autor (2023)

Na Figura 31 está mostrada a segunda etapa que é a seleção do perfil nos quais serão identificadas as falhas, nesta fase da configuração selecionamos um circuito integrado da placa, a partir deste perfil conseguimos identificar se ele está dentro do padrão que foi estabelecido na próxima fase da configuração.

Figura 31 – 2º Etapa da configuração

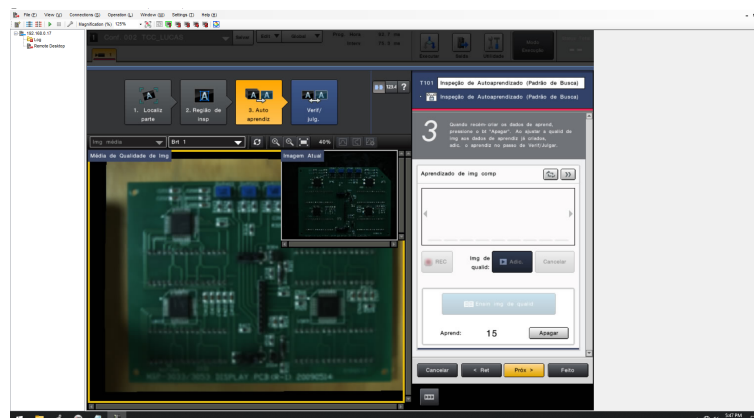


Fonte: O Próprio Autor (2023)

Na Figura 32 está mostrada a fase de seleção das amostras padrão para aprendizado da ferramenta, aqui selecionamos imagens para colocar dentro de um banco de imagens na

memória das ferramentas para que ela saiba identificar a partir destas amostras as placas que devem ser reconhecidas como amostras Ok.

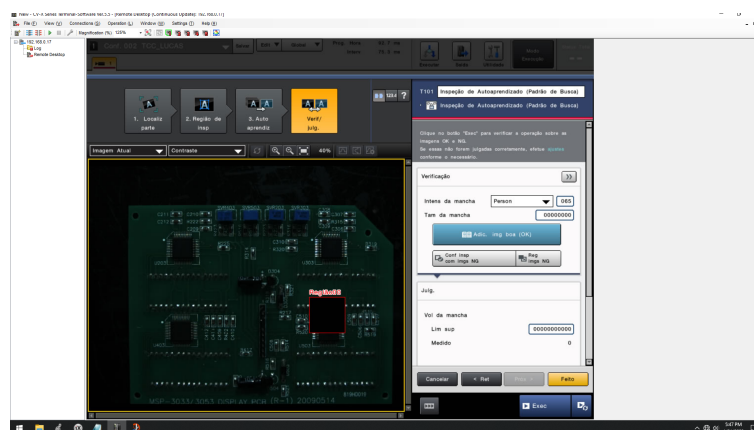
Figura 32 – 3º Etapa da configuração



Fonte: O Próprio Autor (2023)

Na Figura 33 está mostrada a última etapa da configuração da ferramenta de visão, aqui podemos pré-visualizar o julgamento da ferramenta, com isso podemos verificar se o banco de imagens padrão está de acordo com o esperado na identificação de falhas nos perfis selecionados.

Figura 33 – 4º Etapa da configuração

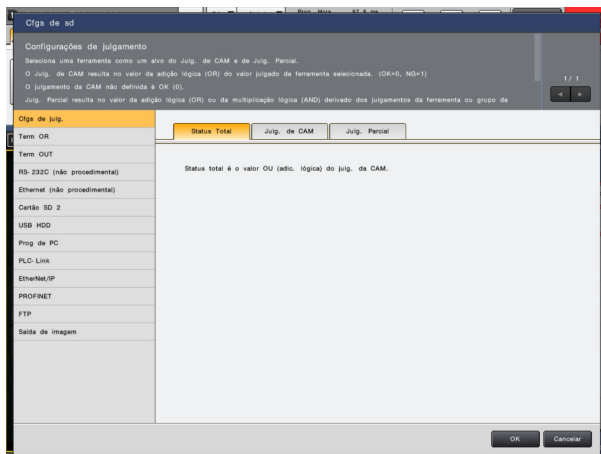


Fonte: O Próprio Autor (2023)

