

Inspeção e classificação de defeitos de componentes nas PCIs aplicando técnicas de *Deep Learning*

Inspection and classification of components assembly on PCBs applying Deep Learning techniques

Fernando A R Finardi¹

 orcid.org/0000-0002-5118-8678

Carlos E Fernandes¹

 orcid.org/0000-0001-6894-011X

Fernando L Amadeu¹

 orcid.org/0000-0001-9067-8162

Maria G Valus¹

 orcid.org/0000-0002-3054-6376

Bruno J T Fernandes²

 orcid.org/0000-0002-6001-3925

¹E Fundação para Inovações Tecnológicas – FITec, Campinas, Brasil.

E-mail: ffinardi@fitec.org.br;
cefernandes@fitec.org.br;
famadeu@fitec.org.br;
goretevalus@fitec.org.br

²Escola Politécnica de Pernambuco, Universidade de Pernambuco, Recife, Brasil.
E-mail: bruno.fernandes@upe.br

Artigo recebido em:
Artigo aceito em:

DOI: 10.25286/rep.v7i2.2220

Esta obra apresenta Licença Creative Commons Atribuição-Não Comercial 4.0 Internacional.

Como citar este artigo pela NBR 6023/2018: Fernando A R Finardi; Carlos E Fernandes; Fernando L Amadeu; Maria G Valus; Bruno J T Fernandes. Inspeção e classificação de defeitos de componentes nas PCIs aplicando técnicas de Deep Learning. Revista de Engenharia e Pesquisa Aplicada, v.7, n. 2, p. 75-85, 2022.

RESUMO

A Placa de Circuito Impresso (PCI) é praticamente usada em todos os produtos eletrônicos que usamos no cotidiano, quer seja para fins comerciais ou em outras aplicações tecnológicas. Devido a relevância da aplicação, as PCIs, após o processo de montagem de componentes, necessitam de um sistema de inspeção e localização de defeitos na montagem para garantir a qualidade das suas aplicações. A montagem de um componente da placa de forma errônea pode causar falhas significativas na etapa final do produto. Para classificar os defeitos dos componentes gerados artificialmente das PCIs de referência foi aplicado o algoritmo baseado nas redes neurais convolucionais (CNNs). E os resultados indicaram que o algoritmo aplicado pode ser utilizado na inspeção e classificação de defeitos em PCIs para um sistema de baixo custo.

PALAVRAS-CHAVE: *Deep Learning*, Inspeção de componentes de PCI, CNN, ResNet50.

ABSTRACT

The Printed Circuit Board (PCB) is used in almost every electronic product we use everyday, whether for commercial purposes or in other technological applications. Due to the relevance of the application, the PCBs, after the component assembly process, need an inspection system and assembly defects location to guarantee the quality of their applications. Mistakenly mounting a board component can cause significant failures in the final product step. To classify the defects of the artificially generated components of the reference PCBs, the algorithm based on convolutional neural networks (CNNs) was applied. And the results indicated that the applied algorithm can be used in the inspection and classification of defects in PCIs for a low-cost system.

KEY-WORDS: *Deep Learning*, Inspection of PCB components, CNN, ResNet50.

1 INTRODUÇÃO

Tecnologia e eletrônica são duas temáticas importantes, ambas se complementam, especialmente no mundo pós-moderno, onde os sistemas estão mais complexos, automatizados e dependentes dos recursos de dispositivos eletrônicos preferencialmente diversificados e com densidades de componentes com nível de milímetros ou mesmo microns. A Placa de Circuito Impresso (PCI) ou *Printed Circuit Board* (PCB) representa essa estrutura na sua composição e é um elemento fundamental em todo e qualquer eletrônico. As PCIs oferecem versatilidade na inserção de componentes utilizando vários designs, de acordo com a complexidade das tarefas a serem realizadas pelo produto. A PCI é uma plataforma que conecta componentes eletrônicos usando trilhas condutoras e não condutoras com um circuito bem projetado e definido. As várias camadas condutoras de PCI são geralmente conectadas por meio de cobre e nas camadas não condutoras são utilizados o compósito de epóxi e a fibra de vidro [1-3].

A montagem de componentes é uma etapa importantíssima no processo de etapa final da produção da placa nas empresas de manufatura. A correta inserção dos componentes permite que essa etapa evite trabalhos adicionais, por exemplo, a troca e a reorganização dos componentes. Além disso, os componentes montados de forma errônea podem causar limitação no desempenho do produto, inclusive podendo ocorrer danos à placa por curto-circuito.

A indústria de PCIs é diversificada, pois cada planta de produção de PCIs produz diferentes tipos de PCIs. Devido a essa diversidade é difícil de desenvolver modelos flexíveis que possam ser aplicados simultaneamente em várias fábricas. Além disso, as PCIs apresentam vários elementos que são desafios no reconhecimento de seus componentes e sua similaridade, e que ainda são difíceis de serem resolvidos pelos algoritmos existentes. Um dos aspectos que contribui é o desenvolvimento tecnológico constante que permite um nível mais alto de complexidade com projetos mais compactos dos componentes das PCIs [3-5].

Uma alternativa para inspeção e detecção de defeitos em PCIs e ao mesmo tempo avaliar a qualidade é a aplicação de métodos de *Machine Learning* e *Deep Learning* com base de processamento de imagem, para localização de defeitos automaticamente nas PCIs que podem

refletir a diminuição de custo, trabalho e esforço humano [3-5].

