

UNIVERSIDADE SANTA CECÍLIA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA
MESTRADO EM ENGENHARIA MECÂNICA

JOÃO LUÍS LOPES FREITAS ORSI KUNTZ

REDES NEURAIS ARTIFICIAIS PARA CONSISTENTES APLICADAS NO
MONITORAMENTO ÓPTICO DE FIGURAS RETANGULARES
POR TÉCNICAS DE FOTOGRAMETRIA

SANTOS/SP

2021

JOÃO LUÍS LOPES FREITAS ORSI KUNTZ

**REDES NEURAIS ARTIFICIAIS PARA CONSISTENTES APLICADAS NO
MONITORAMENTO ÓPTICO DE FIGURAS RETANGULARES POR
TÉCNICAS DE FOTOGRAMETRIA**

Dissertação apresentada a Universidade Santa Cecília como parte dos requisitos para obtenção de título de mestre no Programa de Pós-Graduação em Engenharia Mecânica, sob a orientação do Prof. Dr. Maurício Conceição Mário.

SANTOS/SP

2021

Autorizo a reprodução parcial ou total deste trabalho, por qualquer que seja o processo, exclusivamente para fins acadêmicos e científicos.

Kuntz, João Luís Lopes Freitas Orsi.

Redes Neurais Artificiais Paraconsistentes aplicadas no monitoramento óptico de figuras retangulares por técnicas de fotogrametria.

João Luís Lopes Freitas Orsi Kuntz. - 2021

81 páginas

Orientador: Prof. Dr. Maurício Conceição Mário.

Dissertação (Mestrado) -- Universidade Santa Cecília, Programa de Pós-Graduação em Engenharia Mecânica, Santos, SP, 2021.

1. Rede neural artificial paraconsistente. 2. Fotogrametria. 3. Visão computacional. 4. Lógica paraconsistente. 5. Controle de qualidade.

I. Mário, Maurício Conceição. II. Redes neurais artificiais paraconsistentes aplicadas no monitoramento óptico em controle de qualidade por técnicas de fotogrametria.

Elaborada pelo SIBi – Sistema Integrado de Bibliotecas – Unisanta

DEDICATÓRIA

Dedico este trabalho, como um reconhecimento, a todos meus professores que nos últimos meses – devido à pandemia – tiveram o desafio de se reinventarem profissionalmente em um curto espaço de tempo e que, mesmo com as adversidades, conseguiram, com maestria, manter a qualidade do processo de ensino-aprendizagem.

AGRADECIMENTOS

Em primeiro lugar ao meu orientador, professor Dr. Maurício Conceição Mário, que me acompanha desde a graduação e que foi um dos responsáveis por me apresentar à programação, mostrando suas oportunidades e seus desafios, e que novamente está ao meu lado neste momento tão importante.

Aos professores Dr. Aldo Ramos Santos, Dr. Carlos Teófilo Salinas Sedano, Dr. Deovaldo de Moraes Júnior, Dra. Marlene Silva de Moraes, Dra. Talita Vida e Dr. Vitor da Silva Rosa, cujas carreiras são extremamente inspiradoras, capazes de despertar o interesse imediato e uma curiosidade cada vez maior em explorar este universo tão apaixonante como desafiador, que é a Engenharia.

Não poderia deixar de agradecer ao professor Dr. João Inácio da Silva Filho, sem dúvidas um dos maiores expoentes da Lógica Paraconsistente, de quem tive a honra de ser aprendiz durante este tempo.

Finalmente, à equipe da secretaria do Mestrado, Sandra e Imaculada, que foram incansáveis em ajudar na solução de todas as dúvidas e dificuldades, a qualquer momento, sempre com muita simpatia e extrema boa vontade.

RESUMO

Este trabalho apresenta a utilização da Lógica Paraconsistente (LP) em conjunto com técnicas da fotogrametria para a realização do processamento de imagens e obtenção de dados de interesse com aplicação em um cenário de controle de qualidade de uma linha de produção industrial, com resultados encorajadores para o desenvolvimento de novos trabalhos nesta área: uma diminuição de variância na distribuição de frequências de níveis de cinza na imagem 13 vezes superior em comparação à alcançada pela utilização de uma biblioteca de programação de amplo uso, a realização de medições nos objetos com erro máximo observado de 1,77% em relação ao valor teórico esperado, uma taxa de acerto de 100% no processo de classificação dos objetos com a utilização de algoritmos de baixa intensividade computacional e uma capacidade de identificação de desvios nas dimensões do objeto de apenas 1 mm. A fotogrametria tem sido aplicada com sucesso nos últimos anos em cenários de classificação semântica de imagens e na realização de medições, importantes, por exemplo, para o monitoramento da manutenção de estruturas; a LP, por sua vez, tem sido aplicada, entre outras coisas, em diferentes cenários de controles de processo. Uma vez que a LP permite utilizar a incerteza como parâmetro na realização de cálculos, ela é extremamente interessante para cenários de medição – em que o erro é inerente e é propagado. Descrevem-se, aqui, as etapas de montagem e execução de uma estrutura lógica chamada de Unidade Paraconsistente de Convolução (UPC), capaz de realizar transformações em imagens e extrair propriedades de objetos, utilizadas por Células Neurais Artificiais Paraconsistentes (CNAPs) para a posterior realização das etapas de classificação dos objetos presentes nas imagens e identificação de desvios em suas dimensões.

Palavras-Chave: Rede Neural Artificial Paraconsistente. Fotogrametria. Visão Computacional. Lógica Paraconsistente. Controle de Qualidade.

ABSTRACT

This work presents the use of the Paraconsistent Logic (PL) combined with techniques of the photogrammetry, both aimed at processing images to obtain data of interest in an industrial production line quality control scenario with encouraging results for the development of new works in this area: a decrease of the variance in the frequency distribution of grey levels in the image 13 times higher than that achieved by the use of a widely used programming library, the realization of measurements on objects with an observed maximum error of 1.77% in relation to the expected theoretical value, a 100% accuracy rate in the object classification process with the use of low computational-intensive algorithms and an ability to identify deviations in dimensions of the object as low as 1 mm.. Photogrammetry has been successfully applied in recent years in situations of semantic classification of images and in measurements of objects, which are important, for example, for monitoring the maintenance of structures. PL, in turn, has been applied, among other things, in different process control circumstances. Since PL allows uncertainty to be considered as a parameter when performing calculations, it is extremely interesting for scenarios in which measurement plays a key role - where error is inherent and is propagated. Here, the steps for assembling and executing a logical structure denoted the Paraconsistent Convolution Unit (PCU) are described. The PCU can perform transformations in images and is able to extract object properties, which are later used by Paraconsistent Artificial Neural Cells (PANCs) for performing the steps of classifying the objects present in the images and identifying deviations in their dimensions.

Keywords: Paraconsistent Artificial Neural Network. Photogrammetry. Computer vision. Paraconsistent logic. Quality control.

LISTA DE FIGURAS

Figura 1 – Reticulado finito de Hasse na LPA.....	20
Figura 2 – Reticulado finito de Hasse na LPA2v	21
Figura 3 – Representação do intervalo de certeza no reticulado finito de Hasse na LPA2v ..	22
Figura 4 – Representação geométrica do grau de certeza real na LPA2v para $G_c > 0$	23
Figura 5 – Esquema de funcionamento de uma CAPb.....	24
Figura 6 – Esquema de funcionamento de uma câmara estenopeica	27
Figura 7 – Diferentes padrões de CFA para câmeras digitais comerciais	29
Figura 8 – Exemplo de foto colorida.....	29
Figura 9 – Decomposição de foto colorida em canais RGB	29
Figura 10 – Histograma de frequências de níveis de cinza.	32
Figura 11 – Comparação do efeito da convolução de um Filtro Binomial	36
Figura 12 – Exemplos de fotografias utilizando a peça quadrada e a peça retangular.	43
Figura 13 – Fluxograma ilustrativo para o planejamento do processamento das imagens ...	44
Figura 14 – Fluxograma de processamento na UPC.....	45
Figura 15 – Esquema do funcionamento da UC com 6 CNAPap ligadas em série.....	48
Figura 16 – Esquema de funcionamento da CNAPp.....	53
Figura 17 – Esquema de funcionamento da CNAPdi	55
Figura 18 – Exemplos de erros nas imagens capturadas.....	56
Figura 19 – Imagens convertidas para escala de cinza.....	57
Figura 20 – Histogramas de níveis de cinza encontrado para os objetos quadrado (à esquerda) e retangular (à direita).....	57
Figura 21 – Comparação entre imagens após aplicação do Filtro Binomial (utilizando biblioteca-padrão, à esquerda e via MEEBi + MC, à direita).	58
Figura 22 – Histogramas de intensidades de cinza encontrado para a figura aleatória.	59
Figura 23 – Histogramas de intensidades de cinza encontrado para a figura após aplicação do Filtro Gaussiano – via MEEBi + MC (à esquerda) e via biblioteca OpenCV (à direita). ...	59
Figura 24 – Resultado da convolução com o Operador Scharr via biblioteca na direção Δx e Δy	60
Figura 25 – Resultado da convolução com o Operador Scharr via biblioteca após cálculo das magnitudes.	61
Figura 26 – Exemplo de sensibilidade da MEENi ao reflexo do <i>flash</i>	61
Figura 27 – Exemplo de sensibilidade da MEENi na presença de falhas no objeto.	62
Figura 28 – Extração dos contornos via MEENi + UPC + MoBi (antes e após tratamento com processos de dilatação e erosão).	62
Figura 29 – Determinação dos contornos e transformação de perspectiva.	63
Figura 30 – Exemplo de dimensões obtidas para o objeto quadrado.	64
Figura 31 – Exemplo de dimensões obtidas para o objeto retangular.	64
Figura 32 – Matriz de confusão para a classificação realizada na CNAPp.....	66
Figura 33 – Objeto quadrado modificado utilizado no teste da CNAPdi.	68

LISTA DE QUADROS

Quadro 1 – Graus de evidência para o objeto quadrado.....	65
Quadro 2 – Graus de evidência para o objeto retangular.....	66
Quadro 3 – Resultados encontrados nos conjuntos de teste para a classificação da forma geométrica via CNAPp.....	66

LISTA DE ABREVIATURAS E SIGLAS

ADC	–	Conversor Analógico-Digital
BSD	–	<i>Berkeley software distribution</i>
CAPb	–	Célula Artificial Paraconsistente básica
CCD	–	Dispositivo de Carga Acoplada
CEP	–	Controle Estatístico de Processo
CFA	–	<i>Color filter array</i>
CMOS	–	Metal-Óxido-Semicondutor Complementar
CNAP	–	Célula Neural Artificial Paraconsistente
CNAPadm	–	Célula Neural Artificial Paraconsistente de aprendizagem, desaprendizagem e memorização
CNAPca	–	Célula Neural Artificial Paraconsistente de conexão analítica
CNAPclsemax	–	Célula Neural Artificial Paraconsistente de conexão lógica seletiva para maximização
CNAPclsemin	–	Célula Neural Artificial Paraconsistente de conexão lógica seletiva para minimização
CNAPclsmx	–	Célula Neural Artificial Paraconsistente de conexão lógica simples para maximização
CNAPclsmín	–	Célula Neural Artificial Paraconsistente de conexão lógica simples para minimização
CNAPco	–	Célula Neural Artificial Paraconsistente de Complementação
CNAPd	–	Célula Neural Artificial Paraconsistente de Decisão
CNAPdi	–	Célula Neural Artificial Paraconsistente de detecção de igualdade
CNN	–	Redes Neurais Convolucionais (<i>Convolutional Neural Networks</i>)
ISO	–	<i>International Organization for Standardization</i>
IoT	–	Internet das Coisas (<i>Internet of Things</i>)
LoG	–	Laplaciano de uma Gaussiana
LP	–	Lógica Paraconsistente
LPA	–	Lógica Paraconsistente Anotada
LPA2v	–	Lógica Paraconsistente com Anotação de Dois Valores
MC	–	Módulo de Convolução
MEEBi	–	Módulo Extrator de Evidência Binomial inicial
MEEf	–	Módulo Extrator de Evidência final

MEEi	–	Módulo Extrator de Evidência intermediária
MEENi	–	Módulo Extrator de Evidência Nablaiana inicial
MLP	–	Percéptron de Multicamadas
MoBi	–	Módulo de Binarização
NAP	–	Nó de Análise Paraconsistente
NIR	–	Infravermelho Próximo (<i>Near-Infrared</i>)
QUPC	–	Quadrado Unitário no Plano Cartesiano
RGB	–	Vermelho, Verde e Azul
RNA	–	Rede Neural Artificial
RNAP	–	Rede Neural Artificial Paraconsistente
SNAP	–	Sistema Neural Artificial Paraconsistente
SNAPac	–	Sistema Neural Artificial Paraconsistente de Aprendizado Condicionado
SNAPtc	–	Sistema Neural Artificial Paraconsistente de Tratamento de Contradições
UNAP	–	Unidade Neural Artificial Paraconsistente
UNAPEMM	–	Unidade Neural Artificial Paraconsistente de Extração de Média Móvel
UPC	–	Unidade Paraconsistente de Convolução

LISTA DE SÍMBOLOS

$\perp$	–	Indeterminação / Para completude
φ	–	Intervalo de certeza
λ	–	Grau de evidência desfavorável
μ	–	Grau de evidência favorável
μ_{ER}	–	Grau de evidência resultante
μ_r	–	Saída resultante de uma CNAP
F	–	Falso
F_A	–	Fator de aprendizagem
F_{t_c}	–	Fator de tolerância à certeza
$F_{t_{co}}$	–	Fator de tolerância à complementação
$F_{t_{ct}}$	–	Fator de tolerância à contradição
F_{td}	–	Fator de tolerância à decisão
G_C	–	Grau de Certeza
G_{CR}	–	Grau de Certeza real
G_{CT}	–	Grau de Contradição
P	–	Proposição p
Q	–	Proposição q
T	–	Inconsistência
V	–	Verdadeiro
V_{icc}	–	Valor inferior de controle da certeza
V_{icct}	–	Valor inferior de controle da contradição
V_{scc}	–	Valor superior de controle da certeza
V_{scct}	–	Valor superior de controle da contradição

SUMÁRIO

1	INTRODUÇÃO	15
1.1	Importância.....	15
1.2	Problemática.....	17
1.3	Fundamentação teórica	19
1.3.1.	Lógica Paraconsistente.....	19
1.3.2.	RNAPs.....	23
1.3.3.	Imagens digitais.....	26
1.3.3.1.	Formação de imagens em câmeras digitais	26
1.3.3.2.	Modelos matemáticos para representação de imagens digitais.....	30
1.3.3.3.	Transformações em imagens digitais	32
1.4	Justificativa	38
1.5	Objetivo	40
1.5.1.	Objetivo geral	40
1.5.2.	Objetivos específicos.....	40
2	MATERIAIS E MÉTODOS	42
2.1	Materiais.....	42
2.1.1.	Captura de imagens e preparo do ambiente de desenvolvimento.....	42
2.2	Método	43
2.2.1.	Planejamento.....	43
2.2.2.	Desenvolvimento da Unidade Paraconsistente de Convolução (UPC).....	44
2.2.3.	Fluxo de processamento de imagens na UPC	46
2.2.4.	Módulo de Convolução – MC.....	47
2.2.5.	Suavização de ruídos: Módulo Extrator de Evidência Binomial inicial – MEEBi	48
2.2.6.	Identificação de Contornos: Módulo Extrator de Evidência Nablaiana inicial – MEENi	50
2.2.7.	Módulo Extrator de Evidência Intermediária (MEEi) e Módulo Extrator de Evidência final (MEEf)	51
2.2.8.	Extração de Camadas e retirada de outliers: Módulo de Binarização – MoBi	52
2.2.9.	CNAP de classificação.....	52
2.2.10.	CNAP de detecção de desvios.....	54
3	RESULTADOS E DISCUSSÃO	56
3.1	Seleção das imagens capturadas e conversão para cinza.....	56
3.2	Histogramas de frequências dos níveis de cinza e cálculo da variância	57
3.3	Aplicação do Filtro Binomial.....	58
3.4	Aplicação do Filtro Gradiente, binarização e identificação de contornos.....	60
3.5	Extração das propriedades	63

3.6	Classificação dos objetos	65
3.7	Identificação de desvios.....	67
3.8	Discussão.....	68
4	CONCLUSÕES.....	72
4.1	Trabalhos futuros.....	73
	REFERÊNCIAS	74
	APÊNDICE A – Algoritmo da Célula Neural Artificial Paraconsistente padrão	78
	APÊNDICE B – Algoritmo da Célula Neural Artificial Paraconsistente de detecção de igualdade.....	81

1 INTRODUÇÃO

1.1 Importância

Tentar introduzir o conceito do monitoramento óptico de materiais e estruturas a partir de uma definição geral e abrangente é uma tarefa árdua, visto este abranger uma série de técnicas diversas, utilizadas e adaptadas conforme a necessidade do problema que se deseja solucionar. Comuns a todas elas, estão algumas das suas características que as permitem serem agrupadas em um mesmo conjunto e que fazem deste tipo de abordagem tão relevante atualmente: sua capacidade de monitorar (e daí seu nome) e extrair propriedades sobre determinado objeto-alvo de forma remota, em tempo muito próximo ao real e, normalmente, de forma não-destrutiva, com base em interações com a luz ao longo de diferentes comprimentos de onda, fatores estes que trazem vantagens econômicas e estratégicas bastante importantes para as organizações.

Se, por um lado, a extração de propriedades é a atividade direta destes tipos de monitoramento, a transformação dos dados de observação em informações, utilizadas como insumos para tomada de decisão, representam sua atividade-fim, visto serem estas a justificativa de seu uso – e daqui decorrem todas as vantagens competitivas alcançadas pelas características dos sistemas remotos, capazes de levantar dados de forma quase que instantânea.

Indubitavelmente, seu objetivo (seja numa questão de segurança operacional, preservação de patrimônio ou mesmo atividades de classificação dentro de um contexto tributário, por exemplo) está atrelado a outro termo bastante utilizado, mas difícil de ser explicado: a “qualidade”; esta, podendo se referir a diferentes acepções – indo de uma característica a um processo, ou mesmo podendo se referir a um departamento dentro de uma empresa ou uma cultura organizacional.

De forma bastante particular, as diferentes disciplinas da qualidade estão muito atreladas aos campos de aplicação da Engenharia Mecânica, visto serem considerados aqui os aspectos de eficiência e eficácia operacionais em estruturas mecânicas, como o controle do processo de materiais brutos até o produto finalizado, a prevenção à ocorrência de falhas mecânicas de diferentes gravidades (e consequentes tempos de parada) e a qualidade do processo como um todo – que, entre outras coisas, envolve o uso adequado de materiais e condições operacionais ótimas, definidas pelo engenheiro responsável.

Não chega a ser surpresa, neste ponto, que as principais ferramentas e metodologias de qualidade de uso corrente tenham sido desenvolvidas e evoluído dentro dos departamentos de engenharia nas indústrias; instrumentos extremamente relevantes atualmente, como o uso do Ciclo PDCA, o *Lean Manufacturing* e o Seis Sigma (*Six Sigma*), por exemplo, que reforçam a importância da abordagem da qualidade em disciplinas da Engenharia Mecânica, com o que este trabalho busca contribuir.

A norma brasileira ABNT NBR ISO 9000:2015, documento de referência em gestão de qualidade, a identifica como sendo o “grau no qual um conjunto de características inerentes satisfaz a requisitos”, considerando, aqui, não apenas a “adequação ao uso”, mas também a “conformidade com as exigências” (legais ou não) sobre determinado bem ou serviço (ABNT, 2015).

A qualidade emerge, em nossos tempos, a partir de um longo processo histórico, resultado da evolução das sociedades: de artesãos – que produziam objetos sob medida, de acordo com os desejos de seus consumidores – até modelos mais recentes, adotados em linhas de produção em massa.

Paladini (2011) comenta que a produção “sob medida” pode ser acompanhada até fins do século XIX, exemplificando com o caso de montadoras produzindo veículos sob demanda sendo a qualidade, em suas palavras, “inerente ao processo, porém vista de forma individual, com foco na customização e não na padronização”. Este fenômeno é visto até hoje em nichos específicos e, sem dúvida, representa um tipo de controle de padrões de qualidade em cenários em que é aplicável.

A produção em massa, entretanto, necessária em uma economia capitalista e global, demandou o desenvolvimento de novos meios de gerenciar os processos; um de seus corolários foi o desenvolvimento da administração como ciência. Trabalhos sobre a racionalização do trabalho e estudos sobre tempos e movimentos de Taylor, dos controles de produção de Gantt e da departamentalização de Fayol, por exemplo, deram origem à “inspeção do serviço” como atividade descentralizada do processo da fabricação – e, portanto, criando a “qualidade” como uma área apartada das demais.

Martinelli (2009) comenta que os primeiros métodos de controle estatístico de qualidade foram adotados entre as décadas de 1920 e 1930, como resposta aos trabalhos de Walter Shewhart. O uso da estatística, aqui, apresenta uma importante vantagem competitiva por permitir não apenas o controle em si, mas também, o

monitoramento e a melhoria das etapas de verificação.

Na década de 1950, já no pós-guerra, observaram-se movimentações durante a reestruturação das indústrias japonesas. Noções como as do “Controle da Qualidade Total” e do “Gerenciamento da Qualidade Total”, por exemplo, passaram a serem utilizadas, e a trazerem resultados expressivos neste processo – tornando o Japão um país de referência em qualidade, possibilitando a seus produtos concorrerem com indústrias tradicionais americanas.