#### 4.2.2 FAZENDO SETUP DOS DADOS DE SAÍDA

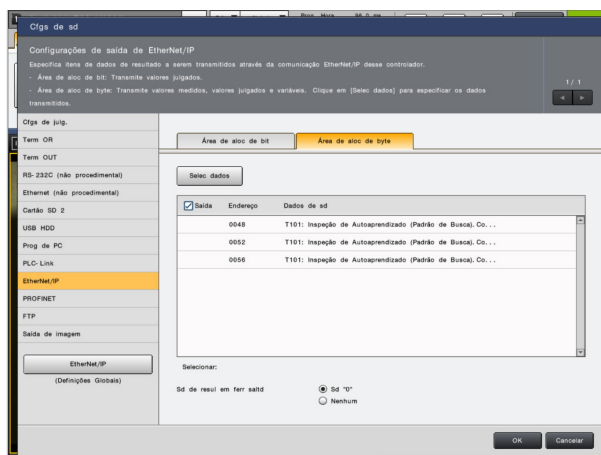
Após a captura e análise da imagem pela ferramenta de visão computacional, a próxima fase consiste em fazer o setup dos dados de saída, nesta fase selecionamos as informações relevantes que devem ser tratadas pelo CLP. Na Figura 34 estão mostrados os tipos de comunicação aceitados pelo software da câmera, e na Figura 35 estão mostradas as informações que foram selecionadas para serem enviadas para o CLP.

Figura 34 – Seleção do protocolo de comunicação



Fonte: O Próprio Autor (2023)

Figura 35 – Dados que serão enviados para o CLP via Ethernet IP



Fonte: O Próprio Autor (2023)

#### 4.2.3 MONITORANDO OS DADOS DE SAÍDA

Na Figura 36 e Figura 37 estão mostradas as telas de monitoramento dos dados de saída do sistema CV – X420F, na primeira podemos acompanhar os dados recebidos e na segunda os dados enviados pelo sistema. Aqui podemos acompanhar também o status da comunicação do sistema com dispositivos externos, neste trabalho estabelecemos uma comunicação cíclica com o CLP, isso nos permite enviar e receber informações e comandos em tempo real.

Nesta etapa podemos observar um número extenso de informações nas telas de dados recebidos e enviados pelo sistema, sendo assim, foi necessária a implementação de um CLP para receber os dados necessários e trabalhá-los de maneira que ficassem filtrados em informações relevantes para serem exibidas ao operador do teste.

Figura 36 – Monitor de dados recebidos

Lista de dados recb da EtherNet/IP			Lista de dados env da EtherNet/IP		
Área de aloc de bit			Área de aloc de byte		
Endereço	Valor	Descrições	Endereço	Valor	Descrições
0000:bit0	<input type="checkbox"/>	Command Request Flag	0012	0000	ExecCondNo
0000:bit1	<input type="checkbox"/>	(Reserv)	0014	0000	(Reserv)
0000:bit2	<input type="checkbox"/>	(Reserv)	0016	00000000	Command Number
0000:bit3	<input type="checkbox"/>	Result ACK Flag	0020	00000000	Command Parameter1
0000:bit4	<input type="checkbox"/>	Error Reset Request Flag	0024	00000000	Command Parameter2
0000:bit5	<input type="checkbox"/>	(Reserv)	0028	00000000	Command Parameter3
0000:bit6	<input type="checkbox"/>	(Reserv)	0032	00000000	Command Parameter4
0000:bit7	<input type="checkbox"/>	(Reserv)	0036	00000000	Command Parameter5
0001:bit0	<input type="checkbox"/>	TRG1	0040	00000000	Command Parameter6
0001:bit1	<input type="checkbox"/>	TRG2	0044	00000000	Command Parameter7
0001:bit2	<input type="checkbox"/>	TRG3	0048	00000000	Command Parameter8
0001:bit3	<input type="checkbox"/>	TRG4	0052	00000000	Command Parameter9
0001:bit4	<input type="checkbox"/>	(Reserv)	0056	00000000	Command Parameter10
0001:bit5	<input type="checkbox"/>	(Reserv)	0060	00000000	Command Parameter11
0001:bit6	<input type="checkbox"/>	(Reserv)	0064	00000000	Command Parameter12
0001:bit7	<input type="checkbox"/>	(Reserv)	0068	00000000	Command Parameter13

