

**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS TIMÓTEO**

Wendel de Carvalho Nogueira

**APLICAÇÃO DE UM MÉTODO EXATO NA DETECÇÃO DA ÁREA DE
INTERESSE EM IMAGEM DIGITAL.**

Timóteo

2020

Wendel de Carvalho Nogueira

APLICAÇÃO DE UM MÉTODO EXATO NA DETECÇÃO DA ÁREA DE INTERESSE EM IMAGEM DIGITAL

Trabalho de Conclusão de Curso apresentado ao Curso de Engenharia de Computação do Centro Federal de Educação Tecnológica de Minas Gerais, campus Timóteo, como requisito parcial para obtenção do título de Engenheiro de Computação.

Trabalho aprovado. Timóteo, 04 de dezembro de 2020:

Prof. Me. Odilon Corrêa da Silva
Orientador

Prof. Me. Douglas Nunes de Oliveira
Professor Convidado

Prof. Me. Fabrício Almeida de Castro
Professor Convidado

Prof. Dr. João Batista Queiroz Zuliani
Professor Convidado

Timóteo
2020



Emitido em 12/12/2020

FOLHA DE ROSTO (PLATAFORMA BRASIL) Nº 3/2020 - DCCTM (11.63.05)

(Nº do Protocolo: NÃO PROTOCOLADO)

(Assinado digitalmente em 12/12/2020 10:50)

DOUGLAS NUNES DE OLIVEIRA
PROFESSOR ENS BASICO TECN TECNOLOGICO
DCCTM (11.63.05)
Matrícula: 2921288

(Assinado digitalmente em 12/12/2020 10:22)

FABRICIO ALMEIDA DE CASTRO
PROFESSOR ENS BASICO TECN TECNOLOGICO
DFGTM (11.63.03)
Matrícula: 2146238

(Assinado digitalmente em 12/12/2020 08:34)

JOAO BATISTA QUEIROZ ZULIANI
PROFESSOR ENS BASICO TECN TECNOLOGICO
DFGTM (11.63.03)
Matrícula: 1324186

(Assinado digitalmente em 12/12/2020 00:12)

ODILON CORREA DA SILVA
PROFESSOR ENS BASICO TECN TECNOLOGICO
DCCTM (11.63.05)
Matrícula: 2794495

Para verificar a autenticidade deste documento entre em <https://sig.cefetmg.br/documentos/> informando seu número:
3, ano: **2020**, tipo: **FOLHA DE ROSTO (PLATAFORMA BRASIL)**, data de emissão: **12/12/2020** e o código de
verificação: **4a6dee84b0**

Dedico a minha Mãe,
Maria Vieira de Carvalho Nogueira,
que sempre esteve por mim.

Agradecimentos

Agradeço a todos que de forma direta ou indireta me ajudaram a chegar onde estou, a me tornar quem sou e desvelar-me quem quero ser.

*“Os grandes navegadores
devem sua reputação aos temporais e tempestades”.*
Epicuro

Resumo

A visão computacional e processamento de imagens têm sido utilizadas com sucesso em aplicações nas áreas de astronomia, medicina, análise de impressões digitais, sensoriamento remoto, multimídia, entretenimento, reconhecimento de assinaturas, automatização de tarefas repetitivas, entre outras. Devido a crescente automatização dos processos produtivos, busca-se transformar os sistemas computacionais para serem capazes de tornar automática a execução de tarefas complexas e/ou repetitivas que são realizadas por operadores humanos. A área da computação se depara com o desafio possibilitar com que o computador detecte as regiões de interesses da imagem a ser processada, extraíndo os dados desejados, e assim concluir com êxito o trabalho automatizado.. O trabalho aplicou um método exato utilizando a linguagem de programação *Python* e a biblioteca de visão computacional *Open-CV*. Após implementado, o método foi testado em um estudo de caso, correção de gabaritos, onde o mesmo, com êxito, detectou e extraiu de forma correta todas as informações das áreas de interesse em todos os 32 casos de teste, mantendo o tempo de execução próxima a 0.4 segundos por cada caso. Ao optar pelo método exato, tem-se um método eficiente, de baixo custo computacional e flexível, o que viabiliza este método em outros problemas reais.

Palavras-chave: processamento de imagem, região de interesse, visão computacional.

Abstract

Computer vision and image processing have been used successfully in applications in the areas of astronomy, medicine, fingerprint analysis, remote sensing, multimedia, entertainment, signature recognition, automation of repetitive tasks, among others. Due to the increasing automation of production processes, the aim is to transform computer systems to be able to make automatic the execution of complex and/or repetitive tasks that are performed by human operators. The computing area faces the challenge of enabling the computer to detect the regions of interest in the image to be processed, extracting the desired data, and thus successfully completing the automated work. The work applied an exact method using the language of programming *Python* and the computer vision library *Open-CV*. After implemented, the method was tested in a case study, correction of templates, where it successfully detected and correctly extracted all information from the areas of interest in all 32 test cases, maintaining the execution time close to 0.4 seconds for each case. By choosing the exact method, you have an efficient method, with low computational cost and flexible, which makes this method feasible in other real problems.

Keywords: image processing, region of interest, computer vision.