Neste projeto, tem sido um desafio obter um conjunto de dados dos componentes de PCIs com os seus defeitos que estejam publicamente disponíveis, mas na literatura, já é possível obter banco de dados de imagens, por exemplo, o DeepPCB [4] está disponibilizado contendo 1500 imagens, incluindo diversos defeitos de PCIs. Tanto que, Silva *et al.* [5] utilizaram o conjunto de dados DeepPCB e propuseram método baseado em CNNs para a detecção de defeitos. A CNN previamente treinada no conjunto de dados *ImageNet* foi usada para extrair as *features*. O modelo atingiu uma precisão de 89%. As imagens DeepPCB usadas em seu estudo eram binárias com texturas simples.

Pensando nisso, o objetivo deste trabalho é o desenvolvimento de um sistema de inspeção e classificação de defeitos em PCIs de baixo custo, executado por qualquer pessoa utilizando câmeras de celular em ambientes não controlados. Este mesmo sistema pode ser agregado ao grupo de desenvolvimento de *hardware* da FITec com o intuito de aumentar a confiabilidade e qualidade de inspeção das PCIs recebidas das montadoras na fase de depuração dos projetos em desenvolvimento. O trabalho consiste em aplicar técnicas de classificação de imagens nos campos de aprendizagem de máquina por meio de um algoritmo *Deep Learning* para inspecionar os componentes e a sua localização.

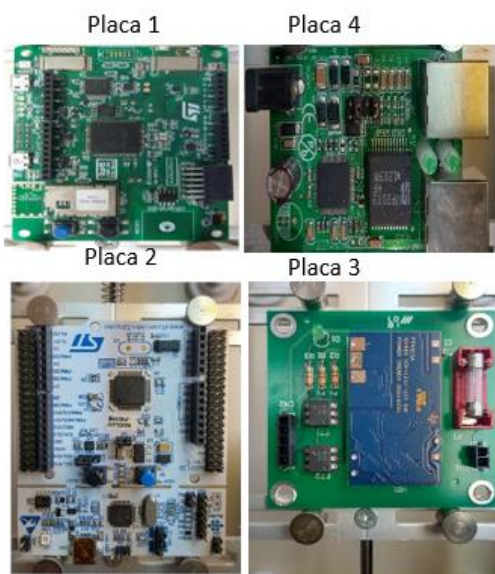
Este artigo está estruturado da seguinte forma: na Seção 2 apresenta e discute brevemente alguns trabalhos relacionados. Enquanto na Seção 3 fornecemos uma análise do *dataset* construído pelos próprios autores e os procedimentos seguidos. Na Seção 4 é realizado o treinamento e teste do *dataset* fornecendo os resultados obtidos. Finalmente, discutimos as conclusões do presente caso do estudo fornecendo orientações futuras devido aos desafios encontrados na construção do *dataset*, no treinamento e teste dos modelos aplicados desde o início do desenvolvimento do projeto proposto e as mudanças e adaptações que foram sendo realizadas.

2 REFERENCIAL TEÓRICO

Nas inspeções de PCIs conduzidas por meios visuais de humanos, podem ocorrer erros de inspeção devido alguns fatores como a inconsistência, a fadiga, a dificuldade de unificar os critérios de teste e entre outros. Doro [2]

afirma que os defeitos podem ser divididos em três conjuntos: defeitos nos componentes ou na placa, defeitos no processo de terminações e defeitos na inserção dos componentes. Os defeitos nos componentes ou na placa ocorrem devido à má qualidade dos componentes ou das placas durante o processo de montagem, assim como, pelo armazenamento, transporte ou manuseio incorreto. Os defeitos nas terminações são aqueles gerados durante o processo de soldagem que podem ocorrer em função aos diversos fatores entre os terminais do componente e a placa. Por exemplo: espalhamento ou excesso de solda causando um curto-circuito; insuficiência ou ausência de solda; buracos nas juntas de solda. Durante as etapas de processo de manufatura de inserção de componentes nas PCIs, os defeitos podem ser causados tanto em operações manuais tanto quanto por automáticas. A Figura 1 apresenta quatro modelos diferentes de PCIs selecionados e nomeados do tipo *Golden* com os componentes inseridos, conforme a especificação, e são placas de referência para o estudo de caso.

Figura 1 - Placa 1, 2, 3 e 4 utilizadas como referência na inserção de defeitos artificiais.

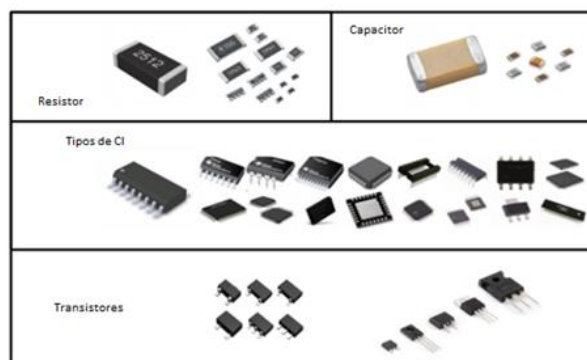


Fonte: Os autores.

Verificando a Figura 1, com as quatro Placas *Golden* de referência de modelos diferentes, observa-se que as PCIs são compostas por vários tipos diferentes de componentes. Os mesmos tipos de componentes parecem semelhantes e são classificados como: baixa variação de intraclasses

que são representados pelos circuitos integrados (CIs) de três pinos e os capacitores. E há exemplos do mesmo tipo de componentes que se parecem diferentes, estes são classificados por alta variação de intraclasses que é representado por uma variedade de diferentes CIs. Inclusive, por exemplo, os resistores e os capacitores, também, podem parecer semelhantes sob certas condições de iluminação. A semelhança entre os componentes pode resultar em detecções falsas positivas. A Figura 2 apresenta exemplos de quatro tipos comuns de componentes de PCIs [6].