Comunicação: Ativ      Status da comunic cíclica: **Conectd**      Fechar

Fonte: O Próprio Autor (2023)

Figura 37 – Monitor de dados enviados

Lista de dados env da EtherNet/IP			Lista de dados recb da EtherNet/IP		
Área de aloc de bit			Área de aloc de byte		
Endereço	Valor	Descrições	Endereço	Valor	Descrições
0000:bit0	<input type="checkbox"/>	Command Complete Flag	0012	0000	Error Code
0000:bit1	<input type="checkbox"/>	Command Error Flag	0014	0000	(Reserv)
0000:bit2	<input checked="" type="checkbox"/>	Command Ready Flag	0016	00000027	Total Count
0000:bit3	<input type="checkbox"/>	Result Ready Flag	0020	00000000	Command Result
0000:bit4	<input type="checkbox"/>	Result OR Flag	0024	00000000	Command Data1
0000:bit5	<input type="checkbox"/>	(Reserv)	0028	00000000	Command Data2
0000:bit6	<input type="checkbox"/>	(Reserv)	0032	00000000	Command Data3
0000:bit7	<input type="checkbox"/>	(Reserv)	0036	00000000	Command Data4
0001:bit0	<input checked="" type="checkbox"/>	READY1	0040	00000000	Command Data5
0001:bit1	<input type="checkbox"/>	READY2	0044	00000000	Command Data6
0001:bit2	<input type="checkbox"/>	READY3	0048	00000000	Result Data1
0001:bit3	<input type="checkbox"/>	READY4	0052	00000000	Result Data2
0001:bit4	<input type="checkbox"/>	(Reserv)	0056	00000000	Result Data3
0001:bit5	<input type="checkbox"/>	(Reserv)	0060	00000000	Result Data4
0001:bit6	<input type="checkbox"/>	(Reserv)	0064	00000000	Result Data5
0001:bit7	<input type="checkbox"/>	(Reserv)	0068	00000000	Result Data6

Saída manual

Comunicação: Ativ      Status da comunic cíclica: **Conectd**      Fechar

Fonte: O Próprio Autor (2023)

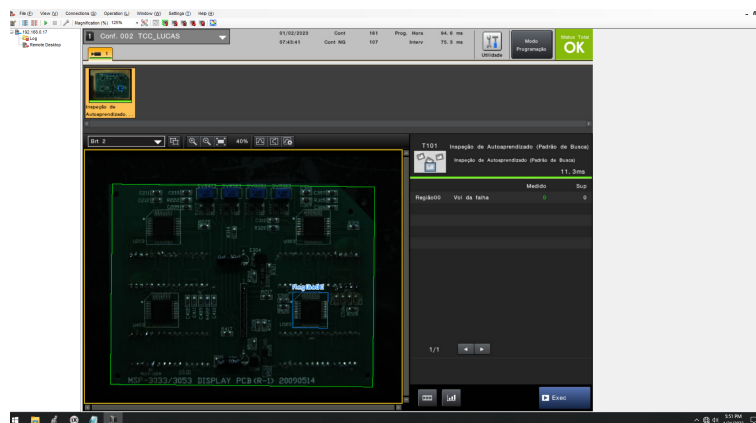


#### 4.2.4 VALIDAÇÃO DO SISTEMA DE VISÃO

Na Figura 38 e Figura 39 estão mostradas as telas de validação da ferramenta criada, isso foi feito utilizando as amostras padrão selecionadas justamente para esta etapa do projeto.