Lista de ilustrações

Figura 1 – Etapas de um sistema de Visão Computacional.	13
Figura 2 – Etapas de um sistema de Visão Computacional II.	14
Figura 3 – Identificação da área de interesse.	15
Figura 4 – Cobertura e ângulo de pulverização do jato utilizado no descarepador.	15
Figura 5 – Exemplos do resultado de detecção de objeto.	16
Figura 6 – Imagens utilizadas para realização do Template Matching.	16
Figura 7 – Detecção de formas geométricas por métodos exatos.	17
Figura 8 – Metodologia utilizada para implementação da proposta.	19
Figura 9 – Sequencia de padrão PADI.	21
Figura 10 – Filtros de Haar: Os pixels das regiões brancas são subtraídos dos pixels da região preta correspondente. Por exemplo, a máscara 1(a) apenas subtrai o valor de um pixel do seu vizinho á direita. A máscara 2(c) subtrai o pixel central dos pixels logo acima e abaixo. A máscara 3(a) subtrai um pixel central dos seus 8-vizinhos.	23
Figura 11 – Exemplos de atributos Haar calculadas sobre uma imagem.	23
Figura 12 – Passo a passo da estratégia do método YOLO.	24
Figura 13 – Estratégia do método <i>Templating Matching</i>	25
Figura 14 – Moeda sendo identificada usando do método <i>Templating Matching</i>	26
Figura 15 – Identificação de camisa e camiseta por métodos exatos.	27
Figura 16 – Ambiente do estudo de caso. Folha de resposta..	29
Figura 17 – Falsos positivos obtidos com o <i>Haar Cascade</i>	30
Figura 18 – Reconhecimento da área de interesse - Identificação do aluno.	31
Figura 19 – Reconhecimento da área de interesse- Gabarito.	31
Figura 20 – Etapas a serem seguidas para concretizar o objetivo proposto neste trabalho.	32
Figura 21 – Exemplo de caso de teste alterado para identificação da orientação correta.	33
Figura 22 – Reconhecimento correto das áreas de interesse- Gabarito e Identificação do Aluno.	34
Figura 23 – Extração correta da área de interesse - Gabarito.	35
Figura 24 – Extração correta das áreas de interesse - Identificação do Aluno.	35
Figura 25 – Processamento das áreas extraídas	36
Figura 26 – Provas antes do processamento	37
Figura 27 – Provas depois do processamento	38

Sumário

1	INTRODUÇÃO	10
1.1	Justificativa	11
1.2	Problema e sua importância	12
1.3	Objetivos	12
2	TRABALHOS RELACIONADOS	13
3	PROCEDIMENTOS METODOLÓGICOS	18
4	REFERENCIAL TEÓRICO	20
4.1	Processamento Digital de Imagens	20
4.2	Estratégias para detecção da área de interesse	22
4.3	<i>Haar Cascade</i>	22
4.4	<i>YOLO (You Only Look Once)</i>	23
4.5	<i>Templating Matching</i>	25
4.6	Métodos Exatos	26
5	ESPECIFICAÇÃO DO PROCESSO	28
5.1	Desenvolvimento	28
5.2	Etapas	32
6	CONCLUSÃO	39
	REFERÊNCIAS	41

1 Introdução

*“A resposta certa, não importa nada:
o essencial é que as perguntas estejam certas”.*
Mário Quintana

O principal recurso utilizado na percepção humana do mundo ao seu redor é a imagem, devido ao fato da visão ser o mais avançado entre cinco os sentidos (GONZALEZ; WOODS, 2010). Através dos nossos olhos capturamos informações por meio das imagens, essas informações transitam entre nossos neurônios retornando um significado. Algo complexo como, raciocínio lógico-matemático, percepção do mundo por meio de sons, tato e visão, que até então eram exclusivos do cérebro humano já fazem parte do cotidiano da computação. É uma evolução natural tentar dotar os computadores com essas e outras capacidades do nosso cérebro.

Devido a complexidade do cérebro humano, com sua grande capacidade e diferentes métodos de processamento da informação, o desenvolvimento de sistemas computacionais que imitam o comportamento humano na capacidade de interpretar e reagir ao que é observado ao nosso redor ainda não é possível pelas tecnologias atuais, porém quando for plenamente compreendido, serão inúmeras as possibilidades.(PEDRINI; SCHWARTZ, 2008).

Visão computacional defini-se como um campo da Ciência da Computação que se utiliza de técnicas e métodos para tornar possível que computadores ou máquinas capturem, identifiquem e processem imagens da mesma forma que humanos (WANGENHEIM; COMUNELLO, 2005). De maneira simples, a visão computacional empenha-se para resolver os mesmos problemas que humanos resolvem com seus próprios olhos e inteligência de forma rotineira. Facilmente o homem consegue reconhecer carros, árvores e cores de objetos, entretanto, o processamento de imagens pode demandar muito tempo quando realizado de forma manual, por um trabalhador.