Figura 2 - Exemplos de quatro tipos comuns de componentes de PCIs: os resistores, os capacitores, os tipos de CIs e os transistores.



Fonte: [6].

A inspeção dos defeitos e classificação de componentes de PCIs podem ser realizadas aplicando diversas técnicas de aprendizado de máquina do domínio da Inteligência Artificial baseadas em imagens na comparação dos componentes a partir das imagens armazenadas no banco de dados. As diversas técnicas de comparação variam desde algoritmos básicos até os mais complexos, por exemplo, técnicas usando as estruturas complexas dos métodos RCNNs (*Region based Convolutional Neural Networks*), YOLO (*You Only Look Once*) e SSD (*Single Shot Detector*) [5-7].

A técnica de *Deep Learning* (DL), nos últimos anos, mostrou-se produzir resultados notáveis quando comparada com algoritmo de visão computacional clássico em uma grande variedade de aplicações. O DL permite modelos computacionais de várias camadas para representar e aprender recursos e dados com muitas camadas de abstração. Os métodos de visão computacional baseados em aprendizagem profunda são fundamentados nas CNNs. As CNNs são uma classe de rede neural artificial aplicada principalmente para a análise e processamento de

imagens digitais e são um dos melhores classificadores de imagens. Uma das vantagens, a quantidade de informações processada é menor devido a capacidade que possui de aprender, considerando somente as regiões de interesse mais relevantes para o conteúdo analisado [8 - 9].

Na classificação, a CNN possui três principais camadas: a camada convolucional que cria um mapa de *feature* para prever as possibilidades de classe para cada *feature* aplicando um filtro que varre toda a imagem; seguida de camada *pooling* que reduz a quantidade de informações da camada convolucional e, finalmente, a matriz passa por uma camada totalmente conectada para classificar as imagens [10]. Para detecção de objetos e classificação de uma imagem, ou região de uma imagem, Girshick *et al.* [11] e Girshick [12] afirmam que pode ser realizada por classificadores binários. O método *region-CNN* (R-CNN), por exemplo, para a detecção de objetos explora o método seletivo de busca composta por três etapas principais: define o conjunto de regiões propostas; extrai características de cada região proposta via CNN e realiza a classificação binária via classificadores específicos de classe. Adibhatla *et al.* [13] apresentaram um método para reconhecer PCIs defeituosas. Seu método é baseado na CNN e ele classifica cada PCI como PCI com defeito ou intacto. Zhang *et al.* [14]. desenvolveram uma abordagem para inspeção de defeitos de PCI aprendendo características discriminativas profundas. A localização de defeitos foi realizada usando uma abordagem de deslizamento de janela. Ding *et al.* [15] propuseram uma abordagem TDD-Net (*Tiny Defect Detection Network*) baseada em *Faster-RCNN* para detectar pequenos defeitos nas PCIs. Seu método resolve as deficiências das redes convolucionais profundas na detecção de pequenos defeitos de PCIs com alta precisão pelos resultados obtidos.

Neste estudo de caso, é aplicado a *ResNet* que é uma rede convolucional como *backbone* nas tarefas de visão computacional para resolver uma série de problemas. Usa o aprendizado residual para conectar o mapa de recursos profundo e o raso anterior. Os recursos de baixo e alto nível são utilizados para combinar suas vantagens, e nisso podem se adaptar melhor na detecção de pequenos alvos. Muitas variantes funcionam no mesmo conceito, mas possuem números diferentes de camadas [16]. Zhang, Jiang e Li. [17] propuseram um modelo de *ResNet* modificado com uma camada

de ajuste ao final na detecção de defeitos cosméticos de PCIs.

3 MATERIAIS E MÉTODOS

Nesta seção estão descritas as fases de preparação dos dados, ou seja, a composição do *dataset* de treino e teste, o pré-processamento dos dados, o treinamento e a avaliação do modelo e o desenvolvimento e testes aplicados na inspeção e classificação das PCIs.

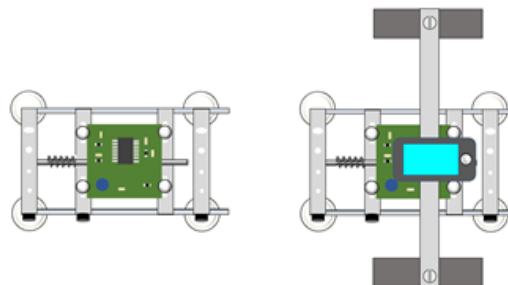
Na primeira fase, realizou-se a preparação dos dados. O *dataset* para verificar a detecção e classificação de defeitos de PCIs são de autoria própria e não serão disponibilizados publicamente devido as informações serem confidenciais de projetos, priorizando as placas que são comumente solicitadas nos projetos desenvolvidos pela FITec (Fundação para Inovações Tecnológicas).

Foram selecionados quatro modelos diferentes de PCIs para servirem de modelo de referência, na qual denominamos de modelos *Golden*. Em seguida, foram retirados manualmente alguns componentes de cada placa para simular a geração de defeito de ausência ou possível falha na montagem da PCI.

Para a construção do *dataset*, foram capturadas centenas de imagens, divididas em imagens para treinamento e imagens para teste.

Todas as imagens da amostra de referência (*Golden*) ou com defeitos foram capturadas de cima para baixo, usando uma configuração de montagem vertical feito por um suporte de fixação de placa para solda e retrabalho profissional modelo *Yaxum Yx-10*, apresentado na Figura 3.

Figura 3 - Suporte de placa.



Fonte: Os autores.