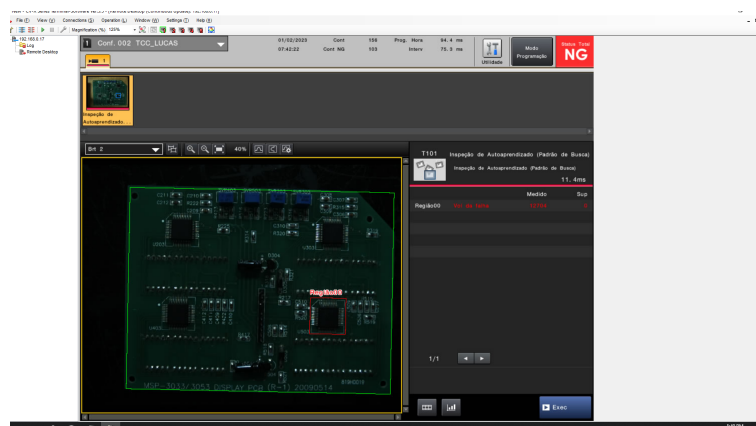
Nesta etapa do processo utilizamos um monitor conectado ao controlador da câmera para acompanhar o julgamento da ferramenta através de testes, com isso conseguimos validar o funcionamento do sistema de computacional. Na Figura 38 podemos verificar um teste Ok e na Figura 39 um teste NG, a próxima etapa do processo é fazer este acompanhamento pela IHM contemplada pelo sistema, para isso precisamos enviar os dados de julgamento para o CLP.

Figura 38 – Teste de placa padrão OK



Fonte: O Próprio Autor (2023)

Figura 39 – Teste de placa padrão NG

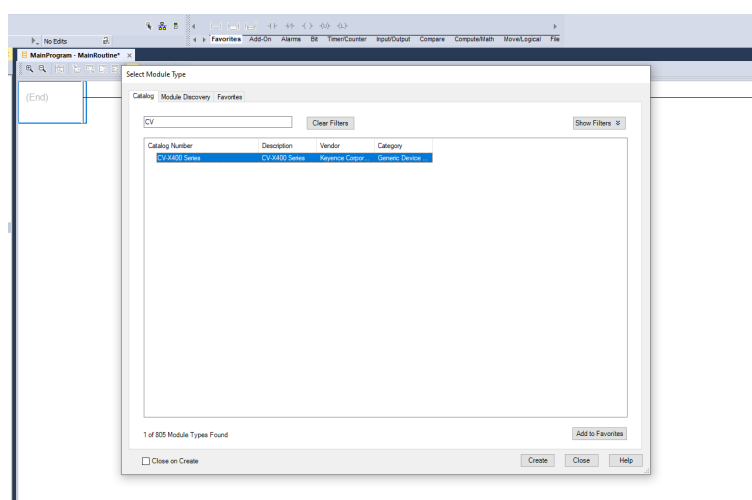


Fonte: O Próprio Autor (2023)

### 4.3 CONFIGURANDO CLP

Após a etapa de processamento da imagem pela ferramenta e envio dos dados via ethernet vamos para configuração do CLP, nesta fase primeiramente temos que adicionar o módulo da câmera no Studio5000 como mostrado na Figura 40 para que possamos estabelecer uma comunicação da câmera com o CLP.

Figura 40 – Importando o módulo para Studio5000



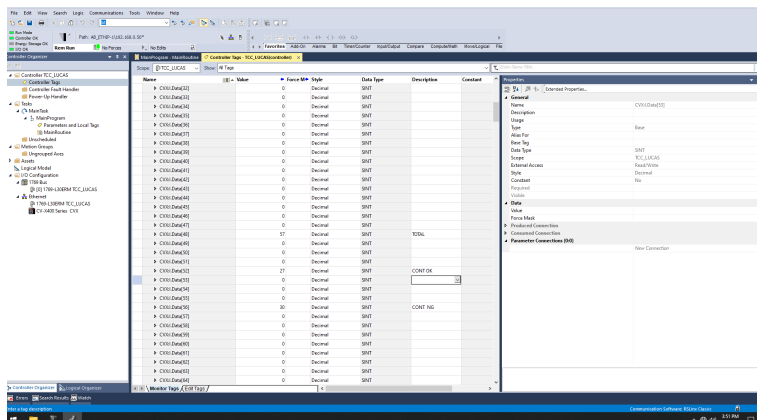
Fonte: O Próprio Autor (2023)

#### 4.3.1 DADOS RECEBIDOS PELO CLP

Configurado módulo no Studio5000 temos que verificar a comunicação do software do controlador com o CV – X420F, como mostrado na Figura 36 irá aparecer a mensagem de comunicação cíclica ativa, e na Figura 41 está mostrada a tela das tags onde podemos acompanhar os dados recebidos no CLP.