A partir da década de 1980 viveu-se um movimento de implementação de “sistemas de gestão da qualidade” nas organizações, que culminou, em 1987, com a publicação pela *International Organization for Standardization* (ISO) de uma série de normas de adoção voluntária que apresentaram uma série de modelos unificados de Sistema de Garantia da Qualidade – a série de normas ISO 9000.

Apesar de não ser uma ciência recente, o processo de controle de qualidade é cada vez mais pertinente dentro de um contexto de um mercado global, com grande competitividade, em que fatores como a inovação, a eficácia e eficiência tanto do produto como de seus processos produtivos possuem grande peso.

1.2 Problemática

A proposta apresentada nesta dissertação é a da aplicação do monitoramento óptico dentro de um cenário de controle de processo operacional através do processamento de imagens obtidas por meio fotográfico.

Aqui, a partir de imagens obtidas de figuras geométricas regulares, utilizadas como a simplificação de objetos que, de forma factível, seriam produzidos em determinado processo de manufatura, busca-se mostrar uma alternativa aplicada a um dos processos de monitoramento óptico possíveis em um cenário de Controle Estatístico de Processo (CEP), método de larga aplicação na indústria.

Considerando que uma das premissas do CEP de Shewhart seja a da realização do “controle da variabilidade” – ou, segundo Montgomery (2012), o fato de “a qualidade ser inversamente proporcional à variabilidade” –, a atividade de monitoramento do processo passa a ser visto com importância. Neste ínterim, a utilização de técnicas automatizadas não-destrutivas e de pouco ou nenhum contato com o produto, como é o caso das técnicas ópticas – e entre elas a fotogrametria – também se tornam relevantes.

Ainda que o cenário apresentado esteja considerando este contexto, a

aplicação das técnicas aqui apresentadas pode ser adaptada e utilizada em outras frentes como, por exemplo, no Monitoramento de Saúde Estrutural, um uso bastante nobre para o qual este tema pode voltar a ser explorado.

Conclusões tomadas baseadas em técnicas de monitoramento óptico fundamentam-se no reconhecimento de padrões e identificação de desvios, quando ocorrem, e, aqui, a execução de algoritmos computadorizados possui importante aplicação. As Redes Neurais Artificiais (RNAs) são um exemplo das técnicas da informática utilizadas neste processo já há alguns anos; Cruz (2015), por exemplo, mencionou a utilização de RNAs em um CEP em trabalho publicado ainda no final do século passado, que já apresentava bons resultados.

Redes Neurais Artificiais Paraconsistentes (RNAPs) são uma classe de RNAs desenvolvidas a partir do trabalho de Da Silva Filho e Abe (2001) e se baseiam na utilização da Lógica Paraconsistente (LP), cuja principal característica é a de aceitar a contradição como proposição lógica, considerando variáveis não-lineares, incertas, imprecisas ou mesmo inconsistentes (DA SILVA FILHO, 2020).

Conquanto que o trabalho de Cruz (2015) tenha sido pioneiro na apresentação da utilização de RNAPs em um cenário de CEP, capaz de avaliar dados em padrões gerados computacionalmente, outras publicações, como as de Damin e Da Silva Filho (2012) e, mais recentemente, de Rocha (2019), também demonstraram a possibilidade da utilização da LP em análises estatísticas em âmbito industrial.

Gonçalves (2019), por sua vez, apresentou a utilização da LP na análise e tomada de decisão na escolha de materiais a serem montados em protótipos de pré-produção em empresa montadora de veículos, enquanto Lopes (2009) apresentou a utilização de uma RNAP no processamento de eletroencefalogramas para diagnóstico de Alzheimer graças ao reconhecimento de padrões. Souza (2013) utilizou a mesma técnica para a realização de reconhecimento de caracteres.

Este trabalho descreve a utilização de uma RNAP no processamento de imagens, com base em técnicas da fotogrametria, com fins de utilização no monitoramento de insumos – capaz de identificar propriedades e realizar classificações de interesse para os processos industriais envolvidos, realizando inferências sobre as formas do material e o controle de suas dimensões, fatores críticos na aplicação do CEP em uma linha de produção.

1.3 Fundamentação teórica

Para o correto entendimento deste trabalho alguns itens precisarão ser revisitados, de forma a apresentar suas bases teóricas e científicas. Isto permitirá ao leitor um melhor esclarecimento durante a leitura, o que tornará seu acompanhamento mais proveitoso.

1.3.1. Lógica Paraconsistente

A lógica aristotélica, clássica, estabelece que determinada proposição pode ser rotulada como verdadeira ou falsa, não admitindo outros valores ou qualquer tipo de combinação destes – ainda que parcial. Segundo suas bases, uma proposição qualquer deve satisfazer a três princípios fundamentais:

- a) Princípio da identidade ($p \rightarrow p$) – indica que toda valoração de proposição ou objeto implica nela mesma;
- b) Princípio do terceiro excluído: ($p \vee \neg p$) – indica que, quando existirem duas proposições contraditórias, uma delas será verdadeira;
- c) Princípio da não contradição: $\neg (p \vee \neg p)$ – indica que, quando existirem duas proposições contraditórias, uma delas será falsa.

Este modelo, binário, vem sendo aplicado com sucesso ao pensamento ocidental desde o início da concepção da Lógica como ciência, com os trabalhos de Aristóteles (384-322 a.C), e sua aplicação foi exitosa ao permitir o desenvolvimento de sistemas eletrônicos. A aparição de tecnologias mais modernas, como a robótica e a inteligência artificial, e o surgimento de problemas de maior complexidade, no entanto, colocou em evidência a necessidade da utilização de um novo paradigma lógico que permitisse uma melhor representação da realidade, melhorando as respostas deste tipo de aplicação (DA SILVA FILHO, 2020).

Dentre os diferentes estudos de lógicas não-classicas, a Lógica Paraconsistente (LP) aparece a partir dos trabalhos de Newton da Costa, matemático brasileiro que, baseando-se em pesquisas de Lukasiewicz e Vasilév, sobre sistemas lógicos que aceitam a contradição como um princípio lógico, elaborou a teoria que a originou – apesar de seu termo ter sido cunhado apenas durante a década de 1970, por Francisco Miró Quesada Cantuarias, durante simpósio ocorrido na Universidade Estadual de Campinas.

A LP tem a capacidade de trabalhar com situações em que uma proposição e sua negação possam ser consideradas verdadeiras, além de incluir dois outros estados lógicos à lógica clássica: a inconsistência (T) e a indeterminação ($\perp$) – também chamado de “estado paraCompleto”.

A Lógica Paraconsistente Anotada (LPA) surge a partir da LP, já na década de 1990, com a publicação “*The Paraconsistent Logics PJ*” de Da Costa et al. (1991). A LPA considera os quatro estados lógicos da LP dentro do conjunto $\tau = \{T, V, F, \perp\}$ representando os quatro vértices de um reticulado finito em um diagrama de Hasse. Nesta representação, a proposição P é acompanhada com a anotação (ou valor correspondente) composta por um grau de evidência favorável (μ), uma constante anotacional do reticulado, que indica de forma a conotação lógica de determinada proposição. O diagrama geral do reticulado de Hasse com a representação dos estados lógicos é mostrado na figura 1.

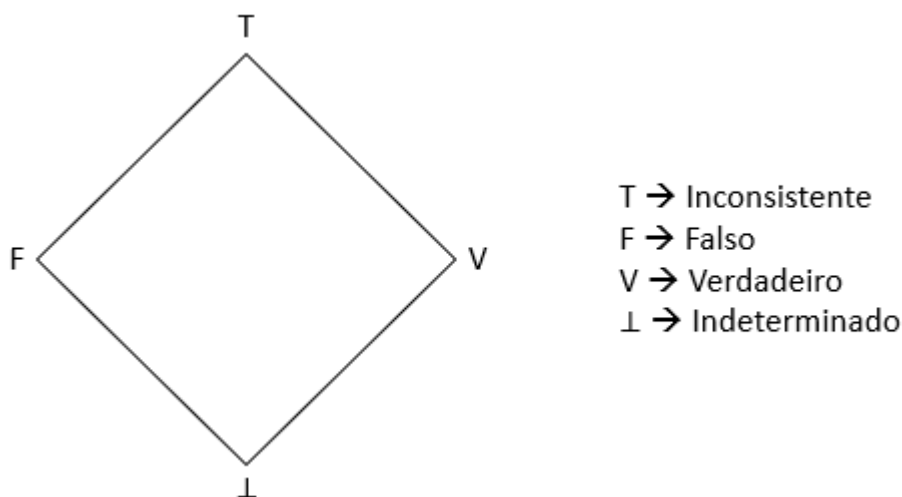


Figura 1 – Reticulado finito de Hasse na LPA
 Fonte: Adaptado de DA SILVA FILHO et al. (2008).

A interpretação dada para determinada proposição, acompanhada de seu respectivo grau de evidência dentro da LPA, atribui uma conotação lógica de verdade, falsidade inconsistência ou indeterminação a esta – assim, uma anotação $P(V)$ atribui conotação de verdade para a proposição, enquanto a anotação $P(\perp)$ atribui conotação de indeterminação (paraCompleto) para ela, ambos acompanhados de seu grau de evidência μ .

A Lógica Paraconsistente Anotada com anotação de 2 valores (LPA2v) diferencia-se da LPA principalmente pelo fato de considerar, além do grau de

evidência favorável, um grau de evidência desfavorável (λ), que rejeita a proposição – este, independente do primeiro. Neste caso, o reticulado finito de Hasse passa a ser definido por pares ordenados de acordo com a relação $\tau = \{(\mu, \lambda) \mid \mu, \lambda \in [0, 1] \subset \mathfrak{R}\}$, sendo válida a relação $\sim(\mu, \lambda) = (\lambda, \mu)$, em que $\mu, \lambda \in [0, 1] \subset \mathfrak{R}$ em que cada um dos vértices do reticulado de Hasse é representado por um par ordenado no intervalo $[0, 1]$, tendo as seguintes combinações como valores-limites:

- a) $P(1,0)$ – Indica evidência favorável total e evidência desfavorável nula. A proposição é verdadeira;
- b) $P(0,1)$ – Indica evidência favorável nula e evidência desfavorável total. A proposição é falsa;
- c) $P(1,1)$ – Indica evidência favorável total e evidência desfavorável total. A proposição é inconsistente;
- d) $P(0,0)$ – Indica evidência favorável nula e evidência desfavorável nula. A proposição é indeterminada (paracompleta).

A figura 2 mostra a representação do reticulado correspondente à LPA2v para a anotação composta por dois valores $P(\mu, \lambda)$.

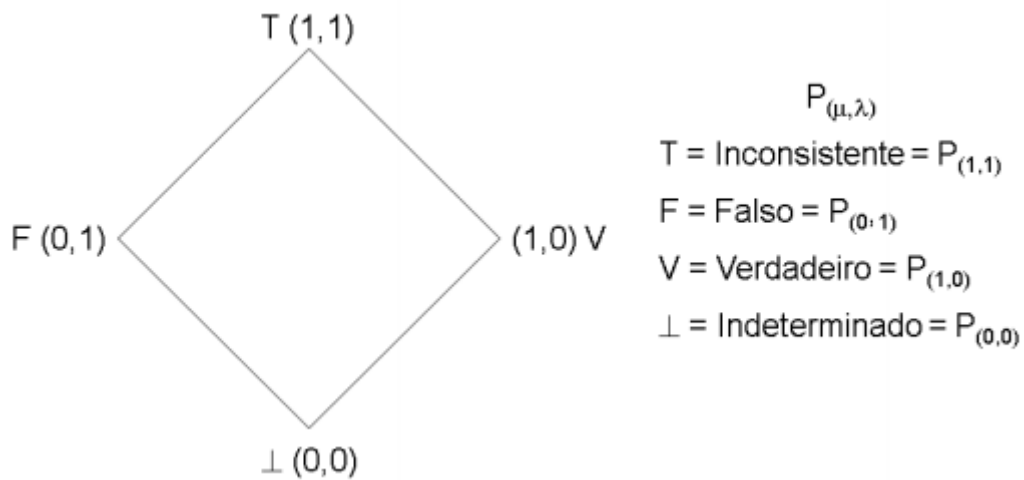


Figura 2 – Reticulado finito de Hasse na LPA2v
 Fonte: Adaptado de DA SILVA FILHO et al. (2008).

A aplicação de transformações lineares da representação do Reticulado de Hasse em um Quadrado Unitário no Plano Cartesiano (QUPC) mostra válida a transformação $T(x, y) = (x - y, x + y - 1)$ e permite, como consequência, a definição dos termos “Grau de Certeza” (G_c) e “Grau de Contradição” (G_{ct}), definidos nas

equações 1 e 2, pertencentes ao conjunto $\mathfrak{R}$ e representados no reticulado de Hasse nos eixos horizontal e vertical, respectivamente.

$$G_c = \mu - \lambda \quad (1)$$

$$G_{ct} = \mu + \lambda - 1 \quad (2)$$

Qualquer valor de G_{ct} diferente de 0 implica a existência de um valor máximo para G_c diferente do grau máximo possível de certeza, isto é, 1. Este valor, determinado por φ , chamado de “intervalo de certeza”, é calculado conforme a equação 3, e seu efeito sobre os valores máximos e mínimos de G_c (cujos valores equivalem, respectivamente, a $+\varphi$ e $-\varphi$) é ilustrado na figura 3.

$$\varphi = 1 - |G_{ct}| \quad (3)$$

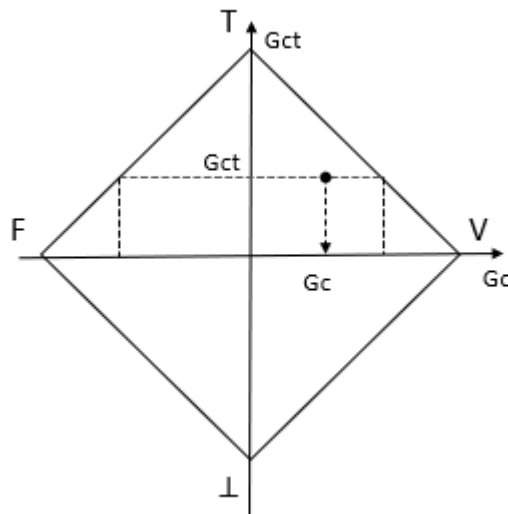


Figura 3 – Representação do intervalo de certeza no reticulado finito de Hasse na LPA2v

Fonte: Adaptado de DA SILVA FILHO et al. (2008).

O resultado desejável de uma análise que utilize a LPA2v deve representar o valor de certeza não contaminado pelos efeitos das incertezas – capaz, portanto, de subtrair o efeito das inconsistências originadas pelas informações conflitantes do grau de certeza das observações. Quando um par de graus de evidência μ e λ é *plotado* em um reticulado de Hasse, alguns valores de controle passam a ser importantes, como os casos dos valores superiores (máximos) e inferiores (mínimo) possíveis para

a certeza dada a contradição e da contradição, dada a certeza, definidos como V_{scc} (valor superior de controle da certeza, V_{icc} (valor inferior de controle da certeza), V_{scct} (valor superior de controle da contradição) e V_{icct} (valor inferior de controle da contradição), que indicam os valores máximos e mínimos aceitáveis no sistema para que determinada proposição seja considerada falsa ou verdadeira.

Após o tratamento das informações do LPA2v para remoção destes efeitos chega-se ao valor do chamado grau de certeza de valor real (G_{cr}), que considera o valor de interpolação do reticulado em um estado lógico paraconsistente $\varepsilon\tau(G_c, G_{ct})$ e o segmento de reta que vai do ponto máximo de G_c (quando G_c é positivo) ou ao valor mínimo de G_c (quando este é negativo), segundo relação mostrada na equação 4. A representação geométrica do G_{cr} é mostrada na figura 4.

$$G_{cr} = \begin{cases} (1 - D), & \text{se } G_c > 0 \\ (D - 1), & \text{se } G_c < 0 \end{cases} \text{ onde } D = \sqrt{(1 - |G_c|)^2 + G_{ct}^2} \quad (4)$$

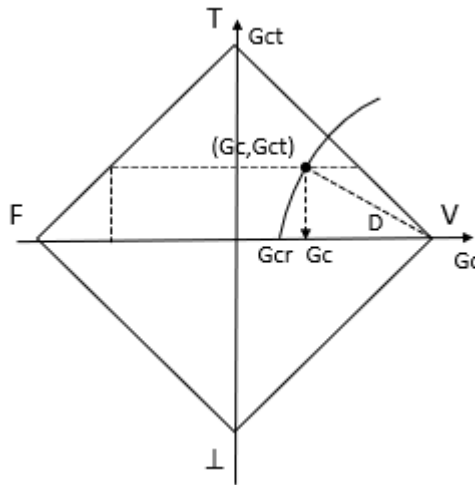


Figura 4 – Representação geométrica do grau de certeza real na LPA2v para $G_c > 0$
 Fonte: Adaptado de DA SILVA FILHO et al. (2008).

1.3.2. RNAPs

Uma RNAP se baseia na teoria das RNAs clássicas, porém, apresenta características distintas – entre elas o fato de usar a LPA2v em sua construção e funcionamento (LOPES, 2009), e por utilizar diferentes tipos de unidades básicas – ao contrário de uma RNA clássica que, normalmente, possui apenas um tipo de neurônio.

A unidade neural básica de uma RNAP é a chamada Célula Neural Artificial Paraconsistente (CNAP) e esta pode ter diferentes configurações, utilizando uma equação estrutural básica (EEB) para equacionar sinais de entrada e convertê-los em um sinal de saída, transmitido para outras CNAPs.

Uma CNAP é formada por uma Célula Artificial Paraconsistente básica (CAPb) modificada, caracterizada por receber dois sinais de entrada: um representando o valor de entrada μ da LPA2v, e o outro o valor λ ; sua saída μ_r , chamada de saída resultante, é calculada pelos valores de G_c e G_{ct} através da execução das análises da LPA2v.

Um esquema da CAPb é mostrado na figura 5, em que μ representa o grau de evidência favorável, λ o grau de evidência desfavorável, V_{scc} o valor superior de controle da certeza, V_{icc} o valor inferior de controle da certeza, V_{scct} o valor superior de controle da contradição, V_{icct} o valor inferior de controle da contradição, G_c o grau de certeza, G_{ct} o grau de contradição e o conjunto $\{V, F, I\}$ as saídas lógicas possíveis – Verdadeiro, Falso e Indeterminado.

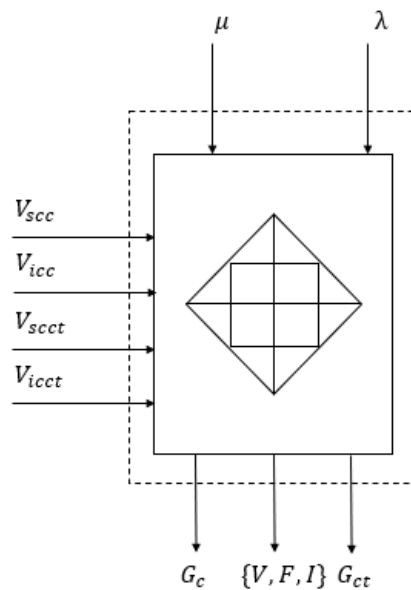


Figura 5 – Esquema de funcionamento de uma CAPb

Fonte: Adaptado de DA SILVA FILHO et al. (2008).

A primeira modificação realizada em uma CAPb para que seja considerada uma CNAP é a normalização de sua saída, para garantir seu pertencimento ao intervalo real $[0,1]$, de acordo com a relação da equação $\mu_r = \frac{(\mu - \lambda + 1)}{2}$.

A segunda diferença em uma CAPb é a existência da “análise da tolerância”. De acordo com Lopes (2009), em alguns níveis da RNAP são necessárias apenas as

respostas verdadeira, falsa e indefinida – sendo a inconsistência ignorada e, no caso da indefinição, há de se entender se esta ocorre por insuficiência de informações ou por excesso de contradição. Estas definições são possíveis graças à existência dos fatores de tolerância que, em uma CNAP são: o fator de tolerância à contradição (Ft_{ct}), o fator de tolerância à certeza (Ft_c), o fator de tolerância à decisão (Ft_d) e o fator de aprendizagem (F_A), todos valores externos de ajuste que variam no intervalo $[0,1]$ (DA SILVA FILHO, 2008).