As imagens de treinamento e teste foram capturadas manualmente sob condições de iluminação moderada, para reduzir o reflexo emitido dos diferentes materiais e componentes das PCIs, utilizando uma câmera de celular modelo

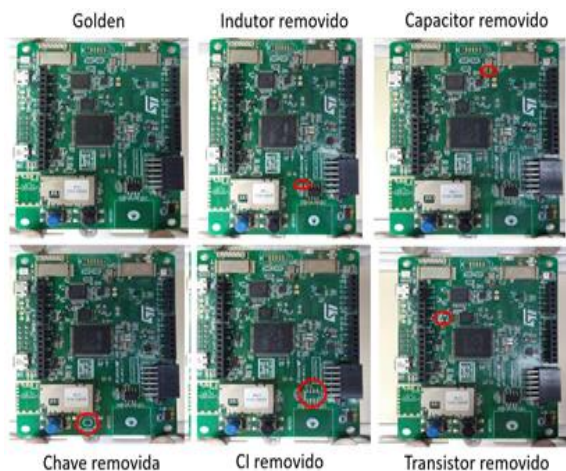
moto G8 Plus com câmera de até 48 MP e zoom ajustado em 1,5x. As imagens foram coletadas em uma bancada, próxima à uma janela para melhor condição de iluminação, sempre no início da manhã.

Para o conjunto de imagens de validação, as imagens foram capturadas sem o suporte de fixação, não levando em consideração as variações de iluminação ambiente, fundo, reflexos ou ajuste de zoom, ou seja, ficou a cargo do operador da câmera definir o local e o melhor ângulo para a coleta da foto. A câmera de celular utilizada foi a do celular *Asus Zenfone 4 Selfie* de 16 MP. Essas imagens foram coletadas para verificar o funcionamento do algoritmo com condições piores de aquisição das imagens.

3.1 COMPOSIÇÃO DE IMAGENS PARA TREINAMENTO, VALIDAÇÃO E TESTE

A Placa 1, apresentada na Figura 4, compõe o primeiro conjunto de imagens de treinamento contendo os seguintes defeitos definidos neste processo: Indutor removido, Capacitor removido, Chave removida, CI (Circuito Integrado) removido e Transistor removido. É importante ressaltar que não foi considerado o algoritmo para defeitos de solda (curto, insuficiência ou solda fria), tendo em vista que o grau de dificuldade de detecção é bem maior.

Figura 4 - Placa 1 de referência *Golden* e os defeitos de componentes, respectivamente: (a) *Golden*; (b) Indutor removido; (c) Capacitor removido; (d) Chave removida; (e) CI removido e (f) Transistor removido.

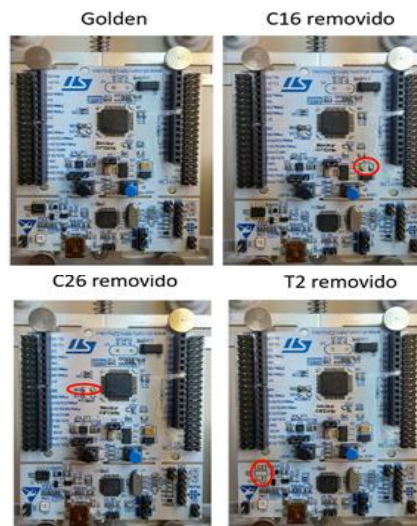


Fonte: Os autores.

A Figura 5, apresenta a Placa 2, compõe o segundo conjunto de imagens de treinamento

contendo os seguintes defeitos definidos neste processo: capacitor C16 removido, capacitor C26 removido, Transistor T2 removido.

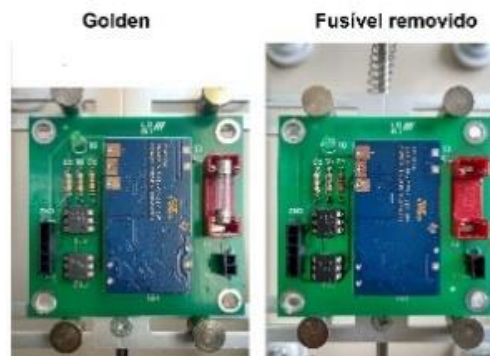
Figura 5 - Placa 2 de referência *Golden* e os defeitos de componentes, respectivamente: (a) *Golden*; (b) C16 removido; (c) C26 removido; (d) T2 removido.

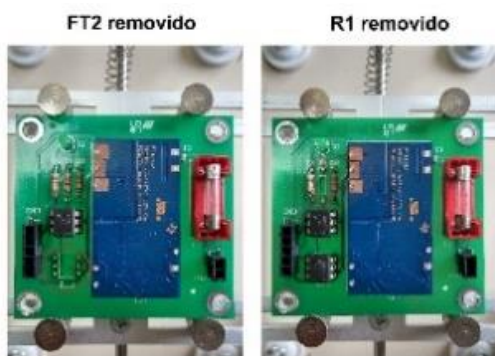


Fonte: Os autores.

A Figura 6, apresenta a Placa 3, compõe o terceiro conjunto de imagens de treinamento contendo os seguintes defeitos definidos neste processo: fusível F1 removido e CI FT2 removido e Resistor R1 removido.

Figura 6 - Placa 3 de referência *Golden* e os defeitos de componentes, respectivamente: (a) *Golden*; (b) F1 removido; (c) FT2 removido e (d) R1 removido.

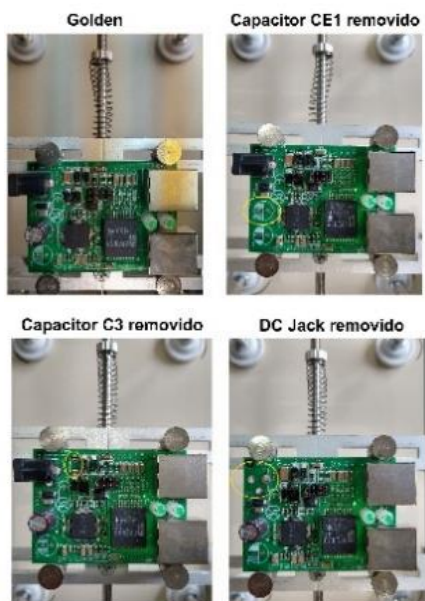




Fonte: Os autores.