A visão computacional e processamento de imagens segundo Gonzalez e Woods (2010) têm sido utilizadas com sucesso em muitas aplicações relevantes. Esses sistemas de visão computacional tem sido utilizados juntamente com técnicas de processamento de imagem, para as mais variadas aplicações, uma vez que um problema da visão computacional é sempre distinto, sendo necessária uma boa análise do problema a ser tratado para escolher qual a técnica mais indicada à aplicação. Devido a crescente automatização dos processos produtivos, busca-se transformar os sistemas computacionais capazes de automatizar a execução de tarefas complexas (que muitas vezes são repetitivas) que até então são realizadas por operadores humanos(ORTH, 1998).

1.1 Justificativa

O avanço nesses estudos aliado as novas tecnologias e novos algoritmos tem permitido a criação de varias aplicações que utilizam de Processamento Digital de Imagem (PDI). Além das aplicações já mencionadas, alguns dos domínios de conhecimento que envolvem a utilização de PDI na resolução de problemas incluem (GONZALEZ; WOODS, 2010):

- Áreas da astronomia
- Medicina
- Análise de impressões digitais;
- Sensoriamento remoto;
- Multimídia;
- Entretenimento;
- Reconhecimento de assinaturas;
- Biologia;
- Microscopia;
- Robótica móvel;
- Automatização de correção de provas.

Das áreas citadas acima e em outras onde se faz o uso de PDI, o desafio e também o ponto em comum, é que elas necessitam de rastrear e extrair a área de interesse desejada por suas aplicações. Define-se como rastreamento o processo de determinação da posição da(s) área(s) de interesse ao longo de uma imagem (GRACIANO, 2007).

Uma das grandes dificuldades em visão computacional é a identificação das regiões de interesse. Em processamento de imagens, pode-se conceituar região de interesse (*ROI – Region Of Interest ou AOI - Area Of Interest*) como sendo uma região definida pelo operador, ou automaticamente a partir de parâmetros obtidos da própria imagem sob análise. A extração da região de interesse tem a finalidade de evitar buscas em regiões da imagem que não possuem informação relevante, otimizando o processamento da área extraída. A extração destas áreas em seu cenário natural apresenta questões desafiadoras. Vários fatores influenciam tais como a qualidade da imagem, o ângulo de captura, iluminação excessiva, ou a falta dela. Variáveis como essas faz com que extração corretas das áreas de interesse em uma determinada imagem digital seja mais complicado.

O trabalho em questão propõe utilizar um método para extrair corretamente a área interesse em uma imagem digital. Para facilitar o entendimento pode-se tomar o seguinte cenário como exemplo: Folha de resposta de uma prova de múltiplas questões. A área de interesse deste campo seria o gabarito e também as informações que identifique quem realizou a prova,

com o intuito de utilizar essas informações para automatizar correção de provas. Nesse tipo de cenário o principal desafio é identificar a região de interesse da imagem, ignorando as outras áreas que não possuem informações relevantes, para que com isso apenas as áreas extraídas sejam as que tenham as informações pertinentes a aplicação, fazendo com que os dados ali presentes nas áreas extraídas esteja de acordo com o esperado, diminuindo assim a margem de erro das correções.

No cenário apresentado a utilização de métodos exatos é uma proposta interessante, tendo em vista que estes classificam objeto de interesse baseado em suas características peculiares, tais como: comprimento, perímetro, raio, cor, altura, rotação, etc (SILVA, 2014). Como neste campo de prova os objetos a serem detectados (gabarito, área de identificação dos dados da prova), possuem características bem definidas e constantes, o uso de métodos exatos pode ser ideal para manter a simplicidade, flexibilidade e também o baixo custo computacional, tendo em vista que métodos exatos baseiam-se em operações matemáticas simples sobre as imagens.

1.2 Problema e sua importância

Efetuar a extração correta da região de interesse ignorando informações desnecessárias não é uma tarefa simples, é uma das dificuldades da área, onde no processamento da imagem é utilizado técnicas para a correção da mesma, para aumentar ou diminuir o brilho ou contraste, melhorar o foco e outros ajustes, todos com o intuito de deixar a imagem ajustada para que o sistema descarte as informações desnecessárias. As diversas variáveis presente na aquisição da imagem pode influenciar diretamente no resultado esperado. Motivado por esta gama de aplicações e recursos disponíveis relacionados a PDI, este trabalho propôs aplicar e avaliar um método para a extração da área de interesse em imagens digitais.

1.3 Objetivos

Este trabalho tem como principal objetivo especificar, implementar e avaliar um método exato para detecção de área de interesse em imagem digital.

Também, objetivam-se mais especificamente:

1. Detectar o contorno e posição da área de interesse;
2. Extrair a identificação e o gabarito da prova;
3. Avaliar o desempenho do método em relação a correção dos dados extraídos.