De modo especial, o F_A tem como função ajustar a rapidez com que uma CNAP aprenderá, variando conforme a seguinte classificação:

- a) $F_A = 0,0$ – não há aprendizado;
- b) $F_A = 0,5$ – aprendizado natural;
- c) $F_A = 1,0$ – função degrau;

Diferentes CNAPs podem ser unidas para formar diferentes unidades neurais artificiais paraconsistentes (UNAPs). CNAPs podem ser de diferentes tipos, entre eles, Lopes (2009) cita:

- a) Célula Neural Artificial Paraconsistente de conexão analítica (CNAPca) – principal célula de toda RNAP, permite calcular o G_c e o G_{ct} a partir das entradas e dos fatores de tolerância;
- b) Célula Neural Artificial Paraconsistente de conexão lógica simples para maximização (CNAPclsmx) – permite selecionar o valor máximo entre as entradas;
- c) Célula Neural Artificial Paraconsistente de conexão lógica simples para minimização (CNAPclsmn) – permite selecionar o valor mínimo entre as entradas;
- d) Célula Neural Artificial Paraconsistente de conexão lógica seletiva para maximização (CNAPclsemax) – permite, além da seleção do valor máximo entre as entradas, escolher o caminho que a informação segue, desviando o fluxo do processamento para uma camada diferente da RNAP;
- e) Célula Neural Artificial Paraconsistente de conexão lógica seletiva para minimização (CNAPclsemin) – permite, além da seleção do valor mínimo entre as entradas, escolher o caminho que a informação segue, desviando o fluxo do processamento para uma camada diferente da RNAP;

- f) Célula Neural Artificial Paraconsistente de complementação (CNAPco) – permite efetuar o cálculo do complemento da entrada, considerando o fator de tolerância à complementação (Ft_{co}) que define se a saída é indefinida ou se pode conter o complemento da entrada;
- g) Célula Neural Artificial Paraconsistente de decisão (CNAPd) – permite identificar a interpretação do valor obtido, podendo ser verdadeiro, falso ou outro, de acordo com a tabela de constantes anotacionais definida;
- h) Célula Neural Artificial Paraconsistente de aprendizagem, desaprendizagem e memorização (CNAPadm) – permite a aprendizagem, desaprendizagem e/ou a memorização de padrões;

Quando CNAPs se unem, elas tomam a capacidade de formarem sistemas neurais artificiais paraconsistentes (SNAPs) e estes, em conjunto, formam uma RNAP. Ainda de acordo com Lopes (2009) as SNAPs podem ser:

- a) SNAPs de aprendizado condicionado (SNAPac) – configurados de maneira a processar dados conforme as leis de aprendizado de Hebb – condicionadas a apresentarem padrões de saída obtidos através de repetições de coincidências;
- b) SNAPs de tratamento de contradições (SNAPtc) – promovem o tratamento das contradições entre sinais de informações com base na LPA, recebendo três sinais de entrada e tendo como saída um sinal de consenso entre eles de tal forma que na saída predomina o valor proposto pela maioria.

1.3.3. Imagens digitais

1.3.3.1. Formação de imagens em câmeras digitais

O modelo mais conhecido de representação da formação da imagem em uma câmera é o da chamada “câmera estenopeica” (ou *pinhole*), em que uma câmara escura, com um pequeno orifício para permitir a passagem dos raios de luz, contendo um filme em seu interior é exposta por um tempo determinado frente a um objeto até que a imagem seja impressa no filme, tal como ilustrado na figura 6, um processo totalmente dependente da interação da luz.

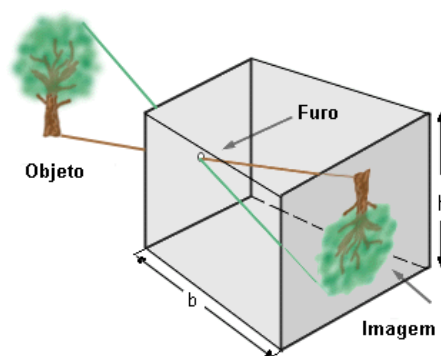


Figura 6 – Esquema de funcionamento de uma câmara estenopeica

Fonte: <http://portaldoprofessor.mec.gov.br/>. Acesso em 10/01/2021.

Câmeras digitais, por sua vez, mantêm o princípio da interação com a luz como base para seu funcionamento, porém têm seu funcionamento um pouco modificado.

Stachniss (2015) apresenta uma relação resumida dos principais componentes de uma câmera digital, com suas respectivas funções. São eles:

- a) Lentes – utilizadas no aumento da eficiência de captação da luz;
- b) Abertura do diafragma – utilizada no controle de passagem da luz para ser processada;
- c) Obturador – com função de controle do tempo de exposição da imagem à câmera;
- d) Sensor – acoplado a um chip, tem função de medir a intensidade da luz (também entendida como o número de fótons) que atingem cada parte de sua célula;
- e) Conversor analógico-digital (ADC) – converte as intensidades de luz medidas pelo sensor e transmitidas via impulsos elétricos, analógicos, para sinais digitais, discretos e quantizados.

São dois os tipos de sensores mais comumente encontrados no trabalho com câmeras digitais dentro do espectro visível: o CCD (dispositivo de carga acoplada) e o CMOS (metal-óxido-semicondutor complementar); a principal diferença entre ambos diz respeito ao momento em que a conversão ACD ocorre – sendo ela realizada fora do sensor no caso do CCD, enquanto no CMOS esta carga é realizada em cada pixel, no próprio sensor, de forma individual.

Se por um lado o CCD é capaz de construir imagens com menos ruídos (uma vez que os sinais de todos os pixels são processados antes de a imagem ser gerada), ele se torna uma tecnologia mais cara; já o CMOS é mais barato, porém, produz imagens com mais ruídos, o que é um contraponto a ser levado em conta na escolha

do sensor de trabalho.

O tamanho dos sensores é outra propriedade importante: quanto maior o sensor utilizado, mais luz pode ser capturada e processada por intervalo de tempo e, logo, maior a qualidade da imagem obtida. Ao mesmo tempo, o espaçamento entre cada uma das suas células (o chamado *sampling pitch*) influencia na resolução da fotografia, por permitir uma maior distribuição na captura dos fótons – diminuindo o ruído advindo das suas interações.

Não menos importante, o número de sensores também possui influência na qualidade da imagem capturada por uma câmera digital: câmeras com apenas um sensor trabalham com filtros de cores individuais em regiões específicas, sendo que cada área da célula poderá capturar apenas uma cor, enquanto as outras cores componentes daquele mesmo espaço são normalmente determinadas por interpolação cromática – o que tem como consequência uma menor qualidade nas imagens.

Alguns padrões pré-definidos de filtros são amplamente aplicados na indústria, e são chamados genericamente de mosaicos de filtros de cores (do inglês “*color filter array*” – CFA). Possivelmente o mais conhecido dos CFAs seja o Filtro de Bayer, que faz o arranjo de cores baseados em canais de intensidades de vermelho, verde e azul (em inglês, *Red*, *Green* e *Blue* e, por isto, conhecido pela abreviação RGB) a partir de um padrão de distribuição de cores de 50% para o verde, 25% para o vermelho e 25% para o azul.

Diferentes padrões comerciais utilizados se baseiam não apenas no Filtro de Bayer, e podem incluir cores como o amarelo, diferenciar o azul claro do azul escuro e trabalhar também com tons de violeta, por exemplo.

Câmeras com três chips possuem valor de venda superior, porém, têm como contrapartida a capacidade de criarem imagens com menor ruído (já que cada chip é dedicado a um único canal de cor). Baseados no arranjo de Bayer, em que imagens são decompostas em diferentes valores de RGB, estes dispositivos permitem a diminuição dos efeitos da interpolação cromática no resultado.

A figura 7 ilustra alguns filtros comerciais existentes, com base no Filtro de Bayer em que W representa o pixel branco, B o pixel azul, G o pixel verde e R o pixel vermelho.

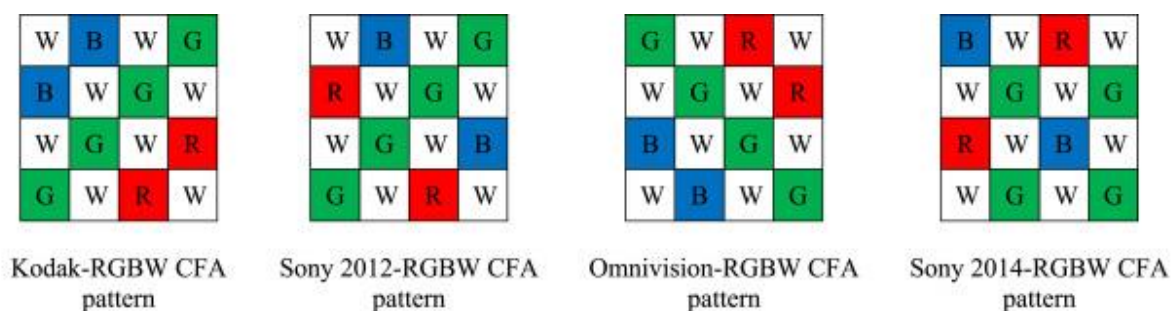


Figura 7 – Diferentes padrões de CFA para câmeras digitais comerciais

Fonte: Singh e Singh (2020).

A figura 8 apresenta uma foto colorida aleatória, enquanto a figura 9 mostra sua decomposição em canais RGB (um para cada cor) realizada através da biblioteca OpenCV 4.2.0 utilizando linguagem Python 3.8.5, considerando o padrão de Filtro de Bayer.



Figura 8 – Exemplo de foto colorida

Fonte: Autor.

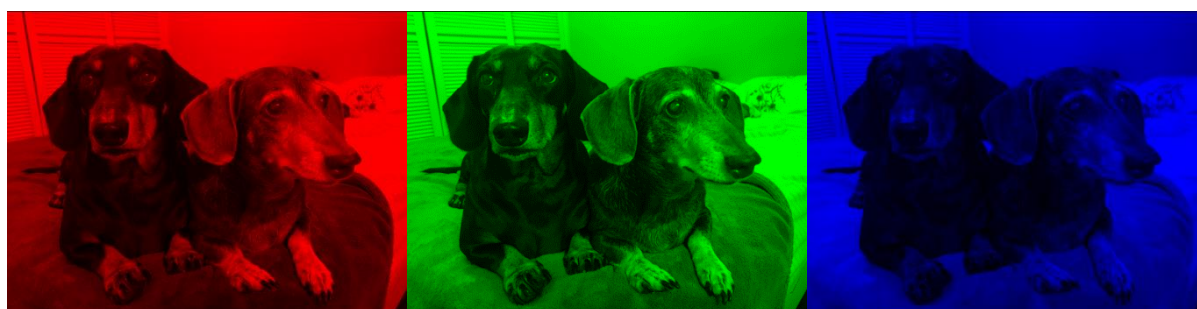


Figura 9 – Decomposição de foto colorida em canais RGB

Fonte: Autor.

Imagens são armazenadas eletronicamente em formatos pré-definidos, que também influem para a sua qualidade.

Formatos comuns, tais como JPG ou PNG, por exemplo, têm capacidade de armazenamento de 1 byte para cada cor – um total de 256 (2^8) diferentes valores possíveis de intensidades, enquanto a maioria dos chips de câmeras trabalha em um formato bruto, com utilização de 2 bytes por canal – podendo, portanto, trabalhar com

65.636 valores de intensidades diferentes por cor.

1.3.3.2. Modelos matemáticos para representação de imagens digitais

Fica claro, pelo exposto anteriormente, a existência de uma correlação direta entre o número de fótons incidentes no sensor com o valor de intensidade dos pixels representados digitalmente, podendo-se estabelecer a relação geral $N \propto I$ (em que o símbolo $\propto$ representa, na matemática, a proporcionalidade), apesar de diferentes escalas poderem ser utilizadas.

O número de fótons incidentes apresenta uma relação entre a quantidade média de fótons que atingem determinada área por intervalo de tempo – chamado também de fluxo de fótons – $b(\lambda)$ – a área do sensor F e sua eficiência $q(\lambda)$. Esta relação é mostrada na equação 5, em que N representa o número de fótons, Δt o tempo de exposição e sendo λ o comprimento de onda específico para determinada cor.

$$N = F\Delta t \int q(\lambda)b(\lambda) d(\lambda) \quad (5)$$

Imagens são formadas a partir de níveis discretos e quantizados de intensidade em diferentes pixels (também chamados de “níveis de cinza”), cuja relação matemática válida é $g(i, j): B \rightarrow G$ em que B representa a matriz da imagem (em linhas e colunas), G as intensidades de nível de cinza sendo $G \in \mathbb{N}$, no intervalo $[0, 255]$ e $\rightarrow$ indicando a existência de uma função parcial de B para G . Esta também é a definição dada por Förstner e Wrobel (2016), que afirmam “as imagens serem matrizes de números naturais representando a aproximação de uma função contínua de intensidades, sob condições de discretização e quantização”.

Estatisticamente as propriedades dos números de fótons envolvidos na formação de uma digital podem ser modeladas através de uma distribuição de Poisson (isto devido aos efeitos de discretização / quantização), cuja probabilidade P é calculada pela equação 6, em que β representa o número médio de fótons incidentes em um segundo e t representa o tempo de exposição.

$$P(N) = \frac{(\beta t)^N}{N!} e^{-\beta t} \quad (6)$$

Corolário direto da aplicação da equação 6 sob a imagem digital é a constatação de que a média possa ser calculada pela relação $\mu_N = \beta t$ enquanto a variância é calculada por $\sigma_N^2 = \beta t$ (representando o mesmo valor). Como consequência, a acurácia relativa pode ser calculada através da relação $\frac{\sigma_N}{\mu_N} = \frac{1}{\sqrt{\mu_N}}$, tornando válida a relação entre a acurácia relativa e o número de fótons medidos, dada por $\hat{\sigma}_N = \sqrt{N}$.

O efeito dos ruídos na formação de imagens digitais provém não apenas das incertezas, mas também por efeitos eletrônicos do próprio aparelho e de erros de arredondamento devido aos efeitos da quantização – podendo contribuir com uma incerteza da ordem de 0,28 ao ruído segundo Förstner e Wrobel (2016), independentemente da intensidade do pixel – entre outros.

A relação apresentada por $g = N_E + N_R + fN$ é válida para um sensor linear, em que g representa o valor de intensidade medido, N_E representa o ruído do sistema eletrônico, N_R o ruído surgido pelos efeitos de arredondamento, f o fator de escala e N o número de fótons responsável por aquela intensidade determinada, enquanto o valor da variância da intensidade calculado para câmeras de boa qualidade (em que a soma das variâncias devidas aos ruídos eletrônicos e de arredondamento) equivale a aproximadamente 1 pode ser calculado por $V(g) \approx 1 + f^2 g$, em que f representa o valor de escala e g o valor da intensidade medido.

Uma forma muito conveniente de se trabalhar com imagens digitais consiste em convertê-la exclusivamente para seus níveis de cinza, o que permite o trabalho com matrizes com um único canal de cor, tendo-se, então, uma matriz de trabalho de duas dimensões com valores no intervalo $[0,255]$, em que cada relação $g(i, j)$ para a linha i e a coluna j representa um pixel da imagem.

Em efeito, a existência de matrizes em um único canal permite a avaliação de suas distribuições; uma representação bastante útil, neste caso, é a utilização de histogramas em que a cada valor de intensidade de pixel g é computado um valor h indicando o número de vezes que aquele valor de intensidade está presente na imagem.

A manipulação das características apresentadas em um histograma permite a realização de transformações na imagem como, por exemplo o “ajuste de contraste”. Estas transformações serão mais detalhadas a seguir.

A figura 10 apresenta o histograma de níveis de cinza extraído utilizando a

biblioteca OpenCV 4.2.0 para a imagem apresentada na figura 8 após conversão para um único canal, em que o eixo horizontal representa a intensidade dos pixels e o eixo vertical, a frequência.

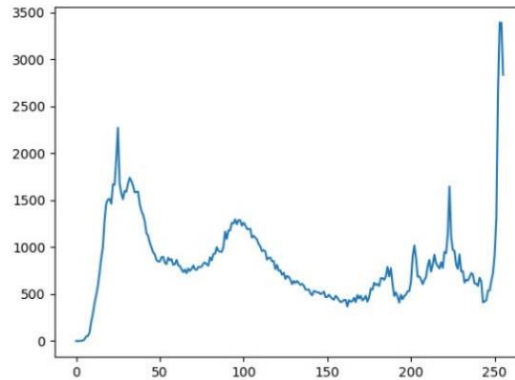


Figura 10 – Histograma de frequências de níveis de cinza.

Outra ferramenta importante no apoio à análise de imagens é o histograma cumulativo, cuja definição matemática é dada pela equação 7, em que $\underline{g}$ representa o pixel-âncora, isto é, o pixel único considerado em questão.

$$H(g) = \sum_{x=0}^g h(x) = n^{\circ} \text{ de pixels com } \underline{g}(i, j) \leq g \quad (7)$$

O trabalho com o histograma de frequências em conjunto com o gráfico de distribuição cumulativa possibilita o tratamento estatístico das imagens com grande eficiência algorítmica, trabalhando com tempos de computação na ordem de $O(N)$.

Valores estatísticos básicos como a média das intensidades, por exemplo, descrevem o brilho de uma imagem, enquanto a variância é um descritor do contraste, e sua manipulação é útil na utilização em transformações.

1.3.3.3. Transformações em imagens digitais

Imagens digitais podem ser transformadas a partir de três tipos diferentes de operadores: operadores globais, que atuam sobre a imagem como um todo, operadores locais, que atuam apenas em parte da imagem e operadores pontuais, cuja atuação se dá em um único pixel.

Operadores pontuais são os operadores mais simples e possuem a forma $b(i, j) = f(a(i, j), p)$ em que b representa a saída, f representa o próprio operador, a

representa a entrada e p um parâmetro utilizado.

Transformações lineares são um exemplo de operações realizadas ponto a ponto, porém, cujo resultado é percebido globalmente. Stachniss (2015) exemplifica através da aplicação do operador $f(a, p) = k + ma$ em uma imagem tal que $k > 0$ e $m = 1$.

Como resultado de sua aplicação, por exemplo, todas as intensidades da imagem são acrescidas de um valor maior do que 0 e sua média, por conseguinte, também terá seu valor aumentado – aumentando o brilho da imagem como um todo. Seguindo a mesma linha de raciocínio, mantendo um valor de $k = 0$ e $m > 1$ observa-se um aumento na variância da imagem, o que se traduz em um aumento do contraste.

Transformações não-lineares, por sua vez, também atuam como operadores pontuais com impacto em toda a imagem. Exemplo bastante prático é a binarização de imagens através da definição de limiares em uma função-degrau, como a apresentada na equação 8, em que b representa o pixel em questão, a a intensidade e T o limiar, utilizada para a separação entre o primeiro plano da imagem e o fundo, por exemplo, entendida como uma soma de duas gaussianas, separáveis por um limiar ótimo determinável.

$$b(a) = \begin{cases} b_0, & \text{se } a < T \\ \text{senão}, & b_1 \end{cases} \text{ para } (b_0, b_1) = (0, 255) \text{ ou } (255, 0) \quad (8)$$

Caso bastante importante de operadores em transformação de imagens digitais é o caso do operador monotônico, atuante no intervalo $[a, a + da]$ mapeando valores para o intervalo $[b, b + db]$ em que a relação $h_b(b)db = h_a(a)da$ é válida. A relação mostrada na equação 9, apresentada por Stachniss (2015) é um exemplo, em que $h_b(b)$ representa o valor mapeado para o ponto b , $h_a(a)$ a função de transformação no ponto a .

$$h_b(b) = \frac{h_a(a)}{\left| \frac{db}{da} \right|} = \frac{h_a(a)}{|f'(a)|} \quad (9)$$

Como a função deve ser dependente de b e não de a , considerando que $f^{-1}(b) = a$, ela pode ser reescrita para a forma mostrada na equação 10.

$$h_b(b) = \frac{h_a(f^{-1}(b))}{|f'(f^{-1}(b))|} \quad (10)$$

Förstner e Wrobel (2016) exemplificam o uso da equação 10 em uma operação

de transformação linear considerando $b = f(a) = k + ma$, sendo $f'(a) = m$ e $f^{-1}(b) = \frac{b-k}{m}$, calculado de forma global como $h_b(b) = \frac{h_a(b-k)}{|m|}$. Este tipo de transformação é utilizado, por exemplo, como operadores globais na equalização do histograma, ou mesmo na equalização da variância dos ruídos da imagem, permitindo uma análise estatística de imagens (em que todas apresentam o mesmo valor de variância, por exemplo).

Operadores locais, por sua vez, possuem aplicações extremamente importantes, como na redução de ruídos nas imagens, na aplicação de filtros de imagem, no cálculo de gradientes, entre outros.

Talvez o operador local mais conhecido seja o *box filter*, este capaz de trocar o valor de intensidade dos pixels individualmente pela média aritmética dos valores das intensidades dos pixels adjacentes, definido, para uma matriz em 2 dimensões, pela relação mostrada na equação 11, em que K e L representam as intensidades originais em cada pixel na horizontal e na vertical e k e l representam a quantidade de vizinhos considerados no cálculo.

$$g(i, j) = \frac{1}{KL} \sum_{k,l} f(1-k, j-l) \quad (11)$$

A equação 11 pode ser reescrita utilizando uma função-peso w , denominada *kernel*, de forma que a equação 12 passa a ser válida.

$$g(i, j) = \sum_{k,l} w(l, k) f(1-k, j-l) \quad (12)$$

Equações que trabalham com *kernels* possuem a capacidade de trabalharem com combinações de diferentes valores nas funções-peso, chamados de operadores de filtro lineares. Filtros no formato $g(i, j) = \sum_{k,l} f(1-k, j-l) w(l, k)$ com $g = f \cdot w$ são chamados de convoluções da função f com *kernel* w , e representam a aplicação de operadores sobre determinado conjunto ou função, com geração de um novo conjunto de trabalho.

Convoluções possuem algumas propriedades importantes, capazes de otimizar as eficiências algorítmicas: a propriedade comutativa – $a \cdot b = b \cdot a$ –, a propriedade associativa – $a \cdot b \cdot c = (a \cdot b) \cdot c = a \cdot (b \cdot c)$ –, a propriedade distributiva – $(a + b) \cdot c = a \cdot c + b \cdot c$ –, a multiplicação escalar – $\lambda(a \cdot b) = (\lambda a) \cdot b = a \cdot (\lambda b)$. Elas são capazes

de serem aplicadas sucessivamente, e seus resultados combinados; a isto se chamam convoluções múltiplas.