A Figura 7, apresenta a Placa 4, compõe o quarto e último conjunto de imagens de treinamento contendo os seguintes defeitos definidos neste processo: Capacitor CE1 removido, Capacitor C3 e Conector DC removido.

Figura 7 - Placa 4 de referência *Golden* e os defeitos de componentes, respectivamente: (a) *Golden*; (b) CE1 removido; (c) C3 removido e (d) DC Jack removido.

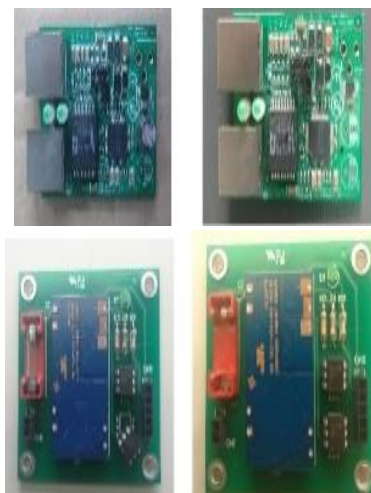


Fonte: Os autores.

Da mesma forma, foi construído um novo *dataset* contendo centenas de imagens para a validação do modelo, seguindo a mesma metodologia apresentada nas Figuras 4, 5, 6 e 7. Algumas das imagens foram pós-processadas a fim de gerar uma variação maior no conjunto de testes, variando brilho, contraste, foco, matiz, realce, sombra e esmaecimento.

Para que o teste do modelo pudesse ser o mais próximo da realidade, solicitamos a um operador fora deste grupo de trabalho que fizesse uma aquisição de 5 fotografias aleatórias das placas 3 e 4 para compor o novo *dataset*. O ângulo, a iluminação do ambiente, o fundo e o *zoom* ficaram a cargo do operador da câmera. A Figura 8 mostra como estas novas imagens ficaram.

Figura 8 - Placas 3 e 4 com imagens aleatórias.



Fonte: Os autores.

Na segunda fase, pré-processamento dos dados, as imagens de placas foram organizadas por tipo de placa (para treinamento do modelo de classificação de placas), e foram também organizadas em diferentes classificações de defeitos, separadas por tipo de placa. A partir do conjunto de imagens disponível, 80% das imagens foram separadas para o treinamento e 20% das imagens, selecionadas aleatoriamente, foram separadas para validação. Foram aplicadas técnicas de aumento de dados no conjunto de treino, gerando variações de brilho e contraste, inversões horizontais e verticais e variação de cores.

A terceira fase, foi dividida em treinamento e avaliação do modelo. Para o treinamento dos modelos de aprendizagem de máquina foi utilizado o *Transfer Learning* com a arquitetura de rede neural *ResNet50*. A partir do uso desta rede pré treinada com o *Dataset ImageNet* foram aplicados os hiperparâmetros. Porém, os que apresentaram melhor ajuste são os seguintes:

- *Learning Rate*: 0,0001;
- decaimento para 10% do *Learning Rate* anterior a cada 7 épocas;
- otimizador Adam;

- *CrossEntropyLoss* como função de perda e
- treinamento entre 10 e 20 épocas a partir de rede pré treinada.

Ao fim da etapa de treinamento, foram gerados cinco diferentes modelos de *Machine Learning*. O primeiro modelo treinado é responsável por realizar uma classificação entre os quatro diferentes tipos de placa. Os demais quatro modelos foram treinados com o objetivo de identificar os defeitos das placas, sendo gerado um modelo de classificação para cada tipo de placa.

Para a avaliação do modelo foram aplicadas algumas métricas, tais como: Precisão, *Recall*, *F1-score* e Acurácia. A aplicação das métricas, detalhadamente, pode ser verificada, por exemplo, em [18-20].

Na quarta fase, desenvolvimento e testes, foi gerado um *notebook* contendo três fases de processamento. Na primeira fase é feito o alinhamento da imagem via *software*, verificando o melhor ângulo de enquadramento da imagem e realizando rotação dela. Em seguida, é utilizado o modelo de classificação do tipo de placa, previamente treinado, para verificar qual a placa que está sendo analisada. Por fim, com o resultado do tipo de placa, é utilizado o modelo de identificação de defeitos específico para o tipo de placa em questão, previamente treinado, para verificar se a placa analisada possui algum defeito ou se está de acordo com o padrão *Golden* (de referência).

Os testes finais do processo foram realizados com as imagens coletadas pelo operador (imagens mais próximas a realidade), simulando um ambiente produtivo para avaliação final do desempenho do processo.

4 RESULTADOS E DISCUSSÕES

Durante os treinamentos, todos os modelos atingiram 100% de acurácia no treinamento e na validação ao fim do treino. A obtenção de 100% de acurácia, onde o algoritmo de aprendizado de máquina se torna altamente adaptado aos dados de entrada, havendo baixa generalização. Isso se justifica devido ao processo de captura e coleta de dados definido na seção 3, fazendo com que as imagens coletadas possuam baixa variância. Os dados de validação, por sua vez, acabam sendo similares as imagens vistas durante o treinamento, fazendo com que o conjunto de validação também acabe tendo um valor alto de

acurácia. Utilizou-se técnica de aumento de dados com objetivo de diversificar o *dataset* e reduzir a ocorrência de *overfitting*.