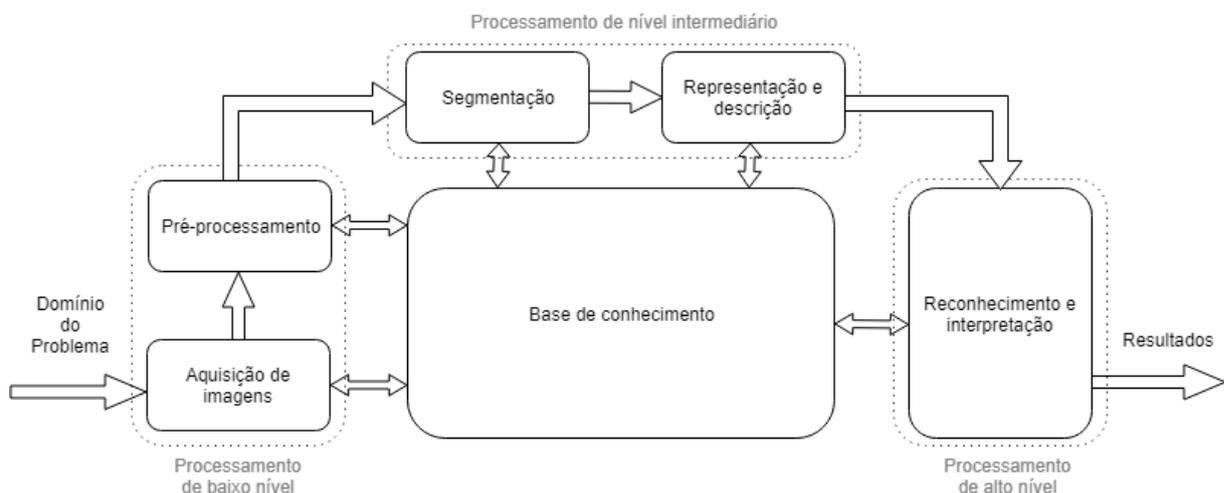
2 Trabalhos Relacionados

*“Há apenas um bem, o saber;
e apenas um mal, a ignorância”.*
Sócrates

Nos materiais relacionados a área de PDI disponíveis na internet foram encontrados trabalhos com propostas semelhantes ao que está sendo desenvolvido, onde dado um cenário é necessário localizar a área de interesse para extrair as informações relevantes para solucionar o problema proposto. Como já foi dito anteriormente, o campo da computação que é responsável por tornar possível que máquinas e computadores vejam é nominado como Visão Computacional.

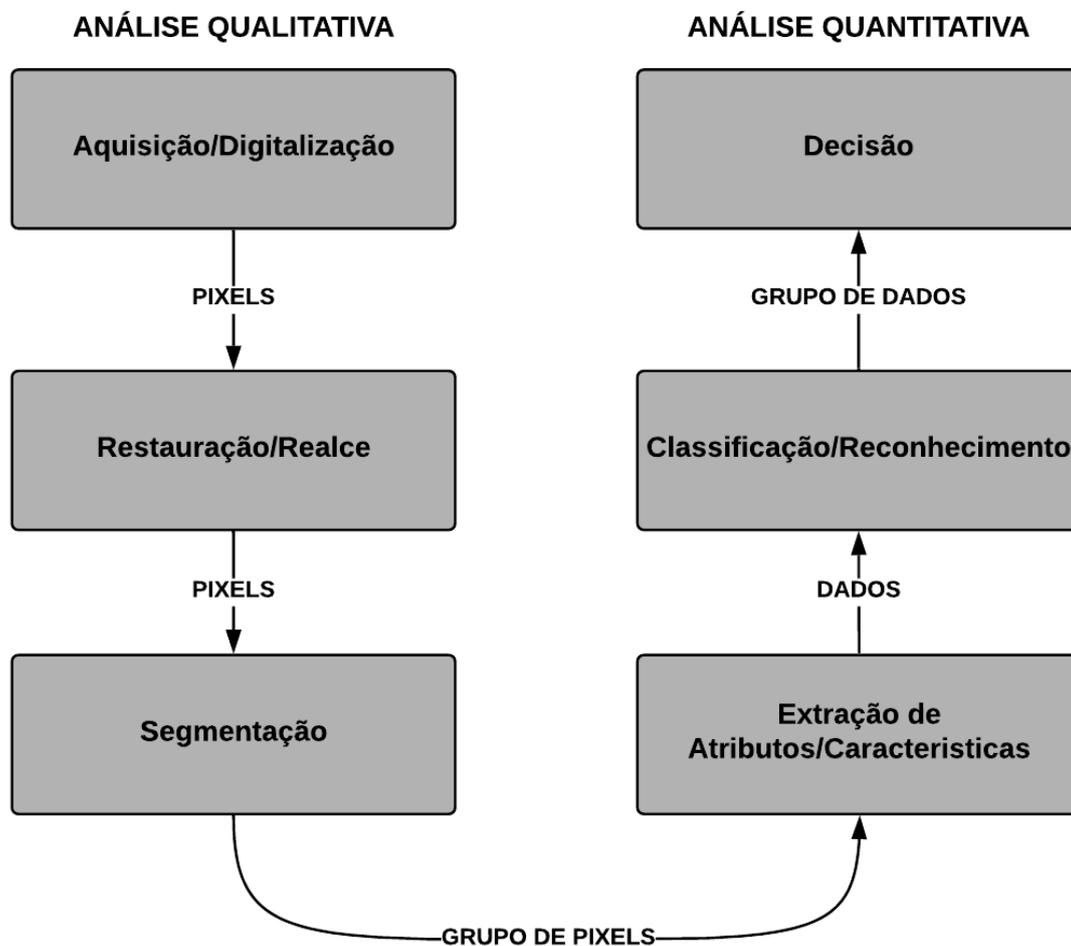
Para que o computador seja capaz de interpretar e processar imagens, a maioria dos trabalhos elencados quando necessário uma base de conhecimento, seguem as etapas propostas por (GONZALEZ; WOODS, 2010) que podem ser observadas na figura 1 ou as etapas proposta por (CONCI; AZEVEDO; LETA, 2009) como mostra a figura 2 quando não se faz necessária uma base de conhecimento. Logo em seguida das figuras serão listados alguns trabalhos que de alguma forma relacionam-se com o tema do trabalho proposto.