Na Figura 41 podemos observar que os dados recebidos no CLP são adicionados ao módulo da câmera no Studio5000, isso torna mais complicado o trabalho com estes dados, por conta disso foi elaborado um programa em Ladder que recebe estes dados e trabalha eles em informações visualmente mais claras para enviar a IHM.

Figura 41 – DADOS RECEBIDOS PELO CLP

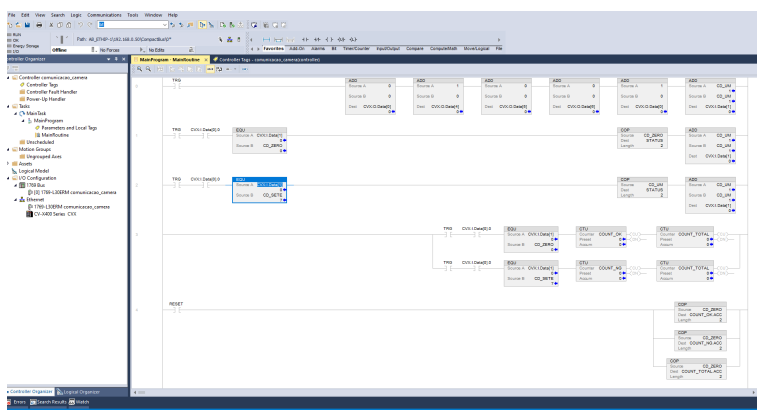


Fonte: O Próprio Autor (2023)

### 4.3.2 PROGRAMA LADDER PARA TRATAMENTO DOS DADOS

Após o recebimento dos dados se iniciou a etapa de estabelecer as necessidades da IHM, como o programa em Ladder deveria ser focado em exibir informações relevantes então estabelecemos quais dos dados precisariam ser trabalhados para se tornarem relevantes. Foi determinado que o programa deveria identificar e fazer a contagem de peças OK, NG e total de produção passados pelo teste, o programa também deveria ter a opção de RESET desta contagem para casos de trabalho em turnos, com isto estabelecido foi desenvolvido um programa que conseguiu atender a estas premissas e enviar as informações tratadas para a IHM, na Figura 42 está mostrado o programa Ladder desenvolvido para este trabalho.

Figura 42 – Programa de tratamento dos dados



Fonte: O Próprio Autor (2023)

### 4.3.3 DADOS DE SAIDA PARA IHM

Na Figura 43 estão mostrados as tags que se comunicam com a IHM e levam as informações tratadas para serem exibidas ao operador do teste.