Kernels, por sua vez, são separáveis dimensionalmente – o que significa que podem ser aplicados separadamente em cada uma das dimensões, que o resultado é o mesmo.

Os *kernels* do operador local *box filter* escrito como uma convolução é representado na equação 13, em que R representa o *kernel*, o valor subscrito o número de pixels vizinhos considerados, e o valor sobrescrito a dimensionalidade da matriz.

$$R_3^{(1)} = \frac{1}{3} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \quad R_3^{(2)} = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \quad (13)$$

A variância de uma convolução é definida de acordo com o descrito na equação 14. Assim, para um valor de $R_m^{(2)}$ o valor de σ_{n_g} equivale a $\frac{1}{m} \sigma_{n_f}$, indicativo, por exemplo, de que a aplicação da convolução do *box filter* em uma imagem seja capaz de diminuir o ruído associado.

$$\sigma_{n_g}^2 = \sum_i (w(i))^2 \sigma_{n_f}^2 \quad (14)$$

Um segundo exemplo de operador local utilizado com o fim de suavizar imagens digitais através da utilização de um *kernel* específico é o Filtro Binomial (também chamado de Filtro Gaussiano). Seu funcionamento se baseia na diminuição os pesos relativos das intensidades individuais da imagem, aproximando-os de uma curva gaussiana, de forma $G_{(x,y)} = \frac{1}{2\pi\sigma^2} e^{\frac{-(x+y^2)}{2\sigma^2}}$, em que x e y representam as dimensões verticais e σ o desvio-padrão desejado a ser utilizado para a distribuição Gaussiana.

Para uma matriz em duas dimensões com $\sigma \cong 0,85$, o *kernel* utilizado é o representado na equação 15; seu efeito imediato na imagem é o surgimento de um borrão, tal como exemplificado na figura 11.

$$R_2^{(3)} = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \quad (15)$$



Figura 11 – Comparação do efeito da convolução de um Filtro Binomial

Fonte: <<https://appdividend.com/2020/09/19/python-cv2-filtering-image-using-gaussianblur-method/>>. Acesso em 10/01/2021.

De forma similar ao *Box Filter*, o Filtro Binomial é capaz de diminuir o ruído da imagem, porém, em menor magnitude; apesar disto, ele normalmente é preferível no trabalho com imagens por preservar mais a estrutura das arestas, utilizadas em processamentos subsequentes.

O cálculo da derivada primeira de um pixel em relação a seus vizinhos é de grande importância para a identificação de características da imagem como, por exemplo, a identificação de contornos – isto graças à diferença brusca de intensidade em pixels vizinhos. Isto é conseguido, por exemplo, pela aplicação do Filtro Gradiente, operador local definido pela aproximação da primeira derivada das funções representativas dos valores de intensidade, usado para a obtenção de propriedades das imagens em pontos isolados.

Para uma matriz unidimensional, a primeira derivada entre dois pontos adjacentes pode ser calculada por $f'(i) \approx \frac{\Delta f}{\Delta x} = \frac{f(i+1) - f(i)}{\Delta x}$, valendo a relação $\Delta f(i) = f(i+1) - f(i)$. Como Δf representa uma operação de subtração, ele pode ser escrito na forma de um operador $\Delta = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$ e utilizado como uma convolução.

Para um espaço bidimensional o operador gradiente ∇ consiste das derivadas parciais $\nabla = \begin{bmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{bmatrix} \approx \begin{bmatrix} \frac{\Delta x}{\Delta X} \\ \frac{\Delta y}{\Delta Y} \end{bmatrix}$, sendo o gradiente da função calculado tanto para a x como para y tal que $\nabla g = \nabla \cdot g = \begin{bmatrix} g_x \\ g_y \end{bmatrix}$, com a magnitude do gradiente definida pela relação apresentada na equação 16 e a sua direção pela relação da equação 17.

$$|\nabla g| = \sqrt{g_x^2 + g_y^2} \quad (16)$$

$$\alpha = \arctan\left(\frac{g_y}{g_x}\right) = \text{atan2}(g_y, g_x) \quad (17)$$

O cálculo dos gradientes pode ser aproximado pela utilização de determinados operadores, em convoluções, tais como o Operador Sobel – em uma janela 3x3 – que combina os resultados da aplicação de um Filtro Gaussiano com o Filtro Gradiente a

partir dos operadores $\Delta x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$ e $\Delta y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$.

O Operador Scharr é uma modificação do Operador Sobel que busca minimizar o erro angular médio ponderado quadrático no domínio de Fourier sob a condição de que os filtros resultantes sejam numericamente consistentes. Ele também é usado no

cálculo das derivadas primeiras e assume as formas $\Delta x = \begin{bmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{bmatrix}$ e $\Delta y =$

$$\begin{bmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix}.$$

As derivadas segundas também são bastante utilizadas no processamento de imagens e, para um espaço unidimensional em uma direção, são definidas pela relação apresentada na equação 18 enquanto para um espaço bidimensional ela é calculada pela matriz Hessiana de f , apresentada na equação 19.

$$\frac{\partial^2 f}{\partial x^2} = \frac{\partial^2}{\partial x^2} \cdot f = \left(\frac{\partial}{\partial x} \cdot \frac{\partial}{\partial x} \right) \cdot f, \text{ sendo } \frac{\partial^2}{\partial x^2} = \begin{bmatrix} 1 \\ -2 \\ 1 \end{bmatrix} \quad (18)$$

$$H(f) = [h(f)_{ij}] = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial y \partial x} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} \quad (19)$$

O cálculo da derivada segunda entre os pontos é bastante útil em tarefas de identificação de arestas, reconhecidas pelas mudanças abruptas e contínuas entre intensidades pixels escuros e claros (e vice-versa), quando apenas a derivada primeira mostra-se insuficiente. Seu cálculo, por sua vez, pode ser aproximado através do Operador de Laplace, definido por $\Delta L = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, para o qual é válida a relação $\Delta L f(x, y) = f_{xx}(x, y) + f_{yy}(x, y) = 0$ e cujo cálculo pode ser feito pelo

operador bidimensional $R_2^{(3)} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$.

1.4 Justificativa

A fotogrametria é uma área de estudo que se utiliza da luz através de dispositivos, tais como câmeras de vídeo e/ou de fotografia, para a realização de medições (tais como determinação de localizações, reconstruções 3D, associação de dados e detecção de objetos) e para a estimação de propriedades geométricas e semânticas com base em imagens ou observações por meio de um ou mais dispositivos similares (STACHNISS, 2020).

Apesar de não ser uma técnica nova – de acordo com Albertz (2001) o termo foi utilizado academicamente pela primeira vez em 1867 por Albrecht Meydenbauer – a fotogrametria tem ganhado especial relevância não somente pela popularização do uso de câmeras, por exemplo as disponíveis em smartphones, como também graças às recentes evoluções em áreas de visão computacional, aprendizagem de máquina e a robótica.

Entre suas principais vantagens de aplicação estão a possibilidade de interações sem contato – especialmente interessante para áreas visíveis, porém inacessíveis, ou materiais que possam causar ou sofrer algum dano decorrente da necessidade de inspeção – e sua capacidade de cobrir grande número de áreas e realizar medições com a utilização de poucos equipamentos e em pouco tempo, quase instantaneamente e de forma automática; métodos como a estereofotogrametria, por sua vez, auxiliam na representação de imagens em 3D tendo como base imagens em apenas duas dimensões. Tudo isto se traduz em vantagens econômicas frente a outras técnicas disponíveis, que podem ser aproveitadas pela indústria, por exemplo, para o controle de qualidade de seus processos.

Aplicações deste assunto são variadas e cobrem uma grande gama de aplicações, que vão da geração de mapas e modelos topográficos ao monitoramento ambiental, preservação de patrimônios culturais e sítios arqueológicos, utilização no desenvolvimento de carros autônomos, entre outros.

Publicações acadêmicas recentes sobre a fotogrametria são bastante variadas. Trabalhos como o de Wu et al. (2021) demonstraram a utilização da fotogrametria na modelagem topográfica de crateras e rochas lunares na missão chinesa Chang'E-4 com precisão na ordem dos centímetros, enquanto Bregar et al. (2021) discutiram sobre o uso de funções de frequência de retorno no uso de câmeras de alta velocidade. Wang et al. (2021) utilizaram sistemas robóticos de

estereofotogrametria e técnicas de nuvens de pontos para a realização de inspeções em superfícies curvas – especificamente aqui hélices de turbinas eólicas; seguindo a mesma linha de pesquisa, Narvydas e Puodziuniene (2020) também trabalharam com o levantamento de propriedades geométricas superfícies não-lineares, porém, a partir do trabalho com imagens binárias (imagens compostas unicamente pelas cores preta e branca).

No campo do monitoramento ambiental, Abdollahnejad et al. (2020) utilizaram fotografias aéreas para realizar a classificação de espécies de árvores em uma área determinada. Zollini et al. (2020) trabalharam com imagens aéreas, porém, para a realização de inspeções em pontes de concreto na Itália, que possui – segundo os autores – cerca de 60 mil pontes consideradas patrimônios históricos que precisam ser constantemente monitoradas para deterioração visual e para acompanhamento de medições periódicas. Menna et al. (2020), por sua vez, utilizaram imagens de sistemas aéreos não-tripulados para a realização de inspeções metrológicas em 3D.

Na área da topologia, Tan et al. (2021) trabalharam com o controle de variações longitudinais na morfologia de canais no norte da China, enquanto Tarolli et al. (2020) se dedicaram ao monitoramento de instabilidades em declives através do sensoriamento remoto. Guidi et al. (2020) demonstraram o uso da fotogrametria em operações de deslocamento lateral; já a aplicação de imagens em inspeções em estruturas de forma não-destrutiva foi descrita por El Masri (2020), enquanto Xu et al. (2020) relataram resultados no monitoramento fotogramétrico da estrutura de túneis, com resolução de 0,01 mm. Odumosu et al. (2020), por sua vez, utilizaram o algoritmo de filtro de Kalman para o estudo de monitoramento de deformação de prédios.

Não menos relevante, Harrison et al. (2021) aplicaram a fotogrametria à medicina, para a modelagem em 3 dimensões em deformidades na face. Ozturk e Rashidzade (2020) utilizaram a modelagem 3D em cálculos envolvendo agregados grossos no concreto, já Yu e Lubineau (2021) relataram a realização de cálculos de deslocamentos utilizando a câmera de smartphones.

Melhorias às técnicas de fotogrametria também vêm sendo investigadas, como demonstradas nos trabalhos de Duran e Atik (2021) sobre transformações lineares, de Mata-Falcon et al. (2020) sobre o uso combinado de medições de imagens digitais com a utilização conjunta de sistemas de fibras ópticas, e os avanços relatados por Liu e Su (2020) no trabalho com segmentação semântica.

Fundamental para a justificativa deste trabalho, a publicação de Catalucci et

al. (2020) apresentou o uso da fotogrametria em sistemas de medição de peças manufaturadas com formas complexas, tais como as utilizadas por sistemas automotivos.

O uso de lógicas não-clássicas no monitoramento de qualidade em um CEP com justificativa no levantamento de características do produto ambíguas ou com certa incerteza (inclusive linguística) foi explorado no trabalho de Boroujeni et al. (2021) a partir da aplicação de modelos baseados na lógica Fuzzy. Já Ma et al. (2021) discorreram sobre a eficácia e a eficiência do controle dimensional, especificamente no cenário de aplicações automotivas, a partir da utilização de recursos computacionais.

Apesar de a aplicação da LP aos processos de CEP já ter sido explorada, por exemplo, pelos já citados trabalhos de Cruz (2015), Damin e Da Silva Filho (2012) e, mais recentemente por Rocha (2019) e Gonçalves (2019), a literatura acadêmica é carente de publicações sobre sua utilização em conjunto com a fotogrametria, área esta que, como demonstrado, vem sendo bastante utilizada no meio acadêmico. Identifica-se, aqui, uma oportunidade de desenvolvimento científico importante que permitirá sua continuidade em aplicações reais, com impactos positivos para a sociedade.

1.5 Objetivo

1.5.1. Objetivo geral

O objetivo principal deste trabalho é o de demonstrar a utilização de uma RNAP no processamento de imagens, capaz de extrair atributos e auxiliar na tomada de decisões, com base em técnicas empregadas na fotogrametria.

1.5.2. Objetivos específicos

Como objetivos secundários, este trabalho busca:

- a) Apresentar a criação da Unidade Paraconsistente de Convolução (UPC) – formada pela conformação de diferentes Células Neurais Artificiais Paraconsistentes (CNAPs), capaz de executar operações de convolução em imagens selecionadas a partir de filtros pré-definidos;
- b) Apresentar o processo de identificação e extração de atributos de interesse das

- imagens processadas pela UPC;
- c) Comparar os resultados das operações de convolução realizadas pela UPC com as operações correlatas realizadas pelo OpenCV – biblioteca de ampla utilização em projetos de visão computacional, sob licença *Berkeley Software Distribution* (BSD);
 - d) Demonstrar a utilização de processo de classificação de objetos a partir dos atributos extraídos das imagens processadas previamente;
 - e) Demonstrar o uso da LP na identificação de desvios em dimensões de objetos cujas imagens são capturadas e processadas pela UPC.

2 MATERIAIS E MÉTODOS

A seguir são descritas as etapas realizadas no desenvolvimento deste projeto de pesquisa, que incluíram, mas não se limitaram, ao desenvolvimento da UPC, à extração das propriedades das imagens analisadas, ao processo de criação e configuração das RNAPs utilizadas para a realização da classificação e sua execução.

2.1 Materiais

2.1.1. Captura de imagens e preparo do ambiente de desenvolvimento

As fotografias utilizadas neste trabalho foram tiradas com um *smartphone* Apple iPhone 8 plus com câmera dupla com resolução de 12 MP, sensor *Apple iSight Duo Camera*(8) tipo CMOS BSI com abertura $f/1.8$, tamanho $1/3$, com tamanho do *pixel* igual a $1.20\ \mu\text{m}$ e *flash* Quad-LED, conforme especificações dadas pelo fabricante. As imagens foram armazenadas em formato comprimido JPG e transferidas para o computador.

Esta pesquisa trabalhou com dois objetos distintos:

- a) Uma peça quadrada, cor salmão, com superfície de dimensões 105 mm x 105 mm e altura desprezível;
- b) Uma peça retangular, cor rosa, com superfície de dimensões 135 mm x 55 mm e altura desprezível.

Todas as dimensões informadas foram medidas com régua de alumínio. No momento da captura das imagens o *flash* ficou definido na opção “automático” e todos os objetos foram dispostos, individualmente, sobre um tecido verde – de forma a facilitar a separação do plano de fundo – e posicionados em cima de uma folha de papel branco tamanho A4, com dimensões nominais aferidas de 210 mm x 297 mm.

A câmera foi posicionada em diferentes posições e sob diferentes perspectivas dos objetos sem qualquer tipo de controle de posição, distância ou luminosidade, garantindo apenas que os objetos estivessem presentes na fotografia sem cortes ou obstáculos. Foram retiradas 25 fotografias de cada um dos objetos, perfazendo um total de 50 imagens. A figuras 12 mostra exemplos de fotografias utilizadas para a peça quadrada e para a peça retangular.

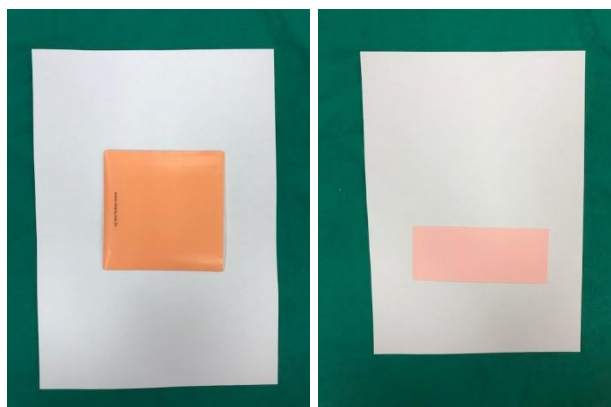


Figura 12 – Exemplos de fotografias utilizando a peça quadrada e a peça retangular.

Os *scripts* para manipulação e coleta das propriedades das imagens assim como para a realização dos estudos estatísticos e execução das RNAPs foram desenvolvidos utilizando linguagem Python 3.8.5, considerando o padrão de Filtro de Bayer, utilizando as bibliotecas OpenCV 4.2.0 e NumPy 1.18.5. Todos foram executados em sistema operacional Linux na distribuição Ubuntu 20.04.1 LTS instalado em computador de uso pessoal com processador Intel Core i5-8265U CPU @ 1.600GHz 1.80 GHz e memória RAM de 8,00 GB.

As fotografias salvas originalmente foram armazenadas em pasta segregada, utilizada como repositório no caso de necessidade de reprocessamento, e foram criadas cópias, armazenadas em local separado. De forma a preservar as escalas da imagem em relação à folha A4 de tamanho conhecido, as fotografias não foram redimensionadas em seu armazenamento.

2.2 Método

2.2.1. Planejamento

A figura 13 ilustra, na forma de um fluxograma, os passos a serem seguidos durante o processamento das imagens, divididos em etapas pré-definidas que vão da preparação dos objetos e captura das fotografias utilizadas, estas descritas anteriormente, até as etapas de classificação e identificação de desvios. De forma didática, as etapas que utilizam a LP como base de seu funcionamento foram agrupadas na área delimitada pela borda na cor vermelha.

O processamento das imagens por meio das RNAPs é feito na chamada

Unidade Paraconsistente de Convolução (UPC), responsável por aplicar as operações de transformação em imagem. Tanto as etapas de classificação como a detecção dos desvios também são realizadas por meio da LP, através de CNAPs especializadas para estas funções.

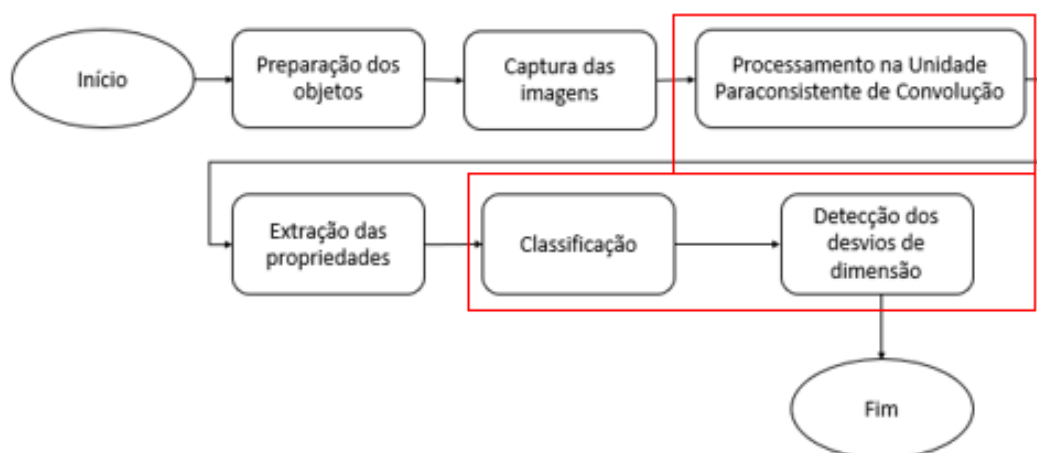


Figura 13 – Fluxograma ilustrativo para o planejamento do processamento das imagens

2.2.2. Desenvolvimento da Unidade Paraconsistente de Convolução (UPC)

O desenvolvimento da UPC foi inspirado pelo Controlador Paraconsistente Estatístico de Processos – CEP-LPA – descrito por Cruz (2015), cujo uso no cálculo da média aritmética de valores simulados por padrões gerados via software demonstrou ser eficaz.

A UPC é composta por módulos extratores de evidência iniciais, com funções de preparar os dados de acordo com os *kernels* dos filtros aplicados – expressando-os como graus de evidência –, módulos extratores de evidência finais, utilizados após cada iteração nos processos de convolução de forma a equalizar os valores processados sem que haja perda de precisão, e do módulo de convolução, utilizado na realização das operações aritméticas do processo de convolução. Ela conta também com módulos auxiliares, não envolvidos na convolução diretamente, mas utilizados por ela – como é o caso do módulo de binarização (MoBi).

Um fluxograma contendo os passos para o processamento das imagens na UPC é mostrado na figura 14 e seus módulos são descritos a seguir.

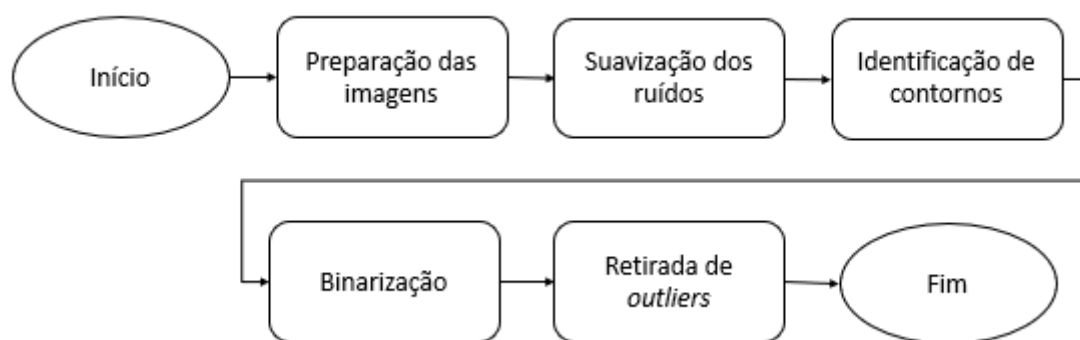


Figura 14 – Fluxograma de processamento na UPC.