Figura 1 – Etapas de um sistema de Visão Computacional.



Fonte: (GONZALEZ; WOODS, 2010)

Figura 2 – Etapas de um sistema de Visão Computacional II.

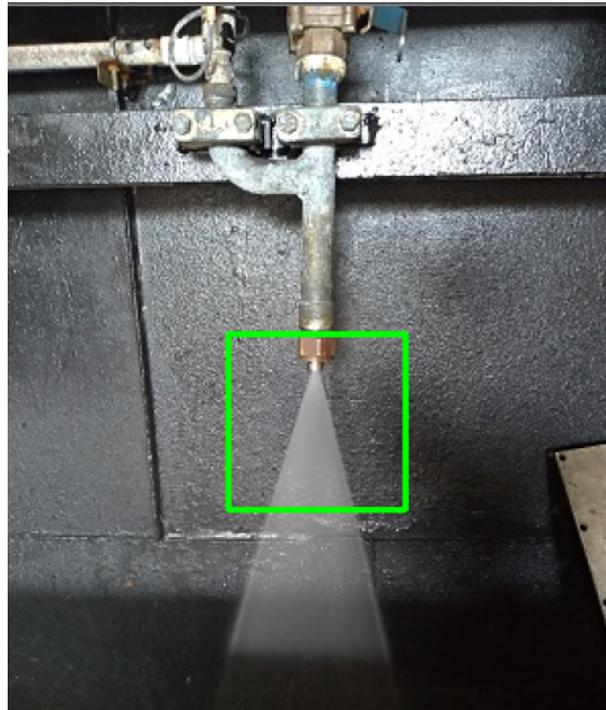


Fonte: Adaptado (CONCI; AZEVEDO; LETA, 2009)

Em Dutra (2019), a autora propõe detectar a área de interesse em imagem digital através de um método de aprendizagem supervisionada. Para a extração da área de interesse ela utilizou o algoritmo *Haar Cascade*. Este método necessita de um treinamento prévio onde lhe é passado a área de interesse para treinamento do classificador, o classificador armazenará as características desta área e posteriormente, na melhor situação possível, torna-se capaz de identifica-lá dentro de um cenário, na figura 3 vemos essa área identificada.

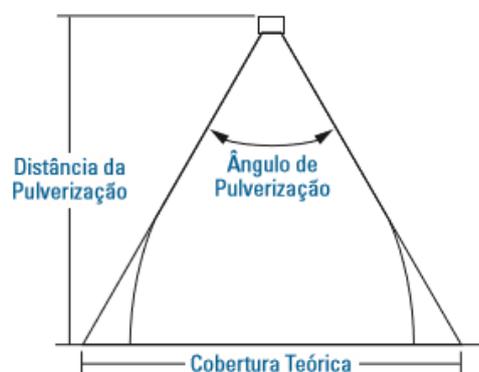
A autora propôs esse método para avaliar o formato de um jato em relação a uma escala de especificações técnicas pré definidas, ela tinha como objetivo achar o ângulo de pulverização e a cobertura teórica do jato como mostra o esquema ilustrado na figura 4, dada a complexidade de seu objetivo, com o uso do *Haar Cascade* ele obteve um acerto de 80,77% do total de imagem submetidas ao teste.

Figura 3 – Identificação da área de interesse.



Fonte:(DUTRA, 2019)

Figura 4 – Cobertura e ângulo de pulverização do jato utilizado no descarepador.