A partir de um conjunto de testes distinto das imagens usadas para treino e validação, e com maior variação entre imagens, foram geradas as matrizes de confusão e métricas de classificação (Acurácia, Precisão, *Recall* e *F1-score*) para os cinco modelos de aprendizagem de máquina. O conjunto de testes segue o procedimento de captura de imagens definido na seção 3 deste trabalho.

O detalhamento dos resultados para cada modelo é apresentado nas seções a seguir.

4.1 IDENTIFICAÇÃO DE TIPOS DE PLACAS

Tabela 1 – Matriz de confusão e métricas da identificação de placas.

	Predição				
	Classe 1	Classe 2	Classe 3	Classe 4	
Real classe	Classe 1	82	0	0	0
	Classe 2	0	56	0	0
	Classe 3	0	0	72	0
	Classe 4	0	0	0	73

	Precisão	<i>Recall</i>	<i>F1-Score</i>	Quantidade imagens
Classe 1	100%	100%	100%	82
Classe 2	100%	100%	100%	56
Classe 3	100%	100%	100%	72
Classe 4	100%	100%	100%	73
	Acurácia			
	100%			

Fonte: Os autores.

O modelo de aprendizagem de máquina para identificação dos diferentes tipos de placas obteve bons resultados. Obtendo 100% em todas as métricas para o conjunto de teste mostradas na Tabela 1. Esse resultado é justificado pelo alto grau de diferença entre as classes.

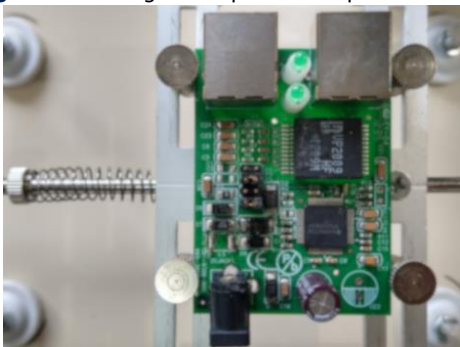
4.4.1 Teste em cenário real para identificar tipos de placa

Imagens mais próximas a realidade, capturadas por um operador, foram usadas no processo produtivo (detalhado na seção 3) para melhor

avaliar o desempenho do modelo e verificar pontos de melhoria.

Para os testes realizados, as imagens, com exceção de uma, tiveram o tipo de placa corretamente classificado, comprovando o bom desempenho do modelo. A imagem em questão, apresentada na Figura 9, foi incorretamente classificada devido à falta de foco na imagem.

Figura 9 – Imagem da placa com problema de foco.



Fonte: Os autores.

4.2 DEFEITOS DA PLACA 1

Tabela 2 – Matriz de confusão e métricas para Placa 1.

Real classe	Predição			
	Classe 1	Classe 2	Classe 3	Classe 4
Classe 1	1	15	0	0
Classe 2	0	0	0	0
Classe 3	0	0	1	15
Classe 4	0	0	0	0

	Precisão	Recall	F1-Score	Quantidade imagens
Classe 1	100%	6%	12%	16
Classe 2	0%	0%	0%	0
Classe 3	100%	6%	12%	16
Classe 4	0%	0%	0%	0
Acurácia				
	5%			

Fonte: Os autores.

O modelo de identificação de defeitos da placa 1 não gerou bons resultados, apresentados na Tabela 2. Pois, além de um conjunto de testes limitado, possui classes com alto grau de similaridade e defeitos que compõem uma parte muito pequena da imagem.

4.3 DEFEITOS DA PLACA 2

Tabela 3 – Matriz de confusão e métricas para Placa 2.

Real classe	Predição				
	Classe 1	Classe 2	Classe 3	Classe 4	Classe 5
Classe 1	1	0	0	0	15
Classe 2	0	1	20	0	0
Classe 3	0	0	0	0	0
Classe 4	0	0	1	18	0
Classe 5	0	0	0	0	0

	Precisão	Recall	F1-Score	Quantidade imagens
Classe 1	100%	6%	12%	16
Classe 2	100%	5%	9%	21
Classe 3	0%	0%	0%	0
Classe 4	100%	5%	10%	19
Classe 5	0%	0%	0%	0
Acurácia				
	5%			

Fonte: Os autores.

O modelo de identificação de defeitos da placa 2 não gerou bons resultados, mostrados na Tabela 3. A área onde se apresentam os defeitos compõe uma parte muito pequena da imagem fazendo com que o algoritmo de aprendizado de máquina, que está analisando a imagem completa, tenha seu desempenho comprometido.

4.4 DEFEITOS DA PLACA 3

Tabela 4 – Matriz de confusão e métricas para Placa 3.

Real classe	Predição			
	Classe 1	Classe 2	Classe 3	Classe 4
Classe 1	21	0	0	0
Classe 2	0	38	0	0
Classe 3	0	0	20	15
Classe 4	0	0	0	21

	Precisão	Recall	F1-Score	Quantidade imagens
Classe 1	100%	100%	100%	21
Classe 2	100%	100%	100%	38
Classe 3	100%	100%	100%	20
Classe 4	100%	100%	100%	21
Acurácia				
	100%			

Fonte: Os autores.

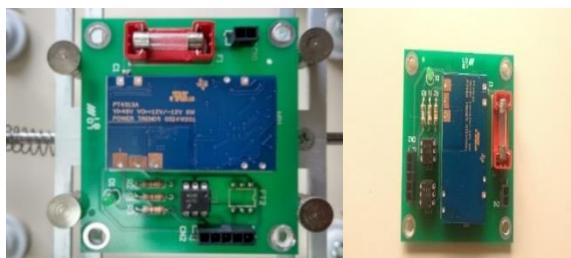
Para a placa 3 todas as classificações de imagens do conjunto de teste, mostradas na Tabela 4, foram corretamente inferidas, sendo possível obter 100% entre todas as métricas. Diferentemente das placas 1 e 2, a placa 3 possui defeitos mais evidentes, obtendo melhor desempenho nas classificações do conjunto de teste.