Figura 43 – DADOS RECEBIDOS PELO CLP

Name	Face	M	S	Data Type	Description	Contact
CS_30V	1			Decimal	DI01	
CS_30M	1			Decimal	DI02	
CS_2000	0			Decimal	DI03	
CS_2000S	1,2	1,2		COUNTER		
CS_2000L	1,2	1,2		COUNTER		
CS_2000R	1,2	1,2		COUNTER		
CS_2000T	1,2	1,2		COUNTER		
CS_2000B	0			Decimal	DI04	
CS_2000C	0			Decimal	DI05	
CS_2000D	0			Decimal	DI06	
CS_2000E	0			Decimal	DI07	
CS_2000F	0			Decimal	DI08	
CS_2000G	0			Decimal	DI09	
CS_2000H	0			Decimal	DI10	
CS_2000I	0			Decimal	DI11	
CS_2000J	0			Decimal	DI12	
CS_2000K	0			Decimal	DI13	
CS_2000L	0			Decimal	DI14	
CS_2000M	0			Decimal	DI15	
CS_2000N	0			Decimal	DI16	
CS_2000O	0			Decimal	DI17	
CS_2000P	0			Decimal	DI18	
CS_2000Q	0			Decimal	DI19	
CS_2000R	0			Decimal	DI20	
CS_2000S	0			Decimal	DI21	
CS_2000T	0			Decimal	DI22	
CS_2000U	0			Decimal	DI23	
CS_2000V	0			Decimal	DI24	
CS_2000W	0			Decimal	DI25	
CS_2000X	0			Decimal	DI26	
CS_2000Y	0			Decimal	DI27	
CS_2000Z	0			Decimal	DI28	
CS_2000AA	0			Decimal	DI29	
CS_2000AB	0			Decimal	DI30	
CS_2000AC	0			Decimal	DI31	
CS_2000AD	0			Decimal	DI32	
CS_2000AE	0			Decimal	DI33	
CS_2000AF	0			Decimal	DI34	
CS_2000AG	0			Decimal	DI35	
CS_2000AH	0			Decimal	DI36	
CS_2000AI	0			Decimal	DI37	
CS_2000AJ	0			Decimal	DI38	
CS_2000AK	0			Decimal	DI39	
CS_2000AL	0			Decimal	DI40	
CS_2000AM	0			Decimal	DI41	
CS_2000AN	0			Decimal	DI42	
CS_2000AO	0			Decimal	DI43	
CS_2000AP	0			Decimal	DI44	
CS_2000AQ	0			Decimal	DI45	
CS_2000AR	0			Decimal	DI46	
CS_2000AS	0			Decimal	DI47	
CS_2000AT	0			Decimal	DI48	
CS_2000AU	0			Decimal	DI49	
CS_2000AV	0			Decimal	DI50	
CS_2000AW	0			Decimal	DI51	
CS_2000AX	0			Decimal	DI52	
CS_2000AY	0			Decimal	DI53	
CS_2000AZ	0			Decimal	DI54	
CS_2000BA	0			Decimal	DI55	
CS_2000BB	0			Decimal	DI56	
CS_2000BC	0			Decimal	DI57	
CS_2000BD	0			Decimal	DI58	
CS_2000BE	0			Decimal	DI59	
CS_2000BF	0			Decimal	DI60	
CS_2000BG	0			Decimal	DI61	
CS_2000BH	0			Decimal	DI62	
CS_2000BI	0			Decimal	DI63	
CS_2000BJ	0			Decimal	DI64	
CS_2000BK	0			Decimal	DI65	
CS_2000BL	0			Decimal	DI66	
CS_2000BM	0			Decimal	DI67	
CS_2000BN	0			Decimal	DI68	
CS_2000BO	0			Decimal	DI69	
CS_2000BP	0			Decimal	DI70	
CS_2000BQ	0			Decimal	DI71	
CS_2000BR	0			Decimal	DI72	
CS_2000BS	0			Decimal	DI73	
CS_2000BT	0			Decimal	DI74	
CS_2000BU	0			Decimal	DI75	
CS_2000BV	0			Decimal	DI76	
CS_2000BW	0			Decimal	DI77	
CS_2000BX	0			Decimal	DI78	
CS_2000BY	0			Decimal	DI79	
CS_2000BZ	0			Decimal	DI80	
CS_2000CA	0			Decimal	DI81	
CS_2000CB	0			Decimal	DI82	
CS_2000CC	0			Decimal	DI83	
CS_2000CD	0			Decimal	DI84	
CS_2000CE	0			Decimal	DI85	
CS_2000CF	0			Decimal	DI86	
CS_2000CG	0			Decimal	DI87	
CS_2000CH	0			Decimal	DI88	
CS_2000CI	0			Decimal	DI89	
CS_2000CJ	0			Decimal	DI90	
CS_2000CK	0			Decimal	DI91	
CS_2000CL	0			Decimal	DI92	
CS_2000CM	0			Decimal	DI93	
CS_2000CN	0			Decimal	DI94	
CS_2000CO	0			Decimal	DI95	
CS_2000CP	0			Decimal	DI96	
CS_2000CQ	0			Decimal	DI97	
CS_2000CR	0			Decimal	DI98	
CS_2000CS	0			Decimal	DI99	
CS_2000CT	0			Decimal	DI100	
CS_2000CU	0			Decimal	DI101	
CS_2000CV	0			Decimal	DI102	
CS_2000CW	0			Decimal	DI103	
CS_2000CX	0			Decimal	DI104	
CS_2000CY	0			Decimal	DI105	
CS_2000CZ	0			Decimal	DI106	