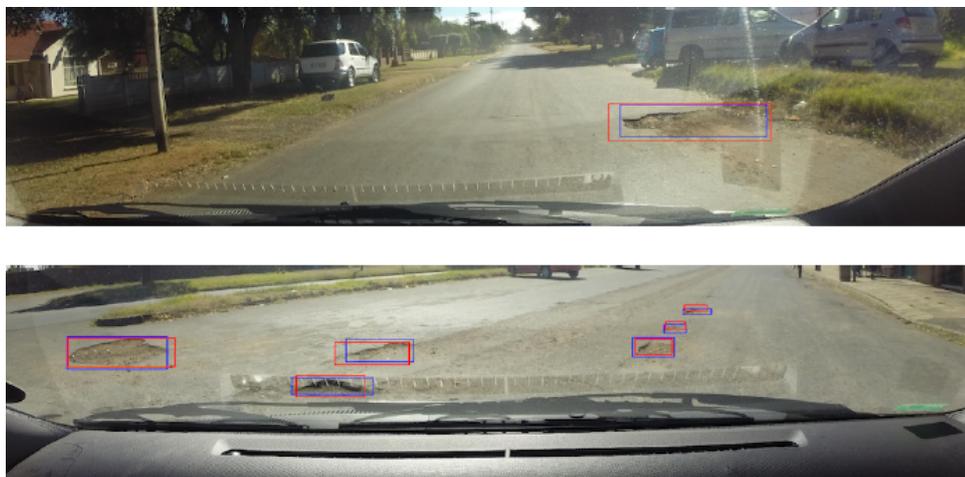


Fonte: Spraying Systems Co

Marcos (2019) propôs aplicar a visão computacional no auxílio de levantamento de defeitos em pavimento, o principal defeito a ser estudado é o buraco. Neste trabalho as fotos foram obtidas por um carro em movimento, além desta dificuldade em ter de capturar as fotos em movimentos, o autor também necessita de fazer correções nas mesmas, pois elas sofrem com algumas variáveis externas tais como iluminação e brilho, providas pelo sol ou pela lua. O autor fez o uso de redes neurais focado nas topologias de ResNet (*Residual Neural Network*) (SZEGEDY et al., 2016) e YOLO (*You Only Look Once*) (REDMON et al., 2016), para alcançar seu objetivo. ResNet e YOLO também são métodos para detecção da área de interesse. Ao

fim dos testes ele optou pelo uso do YOLO e também obteve sucesso na identificação dos buracos como mostra figura 5.

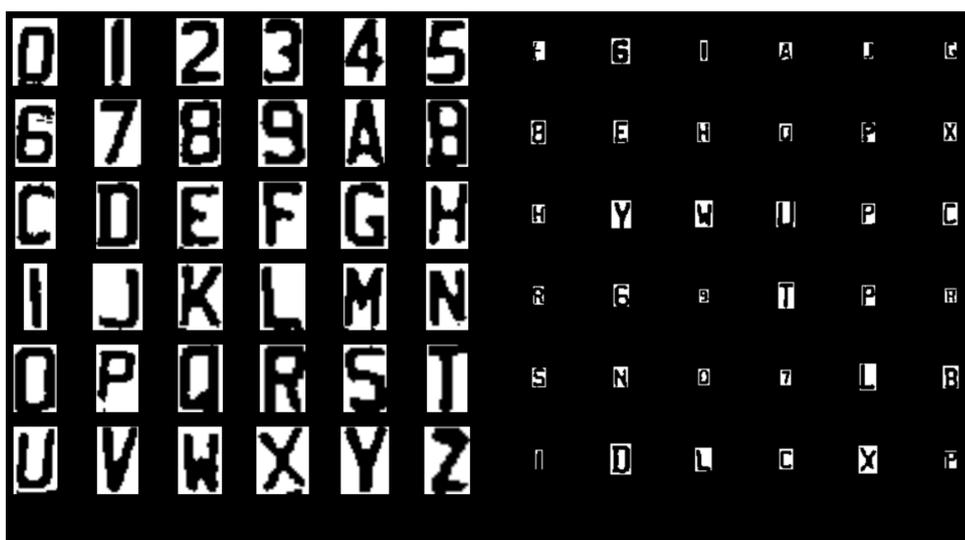
Figura 5 – Exemplos do resultado de detecção de objeto.



Fonte: (HOLLERWEGER, 2019)

Luiz (2018) realiza um estudo de caso em reconhecimento automático de placas veiculares usando linguagem de programação Python e Redes Neurais Artificiais (RNA). Para efetuar a extração da área de interesses uma das técnicas utilizadas pelo autor foi o uso do método chamado *Template Matching* presente na biblioteca OpenCV 2020. Os resultados obtidos para o estudo de caso foram inferiores aos obtidos com as técnicas mais modernas encontradas na literatura. Entretanto, foram úteis como demonstração para o problema proposto como mostra a figura 6.

Figura 6 – Imagens utilizadas para realização do Template Matching.

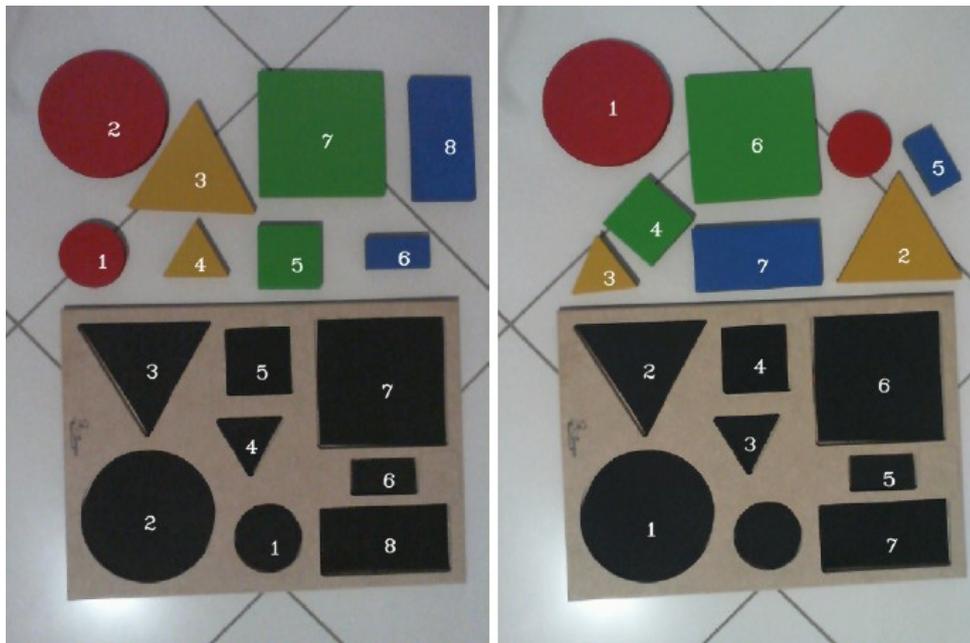


Fonte: (ALMEIDAR, 2018)