Com exceção da etapa de preparação das imagens, cada um dos blocos apresentados no fluxograma da figura 14 representa uma etapa de processamento das imagens por meio da LP, através de diferentes mecanismos.

A etapa de suavização dos ruídos é realizada através da utilização de um filtro baseado no filtro gaussiano, cujo *kernel* foi desenvolvido neste trabalho, com a função de diminuir a variância global da imagem, tal como o filtro original. Após sua aplicação é esperada a obtenção de um histograma de intensidades mais suavizado em relação ao histograma obtido da imagem sem processamento. Sua aplicação é feita via um módulo, denominado “Módulo Extrator de Evidência Binomial inicial”, descrito no item 2.2.5.

Da mesma forma, a etapa de identificação de contornos também utiliza um filtro desenvolvido neste trabalho, com o fim de obter os gradientes entre pixels de regiões vizinhas e, assim, ser capaz de identificar a existência de contornos – representados por um grande contraste entre as duas regiões. Para o processamento paraconsistente, desenvolveu-se um módulo denominado “Módulo Extrator de Evidência Nablaiana inicial” e descrito no item 2.2.6.

Após aplicação dos filtros de ambos os módulos descritos, os resultados são normalizados e corrigidos pela aplicação do “Módulo Extrator de Evidência Intermediária” e do “Módulo Extrator de Evidência final”, ambos descritos no item 2.2.7 e com função de ajustar os valores de evidência encontrados, preparando-os para prosseguirem no fluxo de processamento.

Por fim, tanto as etapas de binarização como de retirada de *outliers* são executadas pelo “Módulo de Binarização”, descrito no item 2.2.8, que utiliza a LP para executar uma função-degrau, deixando que estejam presentes na imagem apenas pixels de cor preta ou branca, que permitem, entre outras coisas, a retirada de

imagens de fundo e também a retirada de *outliers*, isto é, exceções encontradas na imagem que devem ser eliminadas visto que, caso permanecessem aí, poderiam prejudicar a qualidade do seu processamento.

A aplicação de todos os filtros destes módulos foi conseguida através de sua utilização no chamado “Módulo de Convolução”, introduzido no item 2.2.4, responsável por permitir a execução da LP frente aos filtros desenvolvidos.

2.2.3. Fluxo de processamento de imagens na UPC

As cópias das imagens, armazenadas em pastas segregadas, seguiram o fluxo de processamento via código, que incluiu suas conversões para a escala de cinza, considerando o código de conversão de cores `COLOR_BGR2GRAY`, do OpenCV. As imagens não foram redimensionadas de forma a preservar suas escalas originais.

Como estratégia geral, o primeiro objetivo a ser atingido é o de se encontrar a folha de papel A4 na imagem – sendo ele “o maior contorno” presente na fotografia – e considerar apenas sua área no processamento das propriedades. Uma vez que o tamanho da folha de papel é conhecido, isto possibilita ao algoritmo calcular as dimensões dos demais objetos, cujos valores são proporcionais aos da folha A4.

O primeiro passo do processamento na UPC foi a aplicação do filtro binomial, combinando a utilização tanto do MEEBi como da UC, o que proveu uma imagem suavizada, com menor magnitude de ruído. Uma vez que o Filtro Gaussiano foi aplicado, isto permitiu a utilização do MEENi juntamente com a UC, em um processo subsequente, capaz de identificar os contornos presentes na imagem através das diferenças abruptas de intensidades encontradas nos *pixels* vizinhos da fotografia.

A imagem resultante foi binarizada pela aplicação do MoBi; ações de dilatação e erosão disponíveis no OpenCV foram aplicadas, então, de forma a remover os chamados *outliers*, isto é, *pixels* próximos às extremidades com intensidades muito altas ou muito baixas presentes em regiões em que não seriam esperados (uma vez que seu entorno seja, justamente, o contrário).

As informações sobre os contornos existentes nas imagens foram extraídas utilizando métodos próprios da linguagem Python e/ou do OpenCV definidos como “uma curva que une todos os pontos contínuos (ao longo da fronteira), tendo a mesma cor ou intensidade”. Neste caso, o contorno de interesse é o mais externo, portanto

as propriedades foram retiradas utilizando a hierarquia RETR_EXTERNAL, sob modo de aproximação CHAN_APPROX_SIMPLE, que armazena apenas os pontos das 4 coordenadas das arestas em memória, o suficiente para reconstruir, virtualmente, os objetos.

As arestas foram reconstruídas via código e as informações sobre a área e perímetro do maior polígono foram encontradas através dos métodos `contourArea()` e `arcLength()`, respectivamente.

Tendo o maior contorno sido identificado, representando a folha A4, a imagem foi ajustada via código de forma a mudar sua perspectiva, ignorando a existência do conteúdo externo (o plano de fundo, verde) e considerando apenas a área do papel como válida para a imagem. Uma vez que as dimensões da folha se tornaram conhecidas, os dados de medições dos polígonos internos puderam ser determinados proporcionalmente, realizando novamente as operações de determinação de perímetros e áreas dos polígonos internos – agora, os únicos presentes na imagem.

2.2.4. Módulo de Convolução – MC

O MC é o módulo desenvolvido com a função de executar a operação de convolução nas imagens, definida na equação 20, em que V representa o valor do *pixel*-âncora, f_{ij} o coeficiente do kernel de convolução na posição (i, j) , d_{ij} o dado encontrado no *pixel* correspondente a f_{ij} , q representa a dimensão do *kernel* – neste trabalho ele tem um valor $q = 3$ por se tratar de um *kernel* definido por uma matriz 3x3 –, e F a soma dos coeficientes do *kernel*.

$$V = \left\lfloor \frac{\sum_{i=1}^q (\sum_{j=1}^q f_{ij} d_{ij})}{F} \right\rfloor \quad (20)$$

Uma vez que os valores do *kernel* estejam disponíveis, por terem sido calculados pelos módulos extratores através de $f_{ij} d_{ij}$, e que o valor de F também foi utilizado durante as extrações, os valores de intensidade para cada âncora podem ser calculados individualmente através das somatórias em (i, j) .

Sua construção foi baseada na Unidade Neural Artificial Paraconsistente de Extração de Média Móvel (UNAPEmm) descrita no trabalho de Cruz (2015). Sua base é uma CNAPap, capaz de aprender valores reais no intervalo fechado $[0,1]$, com $F_A =$

1, retornando um grau de evidência resultante na saída – $\mu_{E(i)}$ – calculado conforme equação 21, em que μ_i representa o grau de evidência favorável apresentado na entrada e $\mu_{E(i-1)}$ o grau de evidência resultante anterior.

$$\mu_{E(i)} = \frac{\mu_i - (1 - \mu_{E(i-1)}) + 1}{2} \quad (21)$$

Importante notar que o valor de 20 amostras utilizado pelo UNAPEmm, definido de forma a minimizar o erro, foi mantido no MC. Empiricamente outros valores foram utilizados, porém, os resultados mostraram-se inferiores aos encontrados utilizando o mesmo número de iterações que Cruz (2015).

Tal como na UNAPEmm, optou-se pela utilização de 6 CNAPap ligadas em série, porém, entre elas foi incluída uma instância do MEEi, isto de forma a resgatar o valor às somatórias, em detrimento do valor de média, calculado pela UNAPEmm. Ao final, a evidência calculada é convertida novamente ao valor de intensidade por meio do MEEf. Um esquema da UPC é mostrado na figura 15.

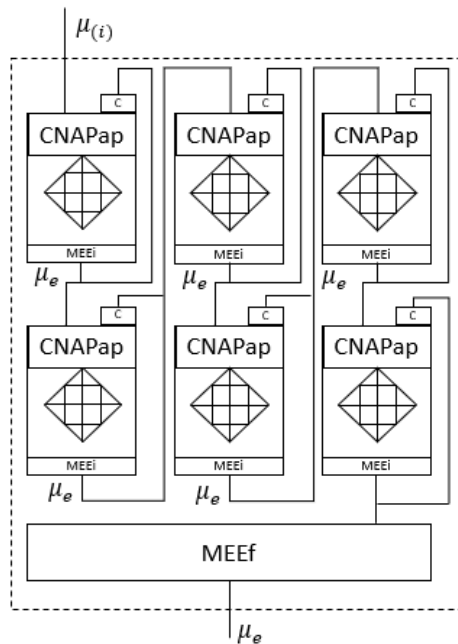


Figura 15 – Esquema do funcionamento da UC com 6 CNAPap ligadas em série.

2.2.5. Suavização de ruídos: Módulo Extrator de Evidência Binomial inicial – MEEBi

O MEEBi foi desenvolvido considerando um *kernel* de dimensões 3x3, em que

o elemento central, chamado âncora, se alinha com o *pixel* de interesse em uma imagem. A partir do *kernel* gaussiano, mostrado na equação 15, e considerando todas as combinações de linha e coluna possíveis em uma matriz 3x3, o grau de evidência $\mu_{i,j}$, em que i representa a posição da linha do *pixel* no kernel K de dimensões $m \times n$ em relação ao elemento-âncora e j a posição da coluna do *pixel* no mesmo kernel K de dimensões $m \times n$ em relação ao elemento-âncora, pode ser calculado a partir das relações mostradas nas equações 22 a 30.

As equações apresentadas foram baseadas no *kernel* gaussiano, adaptadas para serem utilizadas no contexto paraconsistente, considerando a situação-problema apresentada na dissertação podendo, eventualmente, precisarem ser adaptadas de acordo com diferentes objetivos.

$$\mu_{(-1,-1)} = \begin{cases} 0,0625 \cdot \frac{K[i-1][j-1]}{255}, & \text{se } i! = 0 \text{ e } j! = 0 \\ \text{senão}, & 0 \end{cases} \quad (22)$$

$$\mu_{(-1,0)} = \begin{cases} 0,125 \cdot \frac{K[i-1][j]}{255}, & \text{se } i! = 0 \\ \text{senão}, & 0 \end{cases} \quad (23)$$

$$\mu_{(-1,1)} = \begin{cases} 0,0625 \cdot \frac{K[i-1][j+1]}{255}, & \text{se } i! = 0 \text{ e } j! = n - 1 \\ \text{senão}, & 0 \end{cases} \quad (24)$$

$$\mu_{(0,-1)} = \begin{cases} 0,125 \cdot \frac{K[i][j-1]}{255}, & \text{se } j! = 0 \\ \text{senão}, & 0 \end{cases} \quad (25)$$

$$\mu_{(0,0)} = 0,25 \cdot \frac{K[i][j]}{255} \quad (26)$$

$$\mu_{(0,1)} = \begin{cases} 0,125 \cdot \frac{K[i][j+1]}{255}, & \text{se } j! = n - 1 \\ \text{senão}, & 0 \end{cases} \quad (27)$$

$$\mu_{(1,-1)} = \begin{cases} 0,0625 \cdot \frac{K[i+1][j-1]}{255}, & \text{se } i! = m - 1 \text{ e } j! = 0 \\ \text{senão}, & 0 \end{cases} \quad (28)$$

$$\mu_{(1,0)} = \begin{cases} 0,125 \cdot \frac{K[i+1][j]}{255}, & \text{se } i! = m - 1 \\ \text{senão}, & 0 \end{cases} \quad (29)$$

$$\mu_{(1,1)} = \begin{cases} 0,0625 \cdot \frac{K[i+1][j+1]}{255}, & \text{se } i! = m - 1 \text{ e } j! = n - 1 \\ \text{senão}, 0 \end{cases} \quad (30)$$

2.2.6. Identificação de Contornos: Módulo Extrator de Evidência Nablaiana inicial – MEENi

A “evidência nablaiana” recebeu este nome devido ao símbolo da nabla – ∇ – ser utilizado para representar o gradiente.

O cálculo do Filtro Gradiente, na UPC, é realizado utilizando uma modificação do Operador Scharr; isto, porque a LP define que os graus de evidência utilizados estejam dentro do intervalo $[0,1]$, condição não satisfeita na completude pelo operador em questão devido a existência de pesos negativos aí, que não seriam eliminados por uma normalização simples.

O MEENi foi concebido utilizando um *kernel* de dimensões 3×3 , em que o elemento central, chamado âncora, se alinha com o *pixel* de interesse em uma imagem.

A partir do Operador Scharr e, considerando todas as combinações de linha e coluna possíveis em uma matriz 3×3 , o grau de evidência $\mu_{i,j}$, em que i representa a posição da linha do *pixel* no *kernel* K de dimensões $m \times n$ em relação ao elemento-âncora e j a posição da coluna do *pixel* no mesmo *kernel* K de dimensões $m \times n$ em relação ao elemento-âncora, as evidências são calculadas a partir das relações mostradas nas equações 31 a 39, já referentes ao cálculo nas dimensões x e y .

Tal como exposto previamente à apresentação das equações desenvolvidas para a MEENi, torna-se mister reforçar que as equações apresentadas a seguir foram baseadas em *kernel* existente e que foram adaptadas para serem utilizadas no contexto paraconsistente, sendo, esta, uma importante etapa do desenvolvimento deste trabalho, e cujo uso não pode ser generalizado para situações e aplicações diversas.

$$\mu_{(-1,-1)} = \begin{cases} 0,21875 \cdot \frac{K[i-1][j-1]}{255}, & \text{se } i! = 0 \text{ e } j! = 0 \\ \text{senão}, 0 \end{cases} \quad (31)$$

$$\mu_{(-1,0)} = \begin{cases} 0,31250 \cdot \frac{K[i-1][j]}{255}, & \text{se } i! = 0 \\ \text{senão}, & 0 \end{cases} \quad (32)$$

$$\mu_{(-1,1)} = \begin{cases} 0,40625 \cdot \frac{K[i-1][j+1]}{255}, & \text{se } i! = 0 \text{ e } j! = n - 1 \\ \text{senão}, & 0 \end{cases} \quad (33)$$

$$\mu_{(0,-1)} = 0 \quad (34)$$

$$\mu_{(0,0)} = \begin{cases} 0,31250 \cdot \frac{K[i][j]}{255} \end{cases} \quad (35)$$

$$\mu_{(0,1)} = \begin{cases} 0,62500 \cdot \frac{K[i][j+1]}{255}, & \text{se } j! = n - 1 \\ \text{senão}, & 0 \end{cases} \quad (36)$$

$$\mu_{(1,-1)} = \begin{cases} 0,21875 \cdot \frac{K[i+1][j-1]}{255}, & \text{se } i! = m - 1 \text{ e } j! = 0 \\ \text{senão}, & 0 \end{cases} \quad (37)$$

$$\mu_{(1,0)} = \begin{cases} 0,31250 \cdot \frac{K[i+1][j]}{255}, & \text{se } i! = m - 1 \\ \text{senão}, & 0 \end{cases} \quad (38)$$

$$\mu_{(1,1)} = \begin{cases} 0,40625 \cdot \frac{K[i+1][j+1]}{255}, & \text{se } i! = m - 1 \text{ e } j! = n - 1 \\ \text{senão}, & 0 \end{cases} \quad (39)$$

A imagem processada no MEENi ainda passa por processos sucessivos de binarização, de forma que apenas as arestas sejam mantidas na matriz representante.

A primeira binarização, utilizando o MoBi considera um valores-limiare empíricos e ajustados conforme necessidade. A imagem binarizada é limpa através de processos de erosão e dilatação, próprios da biblioteca OpenCV.

2.2.7. Módulo Extrator de Evidência Intermediária (MEEi) e Módulo Extrator de Evidência final (MEEf)

Os módulos extratores de evidência intermediária (MEEi) e final (MEEf) são módulos utilizados no ajuste dos graus de evidência durante seus cálculos no módulo de convolução, de forma a garantir não apenas sua precisão como, ainda, para

realizar o ajuste de intensidade dos *pixels*, diminuídos nos módulos extratores de evidência.

O MEEi define uma função linear $f(\mu)$ tal que $f(\mu) = 2\mu$, enquanto o MEEf define uma função linear $f(\mu)$ tal que $f(\mu) = 255\mu$.

2.2.8. Extração de Camadas e retirada de *outliers*: Módulo de Binarização – MoBi

O MoBi utiliza uma função-degrau com o fim de binarizar a imagem – isto é, permitir apenas valores 0 ou 255 em seus *pixels*. A imagem binária é muito útil para a extração de camadas – como separação de planos –, definição de bordas e continuidades, definição de componentes conectados, identificação e retirada de *pixels outliers* por transformações de erosão e dilatação e mesmo a extração de propriedades topológicas, como número de componentes, e geométricas, como áreas, perímetros, momentos e centros de massa, por exemplo.

Para o MoBi foi definido um limiar de intensidade de *pixel* empírico, para o qual é válida a relação apresentada na equação 40, em que $g(i, j)$ representa o valor de cinza na posição (i, j) em que i representa a linha e j a coluna e T representa o valor-limiar utilizado como referência no degrau.

$$g(i, j) = \begin{cases} 0, & \text{se } g(i, j) < T \\ 255, & \text{senão} \end{cases} \quad (40)$$

2.2.9. CNAP de classificação

Um problema de classificação típico é definido para um conjunto Ω composto por um número K de classes c tal que $\Omega = \{\omega_1 \dots \omega_K\}$. Nele, assume-se a existência de uma característica e para a qual existe uma função f capaz de determinar a que classe determinado dado pertence, tal que $c = f(e)$ com $c \in \Omega$.

Neste trabalho foram utilizados dois objetos – um quadrado e outro retangular – que compõem o universo de classes, sobre os quais, a partir das propriedades extraídas anteriormente deve ser realizada a classificação.

A CNAPp utilizada foi construída considerando um fator de aprendizagem $F_A = 1$, fator de tolerância à contradição $Ft_{ct} = 1$, fator de tolerância à certeza $Ft_c =$

1, fator de tolerância à decisão $Ft_D = 1$ e fator de aprendizagem $F_A = 1$.

Ela recebeu como graus de evidência μ os valores de área e de perímetro das imagens, normalizados de acordo com as equações 41 e 42, de forma a mantê-los no intervalo $[0,1]$, definido pela LPA2v, em que V representa o valor normalizado, A representa o valor da área, P o valor do perímetro e L o maior lado encontrado para os polígonos.

$$V = \frac{\sqrt{A}}{L} \quad (41)$$

$$V = 0,25 \cdot \frac{P}{L} \quad (42)$$

A CNAPP é esquematizada na figura 16. Os graus de evidência normalizados são processados através da execução do algoritmo da célula neural artificial paraconsistente padrão (descrito no Apêndice A), definindo uma saída resultante, μ_r , também é normalizada, responsável pela classificação geométrica da figura.

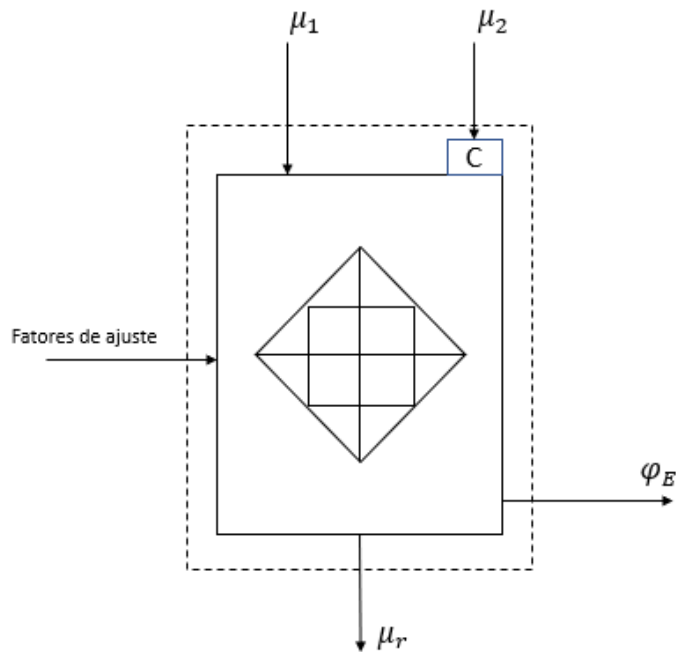


Figura 16 – Esquema de funcionamento da CNAPP

Fonte: Adaptado de DA SILVA FILHO et al. (2008).

O resultado calculado pela μ_r determina a classificação dos objetos, sendo um valor de $\mu_r \geq 0,71$ classificado como um “quadrado” e um valor de $\mu_r < 0,71$ classificado como “retângulo”.

2.2.10. CNAP de detecção de desvios

O módulo de detecção do desvio da média, última etapa do processamento das imagens, tem a função de, a partir da classificação dada pela CNAPp, identificar se as dimensões obtidas da imagem para o objeto em questão estão de acordo com o esperado – o que, em uma linha de produção, por exemplo, tem importante função na identificação de desvios de processo e/ou de materiais com defeitos, podendo serem removidos de uma esteira.

A detecção de valores fora dos parâmetros de controle foi conseguida através da utilização de uma Célula Neural Artificial Paraconsistente de detecção de igualdade (CNAPdi), em arranjo baseado no Módulo de Detecção utilizado por Cruz (2015).

A CNAPdi tem como principal função a comparação de dois valores de graus de evidência aplicados na entrada e gera, como resposta, um valor relativo à igualdade no intervalo $[0,1]$ (DA SILVA FILHO et al., 2008).