4.4.1 Teste em cenário real referente a placa 3

Imagens mais próximas a realidade, capturadas por um operador, foram usadas no processo produtivo (detalhado na seção 3) para melhor avaliar o desempenho do modelo e verificar pontos de melhoria.

Dentre 20 imagens testadas, 15 tiveram os defeitos corretamente classificados. Duas das imagens tiveram uma classificação incorreta devido à falta de foco e estão apresentadas na Figura 10. As 3 imagens restantes foram incorretamente classificadas devido a angulação de captura que diferiu muito em relação ao *dataset* de treino.

Figura 10 – Imagem de detalhamento dos problemas encontrados.



Fonte: Os autores.

4.5 DEFEITOS DA PLACA 4

Tabela 5 – Matriz de confusão e métricas para Placa 4.

		Predição					
		Classe 1	Classe 2	Classe 3	Classe 4	Classe 5	Classe 6
Real classe	Classe 1	21	0	0	0	0	0
	Classe 2	0	21	0	0	0	0

	Classe 3	0	0	21	0	0	0
	Classe 4	0	0	0	16	0	0
	Classe 5	0	0	0	0	20	0
	Classe 6	0	0	0	0	0	16

	Precisão	Recall	F1-Score	Quantidade imagens
Classe 1	100%	100%	100%	21
Classe 2	100%	100%	100%	21
Classe 3	100%	100%	100%	21
Classe 4	100%	100%	100%	16
Classe 5	100%	100%	100%	20
Classe 6	100%	100%	100%	16
	Acurácia			
	100%			

Fonte: Os autores.

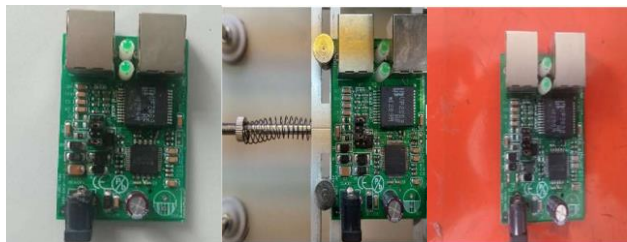
Para a placa 4 todas as classificações de imagens do conjunto de teste foram corretamente inferidas, sendo possível obter 100% entre todas as métricas que pode ser verificado na Tabela 5.

4.5.1 Testes em cenário real referente a placa 4

Imagens mais próximas a realidade, capturadas por um operador, foram usadas no processo produtivo (detalhado na seção 3) para melhor avaliar o desempenho do modelo e verificar os pontos de melhoria.

Dentre 25 imagens testadas, 12 imagens erraram na classificação do defeito. O principal ofensor observado para este resultado foi a angulação de captura das imagens, que difere do padrão de imagens utilizado no treinamento. Assim como, o fundo das imagens, devido a captura em diferentes localizações que pode ser verificado na Figura 11. As deteriorações no foco e na exposição da imagem também impactaram negativamente a classificação do defeito.

Figura 11 – Imagem com diferente angulação à esquerda, placa *Golden* ao centro e imagem com fundo diferente à direita.



Fonte: Os autores.

5 CONCLUSÕES

Diante do objetivo proposto no trabalho, foi desenvolvido um sistema de inspeção e classificação de defeitos em placas PCI, considerando um ambiente não controlado, onde uma imagem de câmera de celular pudesse ser utilizada como ferramenta de captura. Ao todo foram treinados cinco modelos de *Machine Learning* para suportar o processo de inspeção.

O método de classificação melhorado dos defeitos de PCIs proposto neste trabalho usa as redes neurais convolucionais para desenvolver uma solução capaz de identificar defeitos dinamicamente entre diferentes tipos de placas.

Apesar dos altos índices de acerto no primeiro teste dos modelos, o teste considerando as imagens em ambiente real não apresentou o mesmo resultado. Observou-se que o modelo de classificação de defeitos apresentou *overfitting*, devido à baixa variância entre as imagens coletadas e entre diferentes classificações de defeito, pois o *setup* de captura considerou mesmo modelo de câmera, *zoom* e iluminação controlada entre as amostras.

Espera-se que este trabalho contribua para o desenvolvimento de uma ferramenta de baixo custo para a divisão de *hardware* da FITec, aumentando a confiabilidade e qualidade de inspeção das PCIs na fase de depuração dos projetos em desenvolvimento, a qual pode ser executada por qualquer pessoa, a partir de imagens capturadas de câmeras de celular em ambientes não controlados. As imagens submetidas à ferramenta serão processadas e o resultado de retorno da análise será um indicativo de qual é o tipo de placa analisada e qual o defeito encontrado na mesma, caso exista.

5.1 TRABALHOS FUTUROS

No futuro, é necessário aumentar o conjunto de treinamento. Com o objetivo de aumentar a variância entre imagens e obter um modelo mais generalizado para o problema. É recomendado aprimorar o conjunto de treinamento fazendo a coleta de imagens similares as imagens capturadas pelo operador, as quais possuam maior variância de angulação, foco, *zoom* e luminosidade.

Como as placas 1 e 2 não apresentaram resultados positivos para o conjunto de teste adquirido, faz-se necessário realizar novos treinamentos com um conjunto maior de imagens, considerando também maiores variações entre elas. Destaca-se também a necessidade de análise aprofundada dos problemas de classificação para que possam ser feitas novas coletas de dados mais assertivas e melhorar os treinamentos de forma geral.