Pedro (2015) faz a utilização de métodos exatos, através de características bem definidas como cor e forma ele consegue identificar a área de interesse em tempo real em uma ima-

gem digital ou sequência de imagens utilizando apenas de métodos exatos. O autor optou por esse método devido a simplicidade, eficiência, flexibilidade e baixo custo de processamento. Nos testes feitos pelo o autor o software desenvolvido com a implementação dos métodos obteve taxa de acerto superior a 90% e tempo de resposta inferior a 30 milissegundos. Na figura 9 vemos o software em funcionamento na identificação da área de interesse.

Figura 7 – Detecção de formas geométricas por métodos exatos.



Fonte: (PINTO, 2015)

Como pode ser observado nos trabalhos citados acima todos tinham a necessidade de extrair as áreas de interesse de uma imagem digital em seus cenários, cada trabalho possuía sua particularidade, mas mesmo assim não lhe faltaram métodos para serem utilizados na resolução do problema. Foram citados especificamente os trabalhos acima para demonstrar os diversos métodos que podem ser utilizadas para extração da área de interesse. De acordo com o problema ao qual quer resolver basta escolher um método que melhor sobressai perante o cenário e ajustá-lo.

3 Procedimentos metodológicos

“Um bom começo é a metade”.
Aristóteles

Para desenvolvimento trabalho foi necessário assimilar os principais conceitos e técnicas de visão computacional e de processamento digital de imagens. Para o estudo e o entendimento dos conceitos foi necessário recorrer a livros, teses de mestrados, trabalhos de conclusão de cursos e artigos relacionados ao tema proposto.

Para realização do trabalho foi utilizado a biblioteca OpenCV, biblioteca que possui recursos que auxiliam nas aplicações pertinentes a visão computacional, mais sobre a mesma será comentado no capítulo 4. Podendo ser programada em diversas linguagens, optaremos pela linguagem Python, para isso utilizaremos a plataforma Pycharm e também a plataforma Anaconda para gerenciamento da biblioteca OpenCV e as demais bibliotecas que serão utilizadas, também serão explicadas mais sobre as plataformas posteriormente.

Para efetuar a implementação do método foi utilizado um notebook com as seguintes especificações.

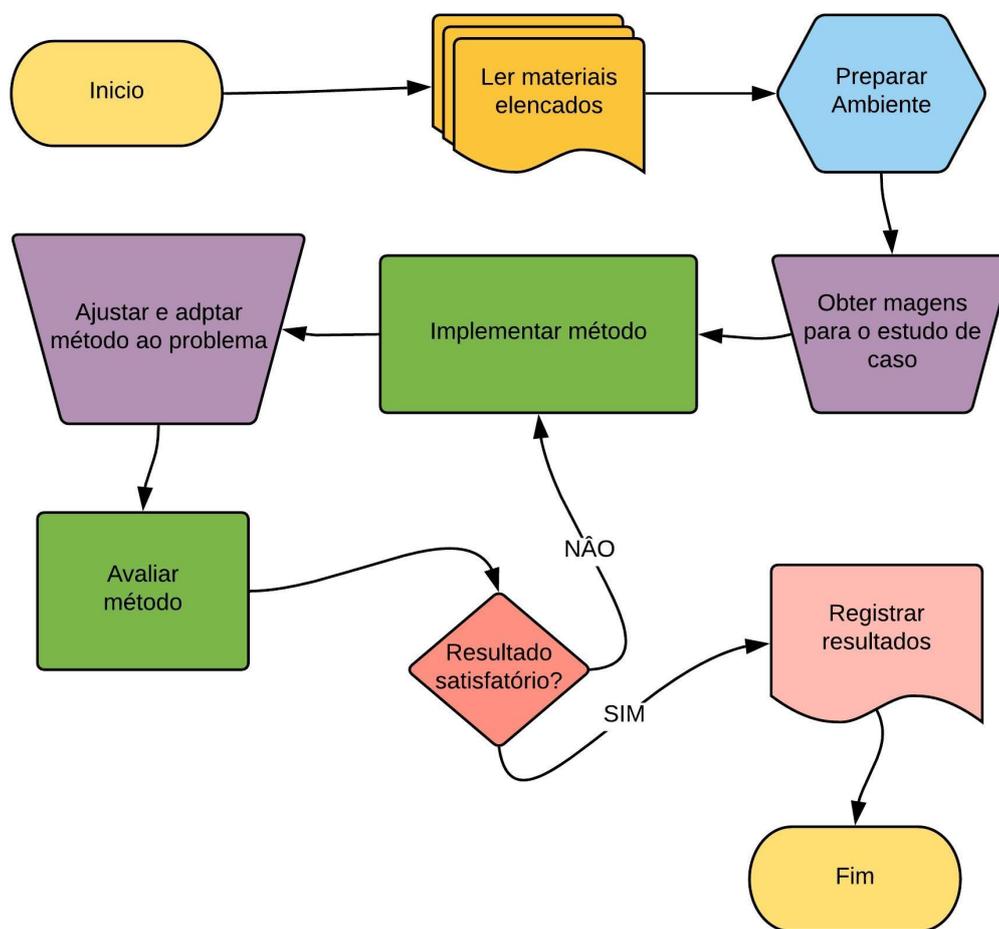
- Processador Intel Core i3 3110M CPU 2.40Ghz;
- Memória RAM 8GB e HD 1TB;
- Sistema Operacional Windows 10 x64.
- Dimensões da Imagem: 4961 x 7016 ;
- Altura 4961 pixels;
- Largura 7016 pixels;
- Resolução 600dpi.