O funcionamento de uma CNAPdi se baseia na execução do algoritmo da célula neural artificial paraconsistente de detecção de igualdade (descrito no Apêndice B) e é realizado pela comparação entre o valor informado para o Fator de tolerância à Contradição Ft_{ct} e o valor resultante do processamento dos graus de evidência μ_1 e μ_2 informados na entrada. Uma saída S_1 com valor lógico Verdadeiro indica que os sinais são iguais; já a saída S_1 com valor lógico Falso indica uma desigualdade.

Um esquema de funcionamento da CNAPdi é mostrado na figura 17.

A sensibilidade do sistema é controlada pelo valor definido para o Ft_{ct} , podendo ser maior ou menor de acordo com a tolerância aos desvios definida pelos processos de qualidade envolvidos. Neste projeto utilizou-se um valor de $Ft_{ct} = 0,001$, considerando um processo bastante estrito com as dimensões dos objetos fabricados.

A CNAPdi utilizada recebe como valor de μ_1 o valor da área normalizado, enquanto μ_2 apresenta valor fixo igual ao valor da área normalizado nominal para o quadrado ou o valor da área normalizado nominal para o retângulo.

Em ambos os casos a equação 41 foi utilizada para a determinação dos respectivos valores normalizados.

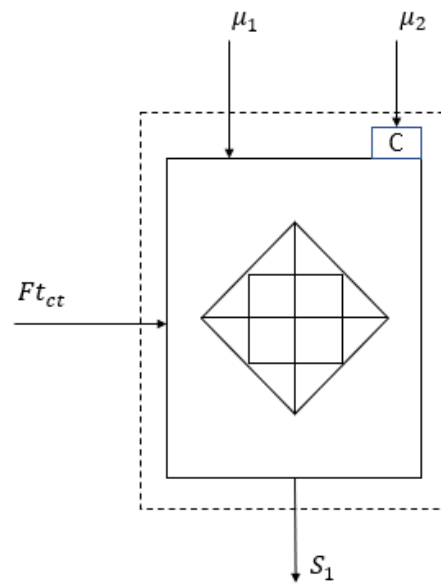


Figura 17 – Esquema de funcionamento da CNAPdi
Fonte: Adaptado de DA SILVA FILHO et al. (2008).

3 RESULTADOS E DISCUSSÃO

A seguir são apresentados os resultados de processamento das imagens para cada uma das etapas descritas no capítulo anterior. Sempre que pertinente, também é realizada uma discussão sobre o resultado apresentado, sendo feitas também comparações com outros trabalhos e/ou resultados semelhantes encontrados pelo processamento em métodos usualmente utilizados.

3.1 Seleção das imagens capturadas e conversão para cinza

As 50 imagens capturadas, descritas no capítulo 2, foram varridas visualmente para identificação de possíveis interferências.

Nesta etapa foram removidas as imagens cuja utilização do *flash* acabou sendo refletida na superfície dos objetos e que, por isso, criaram marcas na fotografia que pudessem causar erros no processamento subsequente, assim como fotografias nas quais a sombra também criou efeitos indesejados; em um único caso a imagem foi descartada por apresentar inconsistências no plano de fundo.

No total foram descartadas 08 imagens, sendo processadas, subsequentemente, 22 imagens do objeto quadrado e 20 imagens do retângulo, que compuseram o universo de imagens do processamento. Alguns exemplos de erros nas imagens são mostrados na figura 18.

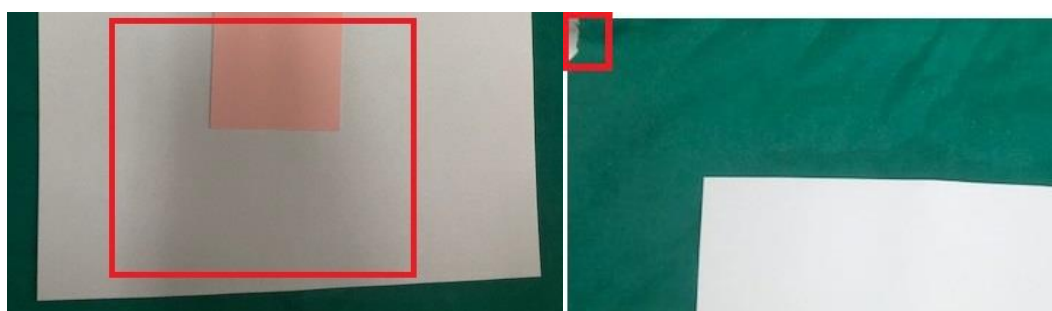


Figura 18 – Exemplos de erros nas imagens capturadas.

As imagens aprovadas para uso foram então convertidas para sua escala de cinza utilizando recursos da própria biblioteca utilizada, cujo resultado é exemplificado na figura 19.

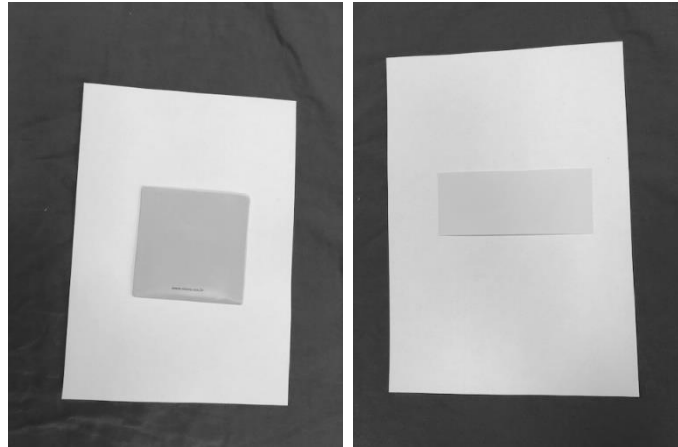


Figura 19 – Imagens convertidas para escala de cinza.

3.2 Histogramas de frequências dos níveis de cinza e cálculo da variância

A análise dos histogramas de frequências dos níveis de cinza, extraídos pela biblioteca OpenCV demonstra, à primeira vista, uma tendência de comportamento semelhante independente do objeto utilizado, como ilustrado na figura 20, em que o objeto quadrado é mostrado à esquerda e o retangular à direita, sendo o eixo horizontal o que representa a intensidade dos pixels e o eixo vertical, a frequência.

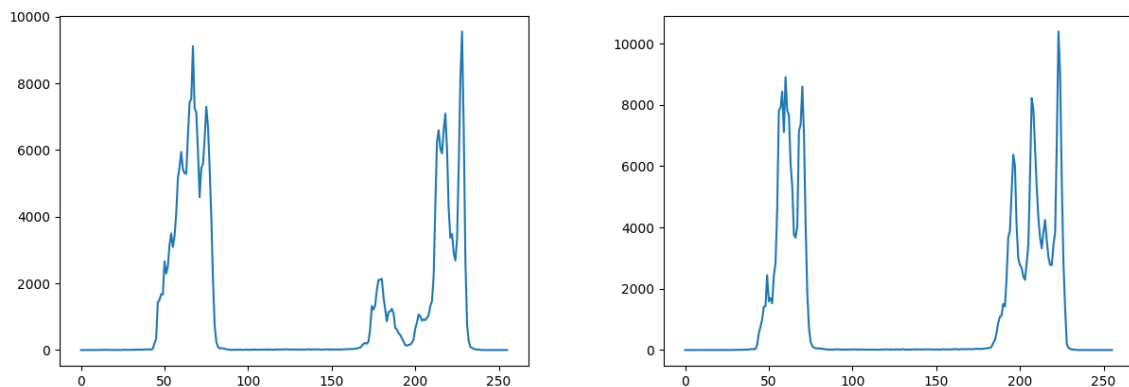


Figura 20 – Histogramas de níveis de cinza encontrado para os objetos quadrado (à esquerda) e retangular (à direita).

Em ambos existe uma concentração entre as intensidades entre 50 e 80, aproximadamente e, logo após, entre 170 e 230.

Chama a atenção, no entanto, a presença de uma frequência em intensidade entre 170 e 200 para o quadrado que é bem menos acentuada para o retângulo. Este comportamento foi constante em todos os histogramas analisados e pode ser explicado pela cor do objeto quadrado que, ao ser convertido em níveis de cinza,

possui um tom bastante escuro, porém, menos intenso que o fundo verde, enquanto o objeto retangular, quando convertido para escala de cinza, apresenta uma tonalidade clara, mais próxima à folha A4.

O cálculo das variâncias das imagens também revelou algumas tendências: enquanto o objeto quadrado apresentou valor de variância médio de $5215,9410 \pm 339,4528$, o objeto retangular apresentou valor médio de $5820,9890 \pm 381,0599$ – uma coesão um pouco menor, porém com valor médio de variância das intensidades significativamente superior, o que visualmente significa um contraste mais acentuado para a imagem retangular do que para o quadrado – explicado pelo fato de, no geral, a diferença entre *pixels* claros e escuros ser mais acentuada para o retângulo do que no quadrado; isto, porque o quadrado, cujas intensidades são escuras, parecidas com as do plano de fundo, e cuja área é maior que a do objeto retangular, acaba diminuindo os efeitos do papel A4, branco, na contribuição para baixos níveis de intensidade, na imagem.

3.3 Aplicação do Filtro Binomial

O filtro binomial aplicado através da utilização do MEEBi em conjunto com a UPC apresentou resultados visuais muito semelhantes aos resultados encontrados pela utilização do mesmo filtro, porém, via biblioteca OpenCV considerando um desvio-padrão $\sigma = 0,85$ e um *kernel* do tipo 3x3, tal como mostrado na figura 21.

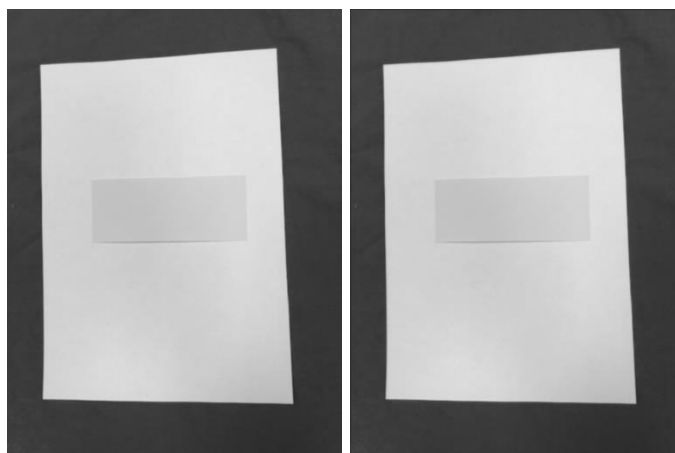


Figura 21 – Comparação entre imagens após aplicação do Filtro Binomial (utilizando biblioteca-padrão, à esquerda e via MEEBi + MC, à direita).

Observa-se na figura 21 o efeito imediato da aplicação do Filtro Binomial, que

é a presença de um borrão bastante discreto, acentuado em algumas regiões, principalmente nas quais se observa contraste de intensidades.

Apesar de os resultados visuais serem muito similares, o tempo de execução, calculado pela biblioteca *Time* da linguagem Python, é aqui um fator que merece destaque: enquanto a aplicação do Filtro Gaussiano via biblioteca demorou uma média de $0,014 \pm 0,005$ segundos (medidos para 10 amostras aleatórias) para ser executado, o processamento via MEEBi + MC levou uma média de $421,593 \pm 27,569$ segundos para as mesmas amostras, valor mais de 30.000 vezes superior.

A figura 22 mostra o histograma de frequências de níveis de cinza para uma figura aleatória, enquanto a figura 23 mostra o histograma de frequências de níveis de cinza para a mesma figura, porém, após aplicação do Filtro Binomial via MEEBi + MC (à esquerda) e via biblioteca OpenCV (à direita), com o eixo horizontal representando a intensidade dos pixels e o eixo vertical, a frequência.

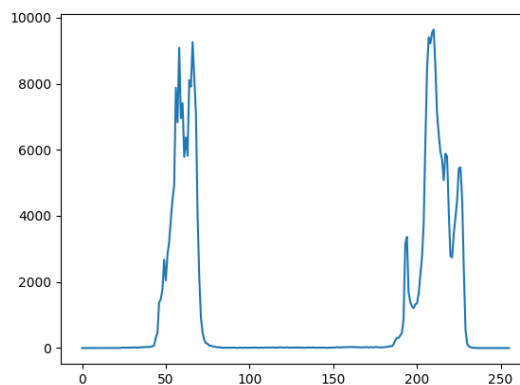


Figura 22 – Histogramas de intensidades de cinza encontrado para a figura aleatória.

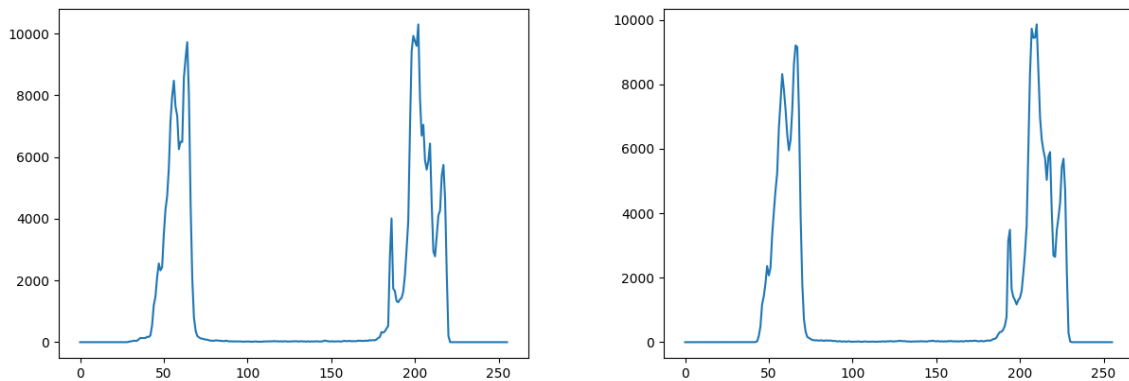


Figura 23 – Histogramas de intensidades de cinza encontrado para a figura após aplicação do Filtro Gaussiano – via MEEBi + MC (à esquerda) e via biblioteca OpenCV (à direita).

A comparação visual entre as figuras 22 e 23 evidencia uma suavização da distribuição de frequências dos níveis de cinza conforme esperado; este efeito é bastante evidente, por exemplo, quando se observam as frequências das intensidades entre 50 e 70 – tanto no histograma gerado para a imagem manipulada via MEEBi + MC como pela biblioteca OpenCV.

A imagem utilizada para a geração do histograma apresentado na figura 21 possui um valor de variância determinado diretamente pela biblioteca NumPy igual a 5704,79. Utilizando a mesma biblioteca, porém, determinando as variâncias das imagens processadas pelo Filtro Binomial via MEEBi + MC e via biblioteca OpenCV, os números são respectivamente de 5241,543 e 5665,9 – uma diminuição na variância da imagem de 8,12% utilizando o MEEBi + MC, contra 0,6% via biblioteca, o que ilustra, neste caso, uma boa vantagem da utilização da UPC em detrimento da biblioteca de larga utilização.

3.4 Aplicação do Filtro Gradiente, binarização e identificação de contornos

A comparação dos resultados da aplicação do Filtro Gradiente através da UPC e via biblioteca-padrão não pode ser feita diretamente; isto, porque o funcionamento de ambas é diferente, já que a LP não permite a utilização de valores negativos como graus de evidência de entrada.

Os resultados de da convolução utilizando o Operador Scharr para Δx e Δy é mostrado na figura 24; já o cálculo da magnitude para cada *pixel*, representando a imagem final após aplicação do Filtro Gradiente, é mostrado na figura 25, em que se observam, claramente, os contornos da folha A4 e do objeto.

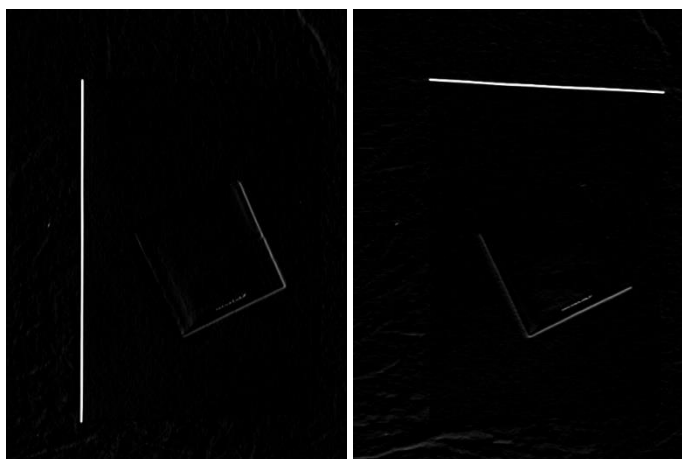


Figura 24 – Resultado da convolução com o Operador Scharr via biblioteca na direção Δx e Δy .

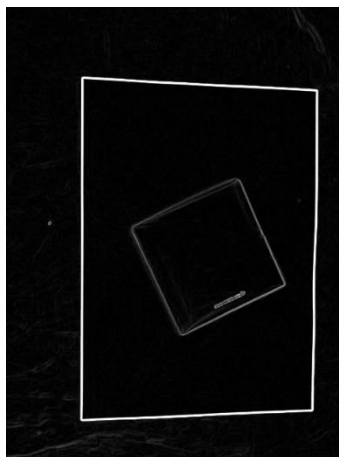


Figura 25 – Resultado da convolução com o Operador Scharr via biblioteca após cálculo das magnitudes.

A realização da convolução através da MEENi+UPC foi realizada conforme descrição feita no capítulo 2. Uma diferença importante em comparação com o uso da biblioteca OpenCV diz respeito à realização posterior de operações de binarização, de forma a isolar os contornos da imagem – sendo, portanto, uma operação realizada em dois passos distintos.

A execução da MEENi (antes da utilização do MoBi como auxiliar) mostrou-se bastante sensível à presença de imperfeições na imagem, como ilustrado na figura 26, em que a região com reflexo do *flash* ficou bem evidente (ainda que na imagem original este reflexo fosse visualmente desprezível).

Outro exemplo de sensibilidade identificado nesta etapa ocorreu na presença de uma falha no objeto quadrado, claramente visível após execução desta etapa – o que abre oportunidades de investigação do uso deste módulo em identificação de defeitos, por exemplo. Este caso é mostrado na figura 27.

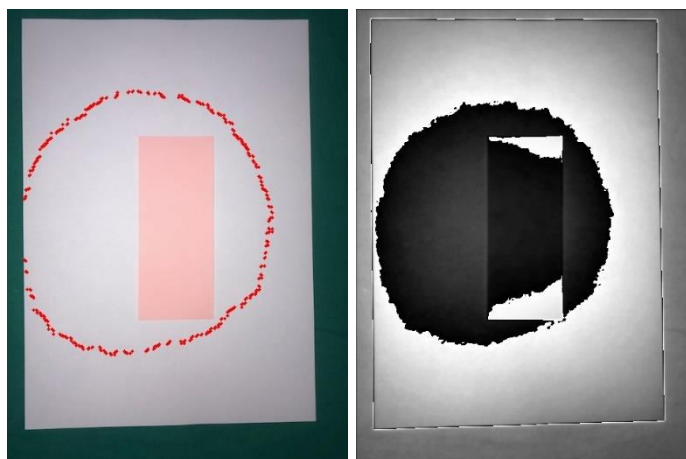


Figura 26 – Exemplo de sensibilidade da MEENi ao reflexo do *flash*.

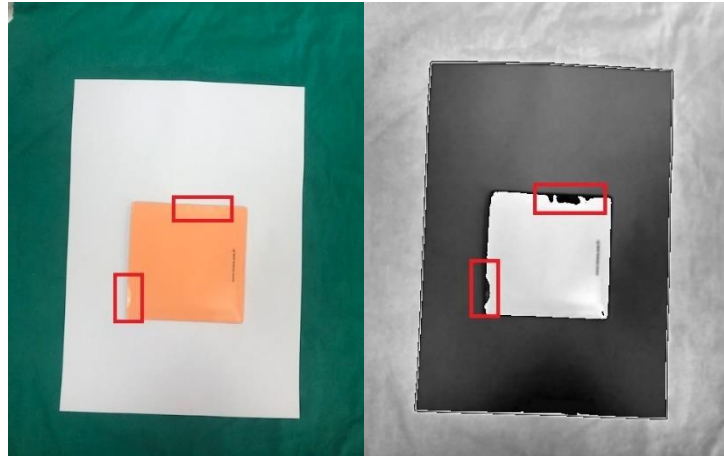


Figura 27 – Exemplo de sensibilidade da MEENi na presença de falhas no objeto.

A aplicação da MoBi mostrou-se um processo extremamente crítico, tendo em vista as diferentes condições sob as quais as fotografias foram tiradas influírem nas intensidades dos *pixels* presentes na imagem; assim, os limiares precisaram ser ajustados empiricamente, tendo os melhores resultados visuais sido encontrados para $T = 220$ para a primeira iteração, e $T = 255$ para a segunda.

O resultado encontrado possui mais imperfeições visíveis que seu correspondente (apresentado anteriormente). Processos de dilatação e erosão foram feitos em cima da imagem resultante com fins de diminuir os ruídos encontrados. Tanto a imagem binarizada sem tratamento e a imagem resultante dos processos de dilatação e erosão são mostradas na figura 28.