Figura 45 – Tela de testes da IHM

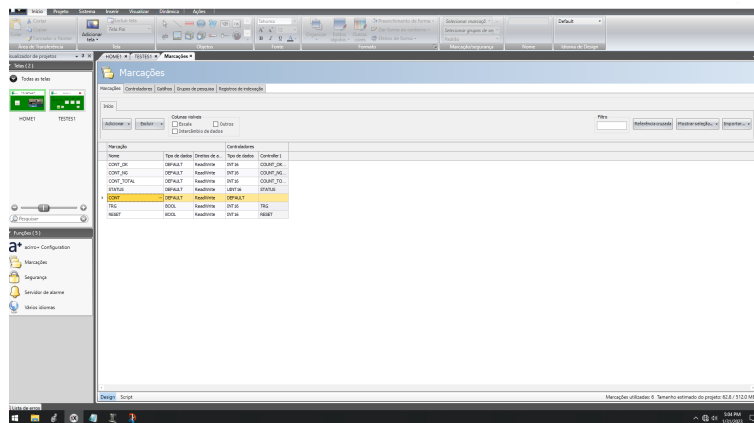


Fonte: O Próprio Autor (2023)

#### 4.4.1 TAGS DA IHM

Na Figura 46 podemos observar o sistema de tags utilizado no IX Developer, aqui nós vinculamos as tags da IHM com as do CLP através do mesmo nome, com isso o controlador consegue enviar e receber informações para as telas desenvolvidas.

Figura 46 – Tela de tags da IHM



Fonte: O Próprio Autor (2023)

#### 4.4.2 TELAS DA IHM COM RESULTADOS ALCANÇADOS

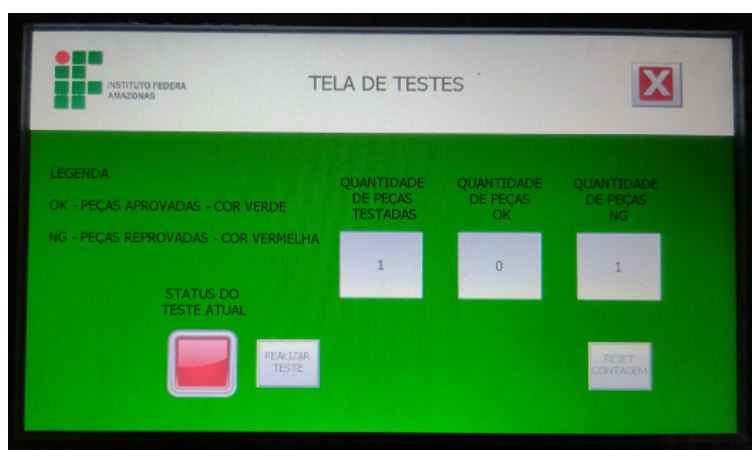
Na Figura 47 e Figura 48 estão mostrados os resultados alcançados nas telas da IHM, nelas podemos visualizar a contagem das peças aprovadas e reprovadas e a troca de cores do sinal de resultado dos testes.

Figura 47 – Tela da IHM pós teste OK



Fonte: O Próprio Autor (2023)

Figura 48 – Tela da IHM pós teste NG



Fonte: O Próprio Autor (2023)

## 5 CONCLUSÃO

Este TCC tem como principal objetivo desenvolver um sistema de detecção de falhas em componentes de placas de circuito impresso. Para atingir este objetivo foi utilizada uma câmera de alta velocidade de 2MP e com ela seu controlador que possui o software para aplicar ferramentas de visão computacional, para melhoria do sistema os dados obtidos foram levados para um CLP que tratou estes dados e os transformou em informações uteis e organizadas, e enviou os mesmos para serem exibidos em uma IHM.

Foram feitos testes nas amostras selecionadas com sistema desenvolvido, houve um grande grau de sucesso durante os testes do sistema, mas durante esta validação notou-se que como o sistema de visão esta fixo as placas precisam estar bem posicionadas para que haja uma boa leitura da câmera, e também foi possível notar que durante os testes como o protocolo de comunicação ethernet não é o mais adequado para este tipo de aplicação havia um certo delay na sincronia entre a leitura da câmera, o tratamento dos dados e o envio das informações em tempo real para a IHM.