Como uma nova abordagem ao problema, é sugerido realizar o treinamento de modelos de *Machine Learning* a partir de resultados de análise de similaridade de regiões da imagem. A partir dos resultados de similaridade destas regiões entre a placa com defeito e a placa de referência, seria realizado um treinamento de rede neural para classificação do defeito. Para essa abordagem, é necessário que as placas estejam na mesma orientação, sendo fundamental um ambiente controlado ou semicontrolado para a coleta das imagens.

Este trabalho foi desenvolvido pensando na utilização em ambiente de desenvolvimento, com poucos protótipos produzidos, mas que necessitam de uma análise rápida da qualidade da montagem. Para testes em linha de produção, é possível adaptar o algoritmo, mas há a necessidade de outras implementações que tornem o processo ainda mais automatizado, uma vez que esse tipo de ambiente requer agilidade.

6 AGRADECIMENTOS

À FITec pelo incentivo. Ao Orientador Dr. Prof. Bruno Fernandes e ao Dr. Prof Carmelo Bastos pela disponibilidade desprendida e a todos os Professores que contribuíram durante a jornada do curso.

REFERÊNCIAS

- [1] FUSE TTN. *Printed circuit board and surface mount design*. Versão 7 Jun. 1999.
- [2] DORO, M. M. **Sistemática para implantação da garantia da qualidade em empresas montadoras de placas de circuito impresso**. Dissertação de Mestrado. Universidade Federal de Santa Catarina, 2004.
- [3] CASTELLANI, M.; OTRI, S.; PHAM, D. T. *Printed circuit board assembly time minimisation using a novel Bees Algorithm*. **Computers & Industrial Eng.**, 2019.
- [4] TANG, S.; HE, F.; HUANG, X.; YANG, J. *Online PCB defect detector on A new PCB defect dataset*. In: **arXiv preprint arXiv:1902.06197**, 2019.
- [5] SILVA, H. de S.; AZEVEDO, G. de A.; FERNANDES, B. J. T.; BEZERRA, B. L. D.; LIMA, E. B.; OLIVEIRA, S. C. *Automatic optical inspection for defective PCI detection using transfer learning*. In **Proceedings of the 2019 IEEE Latin American Conference on Comp. Intelligence, (LA-CCI)**, Guayaquil, Ecuador, 11-15 Nov. 2019.
- [6] SATHIASEELAN, M. A. M.; PARADIS, O. P.; TAHERI, S.; ASADIZANJANI, N. *Why is Deep Learning challenging for printed circuit board (PCB) component recognition and how can we address it?* **Cryptography**, 2021.
- [7] PEALLAT, J.-M.; WARNCKE, R.; CLAYBROOK, R.; BRUN, M. *What is really inside your AOI?* 2018. Disponível em: <<http://amtest.bg/press/AOI/What-is-really-inside-your-AOI.pdf>>. Acesso em: 18 de Jul. de 2021.
- [8] KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, E. G. *Imagenet classification with deep convolutional neural networks*. In: **Advances in neural information processing systems**, 2012.
- [9] ROSEBROCK, A. *Deep Learning for Computer Vision with Python*. 1. ed. New York: Pyimagesearch, 2017.
- [10] GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. Cambridge: MIT press, 2016.
- [11] GIRSHICK, R. *Fast R-CNN*. In: **IEEE Computer Vision (ICCV)**, 2015.
- [12] GIRSHICK, R.; DONAHUE, J.; DARRELL, T.; MALIK, J. *Rich feature hierarchies for accurate object detection and semantic segmentation*. In: **Proceedings of the IEEE conference on computer vision and pattern recognition**, 2014.
- [13] ADIBHATLA, V. A.; SHIEH, J.-S.; ABBOD, M. F.; CHIH, H.-C.; HSU, C.-C.; CHENG, J. *Detecting Defects in PCB using Deep Learning via Convolution Neural Networks*, in **2018 13th International Microsystems, Packaging, Assembly and Circuits Technology Conference (IMPACT)**, Taipei, Taiwan, Taiwan, IEEE, 2018.
- [14] ZHANG, C.; SHI, W.; LI, X.; ZHANG, H.; LIU, H. *Improved bare PCB defect detection approach based on deep feature learning*, *The Journal of Engineering*. 1415-1420, 2018.
- [15] DING, R.; DAI, L.; LI, G.; LIU, H. *TDD-Net: A tiny defect detection network for printed circuit boards*. *CAAI Trans. Intell. Technol.*, 110-116, Jun. 2019.
- [16] HE, K.; ZHANG, X.; REN, S.; SUN, J. *Deep residual learning for image recognition*. **2016 IEEE Conference on Computer Vision and Pattern Recognition**, 2016.
- [17] ZHANG, H.; JIANG, L.; LI, C. *Cs-resnet: cost-sensitive residual convolutional neural network for PCB cosmetic defect detection*. **Expert Systems with Applications**, vol. 185, 2021.
- [18] KIM, J.; KO, J.; CHOI, H.; KIM, H. *Printed circuit board defect detection using deep learning via a skip-connected convolutional autoencoder*. **Sensors**, 2021.
- [19] JUBA, B.; LE, H. S. *Precision-Recall versus Accuracy and the Role of Large Data Sets*. **Proc. of the AAAI Conference on Artificial Intelligence**, 4039-4048, 2019.
- [20] DENG, X.; LIU, Q.; DENG, Y.; MAHADEVAN, S. *An improved method to construct basic probability assignment based on the confusion matrix for classification problem*. **Information Sciences**, 2016.