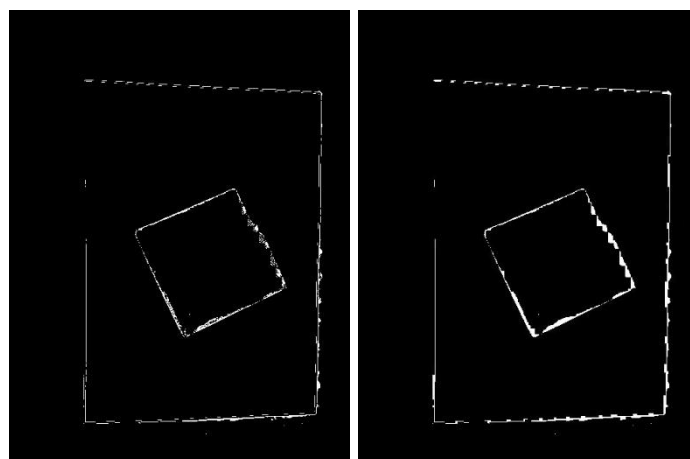


Figura 28 – Extração dos contornos via MEENi + UPC + MoBi (antes e após tratamento com processos de dilatação e erosão).

3.5 Extração das propriedades

A etapa de extração das propriedades das imagens englobou as ações de identificação dos polígonos presentes na fotografia através de aproximações disponíveis pela biblioteca OpenCV, a ordenação por área (de forma a se identificar o maior de todos – no caso, a folha A4), a mudança de perspectiva da imagem via *script* em linguagem Python, uma nova identificação dos polígonos, agora buscando encontrar o objeto quadrado ou o objeto retangular, e a extração das propriedades de perímetro e área.

A partir da imagem tratada pela MEENi + UPC + MoBi e da execução das funções de dilatação e erosão foram extraídos os contornos presentes na imagem, via código, considerando apenas os pontos mais extremos.

A manipulação das matrizes contendo as posições dos contornos externos permitiu indicá-los na figura na forma de pontos vermelhos ao redor da folha de papel, indicando a presença de uma aresta.

A matriz contendo os pontos do contorno externo foi considerada para a realização de uma transformação, que ajustou o conteúdo interno deste polígono à tela, removendo os elementos externos e reajustando a proporção na tela dos elementos internos, o que permitiu uma nova descoberta de contornos e posterior indicação na figura. Estes processos são ilustrados na figura 29.



Figura 29 – Determinação dos contornos e transformação de perspectiva.

O modo de extração das propriedades dos objetos foi baseado no trabalho de Mario et al. (2012) em que, com o auxílio da LP, foram estimadas as alturas de árvores em fotografias, com a premissa da presença de um objeto de referência de dimensões

conhecidas posicionado próximo a elas. Neste trabalho, o objeto de dimensões conhecidas é a folha A4, e são descritas pela altura e pela largura da própria imagem.

Já as distâncias entre os pontos mais externos foram calculadas a partir da obtenção dos dados dos vértices, permitindo a estimativa de suas dimensões reais considerando a proporção de distância da folha de papel, tal como mostrado nas figuras 30 e 31.

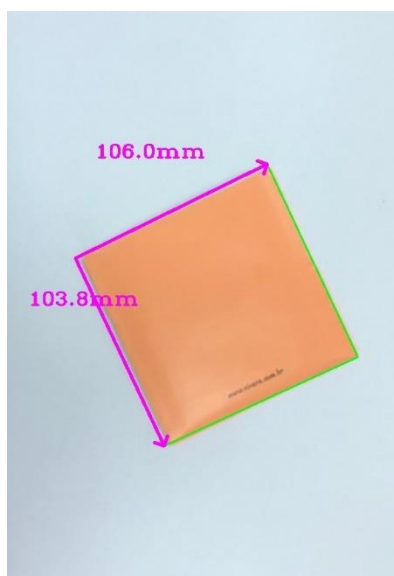


Figura 30 – Exemplo de dimensões obtidas para o objeto quadrado.

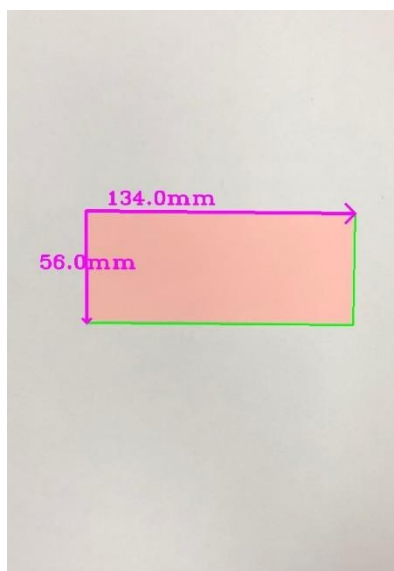


Figura 31 – Exemplo de dimensões obtidas para o objeto retangular.

A distribuição dos valores encontrados para o lado do objeto quadrado apresentou valor médio de $104,84 \pm 1,96$ mm, um erro de 0,1% considerando o valor nominal medido pela régua de alumínio; já sua mediana apresentou o valor exato de

105,0 mm. A área apresentou valor médio calculado de $10989,82 \pm 190,75 \text{ mm}^2$, um erro de 0,32% em relação à área teórica (de $11025,00 \text{ mm}^2$) e o perímetro apresentou valor de $419,37 \pm 3,64 \text{ mm}$, um erro de 0,15% em relação ao valor esperado (de $420,00 \text{ mm}$).

No caso do objeto retangular, a distribuição dos valores encontrados para a base apresentou valor médio de $133,58 \pm 0,97 \text{ mm}$ – um erro de 1,04% em relação ao valor nominal – enquanto a altura apresentou valor médio de $54,6 \pm 0,9 \text{ mm}$ – um erro de 0,72% em relação ao valor teórico; as medianas para a base e a altura encontradas foram de, respectivamente, $134,0 \text{ mm}$ e $55,0 \text{ mm}$. A área média do objeto retangular possui valor médio calculado de $7293,83 \pm 136,15 \text{ mm}^2$, um erro de 1,77% em comparação à área teórica (de $7425,00 \text{ mm}^2$), e o perímetro apresentou valor médio de $376,37 \pm 2,75 \text{ mm}$ – erro de 0,95% em comparação com o valor esperado (de $380,00 \text{ mm}$).

3.6 Classificação dos objetos

Os graus de evidência utilizados na CNAPP de classificação foram extraídos conforme descrição das equações 41 e 42, sendo considerado o valor $L = 135 \text{ mm}$, como o maior lado entre os polígonos, referindo-se à base do objeto retangular.

Foram retiradas 15 amostras aleatórias entre os resultados esperados. Delas, 10 foram utilizadas como conjunto de treinamento da CNAPP, de forma a se estabelecer o melhor limiar utilizado na classificação, e 5 foram utilizadas para teste.

Os resultados da execução do algoritmo da célula neural artificial paraconsistente padrão são mostrados nos quadros 1 e 2.

μ_1	μ_2	Forma	μ_r
0,770701	0,770741	Quadrado	0,770721
0,776994	0,777037	Quadrado	0,777015
0,770741	0,770741	Quadrado	0,770741
0,778116	0,778148	Quadrado	0,778132
0,766822	0,767037	Quadrado	0,766929

Quadro 1 – Graus de evidência para o objeto quadrado.

μ_1	μ_2	Forma	μ_r
0,63115	0,69259	Retângulo	0,66048
0,63011	0,6963	Retângulo	0,66158
0,63354	0,6963	Retângulo	0,66345
0,64167	0,7037	Retângulo	0,67122
0,63829	0,7037	Retângulo	0,66937

Quadro 2 – Graus de evidência para o objeto retangular.

A análise dos graus de evidência reais μ_r permite concluir que o objeto quadrado possui uma tendência a ter valores acima de 0,75 enquanto o objeto retangular tem a tendência de mostrar valores abaixo de 0,70. Ao estabelecer um limiar de classificação tal que $\mu_r \geq 0,71$ representa um objeto quadrado e um $\mu_r < 0,71$ representando um objeto retangular, os resultados mostrados no quadro 3 são encontrados para a classificação das imagens de teste.

μ_1	μ_2	μ_r	Forma esperada	Forma classificada
0,782508	0,782593	0,78255	Quadrado	Quadrado
0,770954	0,771111	0,771032	Quadrado	Quadrado
0,790877	0,791111	0,790994	Quadrado	Quadrado
0,629525	0,695926	0,661095	Retângulo	Retângulo
0,623906	0,69037	0,655531	Retângulo	Retângulo

Quadro 3 – Resultados encontrados nos conjuntos de teste para a classificação da forma geométrica via CNApp.

A matriz de confusão para os resultados encontrados para o conjunto de teste é mostrada na figura 32.

Valor real	Valor predito	
	Quadrado	Retângulo
Quadrado	3	0
Retângulo	0	2

Figura 32 – Matriz de confusão para a classificação realizada na CNApp.

De acordo com a matriz de confusão apresentada para os resultados de classificação do conjunto de testes pela CNApp, calculam-se os valores de acurácia, precisão, *recall* e *f-score*, todos iguais a 1 – valor máximo possível – indicativos de

uma grande capacidade de realização da classificação de forma correta.

3.7 Identificação de desvios

O objetivo da utilização da CNAPdi no sistema proposto é o de atuar como agente capaz de identificar desvios muito acentuados nas propriedades dos objetos em relação aos valores nominais, esperados, para cada categoria – quadrado ou retângulo.

Diferentemente da CNAPp utilizada na classificação dos objetos o objetivo do processamento dos graus de evidência, aqui, não é o de se determinar um limiar na saída ótimo a partir do qual um desvio é ou não caracterizado, mas sim trabalhar em sua sensibilidade – isto, através do ajuste do Ft_{ct} ótimo, o que é variável sob diferentes cenários e valores de erros aceitáveis.

Considerando os valores nominais de área para o quadrado igual a 11025 mm^2 e um fator de tolerância a contradição de $Ft_{ct} = 0,1$, por exemplo, a CNAPdi indicaria um desvio quando um objeto quadrado apresentasse área menor ou igual a $8122,7 \text{ mm}^2$ (lado $90,12 \text{ mm}$) ou maior ou igual a $14049,26 \text{ mm}^2$ (lado $118,53 \text{ mm}$) – um erro de cerca de 15% em ambos os casos; a utilização de um $Ft_{ct} = 0,05$, por sua vez, seria capaz de manter os desvios entre uma área menor ou igual a $9650,99 \text{ mm}^2$ (lado $98,23 \text{ mm}$) ou maior ou igual a $12485,71 \text{ mm}^2$ (lado $111,73 \text{ mm}$) – neste caso, um erro de cerca de 6,5%.

A figura 33 mostra um objeto quadrado, fabricado a partir do objeto original, para validar a capacidade da CNAPdi. Ele possui lado nominal igual a 104 mm – apenas 1 mm menor do que o quadrado original (sugerindo um erro de processo de 0,95%) conforme medições com a régua de alumínio. Extraídas suas propriedades via processamento, um lado L_1 foi determinado com tamanho $L_1 = 104,0 \text{ mm}$ e o outro, L_2 , com valor $L_2 = 104,1 \text{ mm}$. Sua área A foi calculada como $A = 10826,4 \text{ mm}^2$ e seu perímetro $P = 416,2 \text{ mm}$.

A classificação feita pela CNAPp, com graus de evidência já normalizados $\mu_1 = 0,77074$ e $\mu_2 = 0,77074$ o categorizou corretamente como um quadrado. Ao chegar na CNAPdi, contudo, e considerando um valor de tolerância à contradição $Ft_{ct} = 0,001$ e um valor de $\mu_2 = 0,77777$ (correspondente ao valor normalizado para a área nominal de $11025,0 \text{ mm}^2$), o valor de μ_{ctr} é calculado como $\mu_{ctr} = 0,4965$, inferior ao

valor de $V_{icct} = 0,49925$ determinado pelo Ft_{ct} indicando, aqui, um desvio não-aceitável para o processo, o que seria capaz de disparar um alerta no processo – neste caso, um limite bastante crítico, de apenas 1 mm.

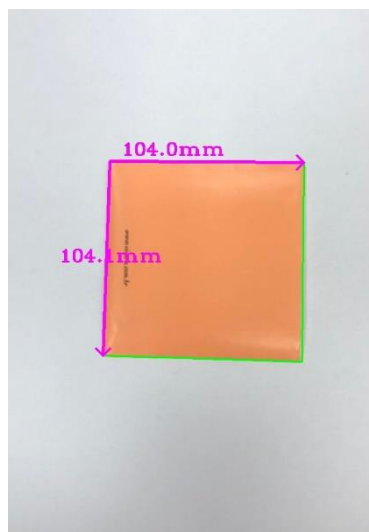


Figura 33 – Objeto quadrado modificado utilizado no teste da CNAPdi.

3.8 Discussão

O uso de métodos ópticos em controle de qualidade vem sendo bastante difundido nos últimos anos, principalmente devido aos avanços recentes em métodos de armazenamento e processamento de volumes de dados em tempo quase real, típicos dos cenários do chamado *Big Data* e da popularização da Internet das Coisas (IoT).

Zude (2008) exemplifica o uso de técnicas ópticas no monitoramento de safras com o uso de comprimentos de onda fora do espectro visível – como o infravermelho próximo (NIR) – e a partir de técnicas de fluorescência e dispersão, por exemplo. Na indústria, diferentes técnicas vêm sendo aplicadas em diferentes cenários, como o controle da umidade – estimada através de índices de refração ou via espectros em comprimentos de onda específicos, como apresentado por Daikos e Scherzer (2021). Já aplicações em controle de processo incluem, por exemplo, o uso no controle da emulsificação no processamento de alimentos (GONZÁLEZ-MARTINEZ et al., 2021).

A utilização de técnicas de visão computacional juntamente com métodos de inteligência artificial tem encontrado vasto campo de aplicação, e a fotogrametria, como exposto anteriormente, também vem recebendo grande atenção acadêmica, especialmente nos últimos anos.

Entre os algoritmos de visão computacional mais largamente aplicados, o Detector de Bordas apresentado por Canny (1986) – e suas modificações – figura entre os mais importantes. Através de um algoritmo de múltiplos estágios que utiliza o cálculo de variações como forma de otimizar a resposta da função, Canny demonstrou a possibilidade da detecção de bordas com boa resposta, boa localização e com o mínimo de ruídos.

Seu processamento inclui não apenas a utilização do Filtro Gaussiano como o cálculo das magnitudes das intensidades após calcular os gradientes nas direções Δx e Δy , e se utiliza etapas de supressão dos valores não-máximos em diferentes direções a partir de um *pixel* âncora, para diminuição dos ruídos. A etapa final, de identificação de contornos, é conseguida a partir de análises de histerese na imagem – um processo bastante complexo, porém, capaz de apresentar resultados extraordinários, identificando contornos contínuos e descontínuos (CANNY, 1986).

O protocolo de processamento de imagens utilizado neste trabalho foi baseado neste algoritmo, em que módulos da UPC foram criados conforme etapas específicas apresentadas por Canny.

Os resultados encontrados para a aplicação do Filtro Binomial na imagem foram especialmente importantes, uma vez que foi conseguida uma diminuição de variância mais de 13 vezes superior à da biblioteca-padrão, utilizada como comparação.

Cruz (2015) havia notado o fato de que, se por um lado a CNAPap não fornece valores aritmeticamente exatos, eles apresentam um erro-padrão da média que se encontra em uma distribuição normal para um índice de confiança de 99% e que este valor converge quão maior seja o número de iterações – e, por isto sua opção pelo uso de 6 CNAPaps ligadas em série, com 20 iterações cada.

Não se deve menosprezar o fato de que os valores de entrada considerados por si são valores medidos e que carregam consigo certo grau de incerteza – que seria ignorado caso os valores tivessem sido apenas somados, como ocorreria em uma convolução natural. Neste caso, o uso da LP – que considera a contradição como componente importante – mostrou-se bastante eficaz.

Conquanto que o resultado tenha sido positivo, deve-se atentar que a necessidade do uso de CNAPaps em série e repetidas iterações traz como contraponto um tempo de processamento bastante elevado em comparação com o que teria sido conseguido através de uma convolução ordinária – fato este que sem

dúvida deve ser considerado quando da opção pelo uso do modelo apresentado aqui.

Na etapa seguinte, ao contrário do que ocorreu com a suavização da imagem, a identificação das bordas da imagem trouxe mais ruídos do que o esperado, que precisaram ser corrigidos via código em momento posterior. Aqui não há como fazer comparações diretas entre o método executado e o que teria sido conseguido via biblioteca, uma vez que ambos os princípios de funcionamento são bastante diferentes.

Inesperadamente a combinação da utilização do MEENi + MoBi na imagem previamente processada permitiu a identificação de defeitos nos objetos, o que não foi explorado aqui, mas que certamente poderia ser aperfeiçoado de forma a gerar mais um alarme de controle de qualidade, extremamente útil em uma linha de produção.

A grande vantagem é que a utilização do protocolo de processamento paraconsistente das imagens permite o processamento da imagem *pixel a pixel*, enquanto os métodos de identificação do OpenCV utilizados como comparação trabalham com aproximações, que ignoram defeitos de menor magnitude – como os encontrados aqui.

A etapa de extração das propriedades também mostrou ser positiva, com um erro encontrado para as duas figuras entre 0,1% e 1,77%, ambos valores muito baixos em comparação com os valores nominais medidos pela régua de alumínio, o que demonstra um grande poder de aplicabilidade da técnica aqui desenvolvida em situações reais, práticas.

A classificação dos objetos via CNAPp também trouxe resultados satisfatórios e demandou apenas a execução de um algoritmo de baixa complexidade computacional, com limiares de resultados com boa separação – não havendo, entre os conjuntos de treinamento e teste – qualquer sobreposição entre eles. Sua taxa de acerto foi de 100% com o conjunto de teste, e seu f-score também representou o valor máximo, indicativo muito importante de seu sucesso.

Normalmente a classificação de imagens é realizada através de RNAs com algoritmos de *backpropagation*; estas, envolvem a definição de funções de ativação e funções de perda a partir das quais os mínimos locais são encontrados através de métodos numéricos de otimização através de redes multicamadas – isto é, com um número variável de neurônios, superior a 1.

A ideia do *backpropagation* envolve um processo de pelo menos duas fases:

o cálculo dos erros, propagados entre as camadas de neurônios, e seu retorno para ajuste dos pesos sinápticos utilizados. Este processo é repetido pelo número de vezes necessário até que a classificação da imagem possa ser feita; isto, em contrapartida da execução única do algoritmo no CNAPp.

Além do *backpropagation*, imagens também são processadas através de redes neurais convolucionais (CNN) – que baseiam seu funcionamento em convoluções, tal como a abordagem deste projeto, e na estratégia de propagação de erros já mencionada – agora, no âmbito da aprendizagem profunda, que exige métodos computacionalmente mais intensivos.

Por fim, a identificação de desvios através da CNAPdi também traz como grande benefício a possibilidade de utilização de um algoritmo pouco intensivo, composto por apenas 8 passos – como mostrado no Anexo B.

A utilização do parâmetro área como alarme nos desvios se deu devido à sua maior sensibilidade à presença de erros nos lados das figuras do que o perímetro, uma vez que sua propagação é multiplicada. No cenário proposto, um objeto modificado manualmente com uma diferença de apenas 1 mm do objeto original – esperado – foi capaz de ativar um alarme de desvio, valor que representa menos de 1% do lado do objeto utilizado – importante indicativo de sua aplicabilidade.

4 CONCLUSÕES

Este trabalho demonstrou com sucesso a possibilidade da utilização da LP em um contexto de análise e manipulação de imagens digitais, aplicados a um cenário de controle de qualidade típico, normalmente encontrado em esteiras de produção em diferentes indústrias.

A utilização da LP no controle de processo já havia sido discutida previamente em trabalhos como os de Cruz (2015), que desenvolveu módulos capazes de tratar estatisticamente dados de entrada. Estes, foram aproveitados nesta dissertação para a geração de estruturas próprias para a transformação de imagens digitais, a partir dos princípios da aplicação de convolução, utilizando conjuntos de pesos-padrão a partir dos quais cada módulo foi desenvolvido.

Ainda que diferentes trabalhos realizando a ligação entre o controle de qualidade e a LP e ações de classificação de imagens já tenham sido apresentados, sua aplicação junto à fotogrametria como forma de se lograr um monitoramento óptico em controle de qualidade é pioneira, sendo a literatura acadêmica carente deste assunto.

A aplicação de técnicas ópticas de monitoramento traz como principal benefício a possibilidade de sua execução em tempo próximo ao real e à distância, de forma automatizada, sem a necessidade de intervenção humana – como, por exemplo, a retirada de amostras para serem levadas a um laboratório e terem seus resultados utilizados como representativos de um lote. Aqui, o controle pode ser individualizado, com a geração contínua de relatórios e dados de medição sobre cada um dos objetos analisados.

A grande vantagem do uso da LP refere-se à consideração da incerteza como informação de importância na realização dos cálculos; uma vez que esta é inerente aos processos de medição e que seus erros são propagados, seus efeitos devem ser considerados durante a computação. Um exemplo benéfico muito claro de sua utilização se deu pela diminuição da variância da imagem-problema 13 vezes mais acentuada do que quando a suavização foi realizada utilizando uma biblioteca de referência.

As etapas de identificação de contornos e extração de imagens também apresentaram resultados positivos e, aqui, salientam-se a possibilidade de detecção

de falhas nos objetos analisados a partir do processamento paraconsistente em detrimento do processamento via OpenCV e um baixo custo computacional nas etapas de classificação das imagens e detecção de desvios, para os quais foram obtidos taxas de acerto de 100%, para a classificação, e a possibilidade da identificação de erros de fabricação de objetos de apenas 1 mm.