Observando o cenário acima foi possível concluir que a confecção de um JIG para posicionamento fixo das placas poderia solucionar alguns problemas de falso negativo que ocorreram por conta de amostras mal posicionadas. E uma mudança no protocolo de comunicação para um mais adequado neste tipo de aplicação como por exemplo, ModBus, OPCUA, entre outros, poderia melhorar a velocidade de transmissão das informações.

O sistema desenvolvido mostrou-se uma excelente alternativa para melhoria da qualidade ao final de uma produção, pois com este sistema é possível substituir o fator humano de uma atividade bastante maçante, e identificar falhas de maneira mais eficiente aplicando visão computacional e pode ser um sistema a ser implementado indústria.

## REFERÊNCIAS

- BARELLI, F. *Introdução à Visão Computacional: Uma abordagem prática com Python e OpenCV*. [S.l.]: Editora Casa do Código, 2018.
- BAUMGARTEN, G. Sistema de visão industrial: descubra tudo o que eles podem fazer por você. Brasil, 2018.
- COUTO, R. T. Ethernet industrial. *Monografia (Graduação em Engenharia de*, 2010.
- DAVENPORT, T. H.; PRUSAK, L. Conhecimento empresarial: como as empresas gerenciam seu capital intelectual. *Rio de Janeiro: Campus*, 1998.
- DECOTIGNIE, J.-D. The many faces of industrial ethernet [past and present]. *IEEE Industrial Electronics Magazine*, IEEE, v. 1, n. 1, p. 8–19, 2019.
- FILHO, J. T. *Gerenciando conhecimento: como a empresa pode usar a memória organizacional e a inteligência competitiva no desenvolvimento dos negócios*. [S.l.]: Senac, 2000.
- GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de pesquisa*. [S.l.]: Plageder, 2009.
- GONZALEZ, R. C.; WOODS, R. E. et al. Digital image processing second edition. *Beijing: Publishing House of Electronics Industry*, v. 455, 2002.
- MARQUES, I. C. Desenvolvimento de aplicativo android para utilização de dispositivos móveis como interface homem máquina de um controlador industrial. Universidade Tecnológica Federal do Paraná, 2017.
- NIEBLES, J. C.; LI., F.-F. *Introduction to “Computer Vision*. [s.n.], 2016. Disponível em: [http://vision.stanford.edu/teaching/cs131\\_fall1617/lectures/lecture1\\_introduction\\_cs131\\_2016.pdf](http://vision.stanford.edu/teaching/cs131_fall1617/lectures/lecture1_introduction_cs131_2016.pdf). Acesso em: 19 Janeiro 2023.
- PROFINET. *PROFINET – Descrição do sistema: Tecnologia e aplicação*. [s.n.], 2018. Disponível em: [http://www.profibus.org.br/files/descricao\\_tecnica\\_profinet.pdf](http://www.profibus.org.br/files/descricao_tecnica_profinet.pdf). Acesso em: 20 Janeiro 2023.
- SCHWAB, K. *The fourth industrial revolution*. [S.l.]: Currency, 2017.
- SENGE, P. M. *A quinta disciplina: a arte e prática da organização que aprende*. [S.l.]: 28ª ed. São Paulo, 2012.
- SILVA, H. S. Soluções lógicas em linguagem ladder utilizando o tlp logixpro simulation como ferramenta didática para roteiros de aulas práticas de clp. Universidade Federal Rural do Semi-Árido, 2020.
- SILVA, M. E. D. *Controladores Lógico Programáveis-Ladder*. [S.l.]: CONTROLADORES, 2007.
- SILVEIRA, C. B. *IHM: Saiba quais os Tipos e como Selecionar*. 2016. Disponível em: <https://www.citisystems.com.br/ihm/>.
- SILVEIRA, C. B. *Indústria 4.0: O que é, e como ela vai impactar o mundo*. 2016. Disponível em: <https://www.citisystems.com.br/industria-4-0/>.



VIEIRA, R. A. U. Aplicação de processamento de imagens no controle de planta modelo: interoperabilidade com controles industriais utilizando comunicação ethernet. Vitória, 2022.

XAVIER, G. T. d. S. Desenvolvimento de um sistema inteligente ocr utilizando visão computacional para leitura de etiquetas de roteador. Brasil, 2022.