4.1 Trabalhos futuros

Ao passo que esta dissertação ajudou a preencher uma lacuna existente na literatura acadêmica sobre a utilização conjunta da LP com a fotogrametria, ela também permitiu identificar a geração de novas hipóteses de estudo e oportunidades de melhoria, abrindo a possibilidade para sua utilização em trabalhos futuros. Entre elas, mencionam-se:

- a) Otimização dos custos computacionais do processamento de imagens utilizando a abordagem apresentada na UPC;
- b) Utilização do cálculo diferencial paraconsistente como forma de obtenção das derivadas primeiras entre os *pixels* durante a etapa de determinação das arestas, visto os algoritmos da LP não permitirem a utilização de valores negativos como graus de evidência, o que impediu, por exemplo, a aplicação direta de operadores como o Sobel e Scharr;
- c) Estudos sobre a aplicação do processamento paraconsistente de imagens para identificação de defeitos em objetos, efeito inesperado, porém observado durante a elaboração deste projeto;
- d) Realização de estudos com um número maior de amostras, como forma de validação da manutenção da efetividade em cenários mais complexos e sob condições com menor controle sobre os parâmetros;
- e) Utilização de objetos com diferentes formas geométricas, mais complexas, para avaliação da manutenção das taxas de efetividade e determinação de diferentes limiares de classificação. Aqui, citam-se também os objetos com altura não-desprezível, em que os efeitos de profundidade podem influir nos resultados encontrados.

REFERÊNCIAS

ABDOLLAHNEJAD, A; PANAGIOTIDIS, D. **Tree Species Classification and Health Status Assessment for a Mixed Broadleaf-Conifer Forest with UAS Multispectral Imaging**. Remote Sensing. Vol 12, ed 22, artigo 3722, 2020. DOI: 10.3390/rs12223722.

ALBERTZ, J. **Albrecht Meydenbauer – Pioneer of Photogrammetric Documentation of the Cultural Heritage**; Proceedings 18th International Symposium CIPA 2001, Potsdam (Alemanha).

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS – ABNT. **ABNT NBR ISO9000:2015: Sistemas de gestão da qualidade – Fundamentos e vocabulário**. Rio de Janeiro, 2015.

BOROUJENI, MN; SANIMI, Y; ROGHANIAM, E. **Monitoring fuzzy linear quality profiles: A comparative study**. International Journal of Industrial Engineering Computations. Vol 12, Ed. 1, pp. 37-48, 2021. DOI: 10.5267/j.ijiec.2020.10.002.

BREGAR, T; ZALETELJ, K; CEPON, G; SLAVIC, J; BOLTEZAR, M. **Full-field FRF estimation from noisy high-speed-camera data using a dynamic substructuring approach**. Mechanical Systems and Signal Processing. Vol 150, artigo 107263, 2021. DOI: 10.1016/j.ymssp.2020.107263.

CANNY, J. **A Computational Approach to Edge Detection**. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. PAMI-8, 6, pp. 679-698, 1986. DOI:10.1109/TPAMI.1986.4767851.

CATALUCCI, S; SENIN, N; SIMS-WATERHOUSE, D; ZIEGELMEIER, S; PIANO, S; LEACH, R. **Measurement of complex freeform additively manufactured parts by structured light and photogrammetry**. Measurement. Vol 164, artigo 108081, 2020. DOI: 10.1016/j.measurement.2020.108081.

CRUZ, C. Misseno da. **Aplicação da rede neural artificial paraconsistente em controle estatístico de processos**. 2015. 76 f. Dissertação (Mestrado em engenharia mecânica) – Universidade Santa Cecília, Santos, 2015.

DA COSTA, N.C.A; SUBRAHMANYAN, V.S.; VAGO, C. **The Paraconsistent Logics PJ**. Mathematical Logic Quarterly, 37:139-148. DOI: <https://doi.org/10.1002/malq.19910370903>.

DA SILVA FILHO, J.I. **Noções básicas da lógica paraconsistente anotada**. Santos: Universidade Santa Cecília, 2020. Notas de Aula.

DA SILVA FILHO, J. I.; ABE, J. M. **Fundamentos das redes neurais artificiais paraconsistentes – Destacando Aplicações em Neurocomputação**. São Paulo: Editora Arte & Ciência, 2001.

DA SILVA FILHO, J. I.; ABE, J. M.; TORRES, G, L. **Inteligência Artificial com as**

Redes de Análises Paraconsistentes Teoria e Aplicações. Rio de Janeiro: Editora LTC – Livros Técnicos e Científicos, 2008.

DAIKOS, O.; SCHERZER, T. **Monitoring of the residual moisture content in finished textiles during converting by NIR hyperspectral imaging.** Talanta. Vol 221, artigo 121561, 2021. DOI: 10.1016/j.talanta.2020.121567.

DAMIN, O. C. B.; DA SILVA FILHO, J. I. **Utilização da Distribuição de Frequência Estatística Associada à Lógica Paraconsistente Anotada de 2 valores: o Modelo de dois Peritos em Vistorias de Engenharia.** Revista Seleção Documental, n.26, a.7, p.13-20, 2012.

DURAN, Z; ATIK, ME. **Accuracy comparison of interior orientation parameters from different photogrammetric software and direct linear transformation method.** International Journal of Engineering and Geosciences. Vol 6, ed 2, pp. 74-80, 2021. DOI: 10.26833/ijeg.691696.

EL MASRI, Y; RAKHA, T. **A scoping review of non-destructive testing (NDT) techniques in building performance diagnostic inspections.** Construction and Building Materials. Vol 265, artigo 120542, 2020. DOI: 10.1016/j.conbuildmat.2020.120542.

FÖRSTNER, Wolfgang e WROBEL, Bernhard P. **Photogrammetric Computer Vision: Statistics, Geometry, Orientation and Reconstruction.** Estados Unidos: Springer, 2016. ISBN 978-3319115498.

GONÇALVES, E.D. **Desenvolvimento de um modelo estruturado em lógica paraconsistente para análise e tomada de decisão na escolha de materiais a serem montados nos protótipos e pré-produção de uma empresa montadora de veículos automotores.** 2019. 75 f. Dissertação (Mestrado em engenharia mecânica) – Universidade Santa Cecília, Santos, 2019.

GONZALEZ-MARTINEZ, Z.; ZAMORA, A.; MOR-MUR, M.; CASTILLO, M. **An in-line optical technology to control the emulsification degree in a continuous industrial emulsifier for meat sausage production.** Food and Science Technology, 137, artigo 110562, 2021. DOI: 10.1016/j.lwt.2020.110562.

GUIDI, G; MALIK, US; MICOLI, LL. **Optimal Lateral Displacement in Automatic Close-Range Photogrammetry.** Sensors. Vol 20, ed 21, artigo 6280, 2020. DOI: 10.3390/s20216280.

HARRISON, LM; HALLAC, RR; DERDERIAN, CA. **Three-Dimensional Analysis of Bilateral Cleft Lip and Palate Nasal Deformity.** Cleft Palate-Craniofacial Journal. Vol 58, ed 1, pp 105-113, 2021. DOI: 10.1177/1055665620940190.

LIU, W; SU, FL. **Unsupervised Adversarial Domain Adaptation Network for Semantic Segmentation.** IEEE Geoscience and Remote Sensing Letters. Vol 17, ed 11, pp 1978-1982, 2020. DOI: 10.1109/LGRS.2019.2956490.

LOPES, H. F. S. **Aplicação de redes neurais artificiais paraconsistentes como método de auxílio no diagnóstico da doença de Alzheimer**. 2009. 84 f. Dissertação (Mestrado em ciências) – Universidade de São Paulo, São Paulo, 2009.

MA, Y. CHEN, G; LV, MS; YI, W; LIU, X; CHEN, H; ZHU, B. **Efficient and Effective Dimension Control in Automotive Applications**. IEEE Transactions on Industrial Informatics. Vol 17, ed 3, pp. 1583-1591, 2021. DOI: 10.1109/TII.2020.2990168

MARTINELLI, Fernando Baracho. **Gestão da qualidade total**. Curitiba: IESDE, 2009.

MARIO, MC; DA SILVA FILHO, JI; DAMIN, OCB; LOPES, RGC. **Modelo digital com interação do usuário para obtenção de medidas de árvores utilizando Lógica Paraconsistente Anotada**. Revista Seleção Documental. N.28. Ano 7. Santos, 2012

MATA-FALCON, J; HAEFLIGER, S; LEE, M; GALKOVSKI, T; GEHRI, N. **Combined application of distributed fibre optical and digital image correlation measurements to structural concrete experiments**. Engineering Structures. Vol 225, artigo 111309, 2020. DOI: 10.1016/j.engstruct.2020.111309.

MENNA, F; NOCERINO, E; REMONDINO, F; SALADINO, L; BERRI, L. **Towards online UAS-based photogrammetric measurements for 3D metrology inspection**. Photogrammetric Record. 2020. DOI: 10.1111/phor.12338.

MONTGOMERY, D. C. **Introdução ao controle estatístico da qualidade**. 4ª ed. Rio de Janeiro: LTC, 2012.

NARVYDAS, E; PUODZIUNIENE, N. **Curved beam stress analysis using a binary image of the cross-section**. Computer Applications in Engineering Education. Vol 28, ed 6, pp 1696-1707, 2020. DOI: 10.1002/cae.22348.

ODUMOSU, JO; ADELEKE, OO; NNAM, VC; AVOSEH, OJ; OGUNDEJI, OA. **Stereophotogrammetry for 2-D building deformation monitoring using Kalman Filter**. Reports on Geodesy and Geoinformatics. Vol 110, ed 1, pp 1-7, 2020. DOI: 10.2478/rgg-2020-0006.

OZTURK, HI; RASHIDZADE, I. **A photogrammetry based method for determination of 3D morphological indices of coarse aggregates**. Construction and Building Materials. Vol 262, artigo 120794, 2020. DOI: 10.1016/j.conbuildmat.2020.120794.

PALADINI, E. P. **Avaliação estratégica da qualidade**. 2ª ed. São Paulo: Atlas, 2011.

ROCHA, Daniele Aline Pessoa da. **Aplicação de métodos estatísticos com apoio da lógica paraconsistente para o estudo de caso da confiabilidade de ativos industriais**. 2019. 95 f. Dissertação (Mestrado em engenharia mecânica) – Universidade Santa Cecília, Santos, 2019.

SINGH, G., SINGH, K. **Digital image forensic approach based on the second-order statistical analysis of CFA artifacts**. Forensic Science International: Digital Investigation 32, 200899, 2020. doi:10.1016/j.fsidi.2019.200899.

SOUZA, S. **Sistema de reconhecimento de caracteres numéricos manuscritos baseado nas redes neurais artificiais paraconsistentes**. 2013. 71f. Dissertação (Mestrado) – Faculdade de Medicina da Universidade de São Paulo. Programa de Fisiopatologia Experimental, São Paulo, 2013.

STACHNISS, Cyrill. **Photogrammetry I & II**. Alemanha: University of Bonn, 2015. Notas de aula.

TAN, CP; FENG, SQ; ZHAO, XM; SHAN, X; FENG; SL. **Longitudinal variations in channel morphology of an ephemeral stream from upland to lowland, Daihai Lake basin, North China**. Geomorphology. Vol 372, artigo 107450, 2021. DOI: 10.1016/j.geomorph.2020.107450.

TAROLLI, P; PIJL, A; CUCCHIARO, S; WEI, W. **Slope instabilities in steep cultivation systems: Process classification and opportunities from remote sensing**. Land Degradation & Development. 2020. DOI: 10.1002/ldr.3798.

WANG, JS; TAO, B; GONG, ZY; YU, SP; YIN, ZP. **A Mobile Robotic Measurement System for Large-scale Complex Components Based on Optical Scanning and Visual Tracking**. Robotics and Computer-Integrated Manufacturing. Vol 67, artigo 102010, 2021. DOI: 10.1016/j.rcim.2020.102010.

WU, B; LI, Y; LIU, WC; WANT, YR; LI, F; ZHAO, Y; ZHANG; H. **Centimeter-resolution topographic modeling and fine-scale analysis of craters and rocks at the Chang'E-4 landing site**. Earth and Planetary Science Letters. Vol 553, artigo 116666, 2021. DOI: 10.1016/j.epsl.2020.116666.

XU, XY; SHI, PX; ZHOU, XQ; LIU, W; YANG, H; WANG, T; YAN, MJ; FAN, WB. **A novel vision measurement system for health monitoring of tunnel structures**. Mechanics of Advanced Materials and Structures. 2020. DOI: 10.1080/15376494.2020.1854908.

YU, LP; LUBINEAU, G. **A smartphone camera and built-in gyroscope based application for non-contact yet accurate off-axis structural displacement measurements**. Measurement. Vol 167, artigo 108449, 2021. DOI: 10.1016/j.measurement.2020.108449.

ZOLLINI, S; ALICANDRO, M; GIALONARDO, M. **UAV Photogrammetry for Concrete Bridge Inspection Using Object-Based Image Analysis (OBIA)**. Remote Sensing. Vol 12, ed 19, artigo 3180, 2020. DOI: 10.3390/rs12193180.

ZUDE, M. **Optical Monitoring of Fresh and Processed Agricultural Crops**. 1ª edição. Estados Unidos: CRC Press, 2008. ISBN 9781420054026.

APÊNDICE A – Algoritmo da Célula Neural Artificial Paraconsistente padrão

1. Entre com o valor do Grau de Evidência de entrada 1

$$\mu_1 \text{ tal que } 0 \leq \mu_1 \leq 1$$

2. Entre com o valor do Grau de Evidência de entrada 2

$$\mu_2 \text{ tal que } 0 \leq \mu_2 \leq 1$$

3. Entre com o valor do Fator de tolerância à Contradição

$$Ft_{ct} = C_1 \text{ tal que } 0 \leq Ft_{ct} \leq 1$$

4. Calcule os valores superior e inferior de controle de Contradição

$$V_{scct} = \frac{1+C_1}{2} \text{ e } V_{icct} = \frac{1-C_1}{2}$$

5. Entre com o valor do Fator de tolerância à Certeza

$$Ft_c = C_2 \text{ tal que } 0 \leq Ft_c \leq 1$$

6. Calcule os valores superior e inferior de controle de Certeza

$$V_{scc} = \frac{1+C_2}{2} \text{ e } V_{icc} = \frac{1-C_2}{2}$$

7. Entre com o valor do Fator de tolerância à Decisão

$$Ft_D = C_3 \text{ tal que } 0 \leq Ft_D \leq 1$$

8. Calcule o valor superior e o valor inferior de Decisão

$$Vl_v = \frac{1+C_3}{2} \text{ e } Vl_f = \frac{1-C_3}{2}$$

9. Entre com o valor do Fator de Aprendizagem

$$F_A = C_4 \text{ tal que } 0 \leq F_A \leq 1$$

10. Transforme o Grau de Evidência 1 em Grau de Evidência desfavorável

$$\lambda_1 = 1 - \mu_1 \text{ tal que } 0 \leq \lambda_1 \leq 1$$

11. Transforme o Grau de Evidência 2 em Grau de Evidência desfavorável

$$\lambda_2 = 1 - \mu_2 \text{ tal que } 0 \leq \lambda_2 \leq 1$$

12. Calcule o Grau de Contradição normalizado

$$\mu_{ctr} = \frac{\mu_1 + \lambda_2}{2}$$

13. Calcule o Intervalo de Evidência resultante

$$\varphi_E = 1 - |2\mu_{ctr} - 1|$$

14. Calcule o Grau de Evidência resultante

$$\mu_E = \frac{(\mu_1 + \lambda_2) + 1}{2}$$

15. Calcule a complementação do Grau de Evidência resultante

$$\mu_{EC} = 1 - \mu_E$$

16. Calcule a distância D

$$D = \sqrt{(1 - |\mu - \lambda|)^2 + ((\mu + \lambda) - 1)^2}$$

17. Calcule o Grau de Evidência resultante real

$$\text{Se } \mu_E > 0,5 \rightarrow \mu_{ER} = \left(1 - \frac{D}{2}\right)$$

$$\text{Se } \mu_E < 0,5 \rightarrow \mu_{ER} = \frac{D}{2}$$

$$\text{Se } \mu_E = 0,5 \rightarrow \mu_{ER} = \mu_E$$

18. Faça a complementação do Grau de Evidência resultante real

$$\mu_{ERC} = 1 - \mu_{ER}$$

19. Apresente os sinais resultantes das saídas a partir das condicionais:

$$\text{Se } V_{scct} > \mu_{ctr} > V_{icc} \text{ e } V_{sc} \leq \mu_{ER} \leq V_{icc}, \text{ então } S_1 = \mu_E \text{ e } S_2 = \varphi_E$$

$$\text{Senão, } S_1 = 0,5 \text{ e } S_2 = \varphi_E$$

20. Faça $Ft_{ct} = 1$ e apresente na saída

$$\text{Se } \varphi_E \leq 0,25, \text{ então: } S_1 = 0,5 \text{ e } S_2 = \varphi_E$$

$$\text{Senão, } S_1 = 0,5 \text{ e } S_2 = 0,5$$

21. Faça $Ft_{ct} = 1$ e apresente na saída

$$\text{Se } V_{scct} \leq \mu_{ER} \leq V_{icct}, \text{ então: } S_1 = \mu_1 \text{ e } S_2 = 0,5$$

$$\text{Senão, } S_1 = \mu_E \text{ e } S_2 = \varphi_E$$

22. Faça $Ft_{ct} = 1$ e $Ft_c = 0$ e apresente na saída o valor do Grau de Evidência aprendido

$$S_1 = \mu_{ER(K+1)} \text{ e } S_2 = 0,5$$

23. Faça $Ft_{ct} = 1$ e $Ft_c = 0$ e apresente os sinais resultantes das saídas a partir das condicionais:

Para maximização:

$$\text{Se } \mu_E > 0,5, \text{ faça } S_1 = \mu_1 \text{ e } S_2 = 0,5$$

$$\text{Se } \mu_E < 0,5, \text{ faça } S_1 = \lambda_2 \text{ e } S_2 = 0,5$$

Para minimização:

$$\text{Se } \mu_E > 0,5, \text{ faça } S_1 = \lambda_2 \text{ e } S_2 = 0,5$$

$$\text{Se } \mu_E < 0,5, \text{ faça } S_1 = \mu_1 \text{ e } S_2 = 0,5$$

24. Faça $Ft_{ct} = 1$ e $Ft_c = 0$ e apresente os sinais resultantes das saídas a partir das condicionais:

Para maximização:

Se $\mu_E > 0,5$, faça $S_1 = \mu_1$ e $S_2 = 0,5$

Se $\mu_E < 0,5$, faça $S_1 = \lambda_2$ e $S_2 = 0,5$

Para minimização:

Se $\mu_E > 0,5$, faça $S_1 = \lambda_2$ e $S_2 = 0,5$

Se $\mu_E < 0,5$, faça $S_1 = \mu_1$ e $S_2 = 0,5$

25. Faça $Ft_{ct} = 1$ e $Ft_c = 0$ e apresente os sinais resultantes das saídas a partir das condicionais:

Para maximização

Se $\mu_E > 0,5$, $\rightarrow \mu_1 > \mu_2$, resulta nas saídas em: $S_1 = \mu_1$ e $S_2 = 0,5$

Senão $S_1 = 0,5$ e $S_2 = \mu_2$

Para minimização

Se $\mu_E < 0,5$, $\rightarrow \mu_1 < \mu_2$, resulta nas saídas em: $S_1 = \mu_1$ e $S_2 = 0,5$

Senão $S_1 = 0,5$ e $S_2 = \mu_2$

26. Faça $Ft_c = 0$ e apresente os sinais resultantes das saídas a partir das condicionais:

Se $V_{scct} > \mu_{ctr} > V_{icct}$, então: $S_1 = 1$ e $S_2 = \varphi_E$

Senão $S_1 = 0$ e $S_2 = \varphi_E$

27. Faça $Ft_c = 0$ e apresente os sinais resultantes das saídas a partir das condicionais:

Se $\mu_E \geq Vl_V$, então: $S_1 = 1$

Se $\mu_E \leq Vl_F$, então: $S_1 = 0$

Senão $S_1 = 0,5$

28. Fim.

APÊNDICE B – Algoritmo da Célula Neural Artificial Paraconsistente de detecção de igualdade

1. Entre com o valor do Grau de Evidência de entrada 1

$$\mu_1 \text{ tal que } 0 \leq \mu_1 \leq 1$$

2. Entre com o valor do Grau de Evidência de entrada 2

$$\mu_2 \text{ tal que } 0 \leq \mu_2 \leq 1$$

3. Entre com o valor do Fator de tolerância à Contradição

$$Ft_{ct} = C_1$$

4. Calcule os valores superior e inferior de controle de Contradição

$$V_{scct} = \frac{1+C_1}{2} \text{ e } V_{icct} = \frac{1-C_1}{2}$$

5. Transforme o Grau de Evidência 2 em Grau de Evidência desfavorável

$$\lambda_2 = 1 - \mu_2 \text{ tal que } 0 \leq \lambda_2 \leq 1$$

6. Calcule o Grau de Contradição normalizado

$$\mu_{ctr} = \frac{\mu_1 + \lambda_2}{2}$$

7. Apresente os sinais resultantes das saídas a partir das seguintes condicionais:

$$\text{Se } V_{scct} > \mu_{ctr} > V_{icct}, \text{ então } S_1 = 1$$

$$\text{Senão, } S_1 = 0$$

8. Fim.