Os procedimentos metodológicos são organizados nas seguintes etapas:

1. Reunir e estudar trabalhos relacionados com o tema proposto para que se faça uma organização de informações sobre os principais métodos e técnicas de PDI para efetuar a detecção da área de interesse.
2. Obter as imagens para teste ;
3. Escolher e especificar o método que será utilizado no processo;
4. Avaliar o desempenho do método escolhido em um estudo de caso;

No figura 8 mostrada logo abaixo é ilustrado os passos para melhor entendimento das atividades realizadas para a conclusão do problema proposto nessa trabalho.

Figura 8 – Metodologia utilizada para implementação da proposta.



Fonte: Elaborada pelo autor

4 Referencial Teórico

“Um bom começo é a metade”.
Aristóteles

Nas próximas sessões são apresentados alguns conceitos necessários para o desenvolvimento deste projeto.

4.1 Processamento Digital de Imagens

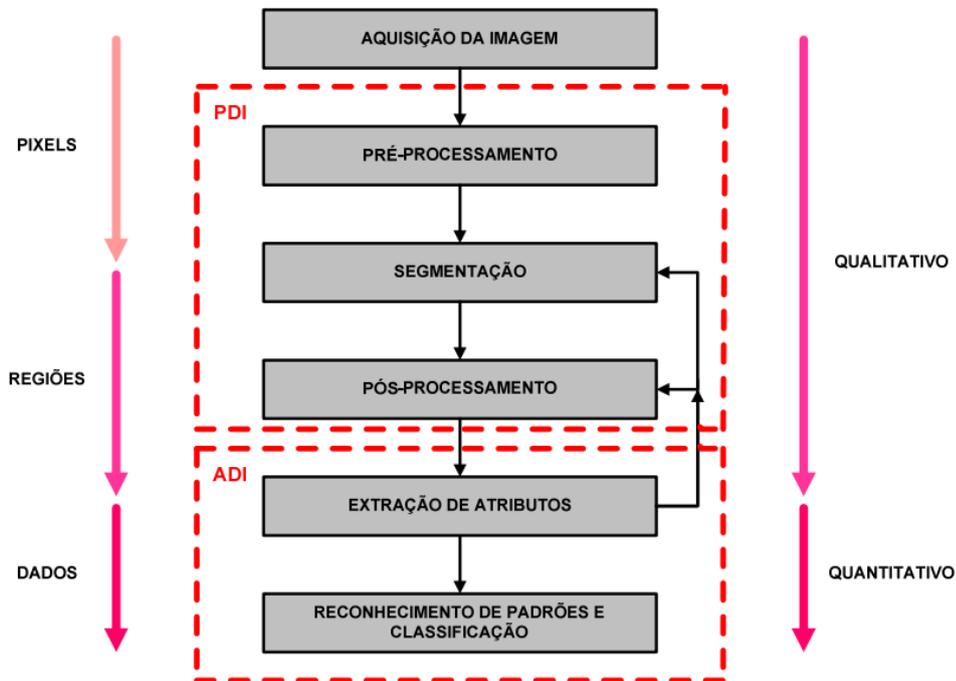
Ao longo dos anos a área de processamento digital de imagens vem evoluindo continuamente, também ocorreu um aumento significativo de estudos envolvendo morfologia matemática, redes neurais, processamento de imagens coloridas, compressão de imagens, reconhecimento de imagens e sistemas de análise de imagens baseado em conhecimento (GONZALEZ; WOODS, 2010).

O processamento de imagens é realizado em etapas bem definidas, onde elas compartilham informações entre si, com intuito de produzir o resultado esperado. Sendo assim o resultado de cada etapa influencia diretamente nas etapas seguintes. As etapas de processamento digital de imagem se dividem em: aquisição da imagem, filtros de imagem, segmentação, reconhecimento e interpretação (GONZALEZ; WOODS, 2010).

O processamento tem como função realizar as correções de defeito da imagem, a fim de facilitar as análises quantitativas, pode também realçar as regiões de interesse na imagem ou facilitar a extração e o tratamento de dados quantitativos ou a visualização das imagens, feitas pelo próprio computador (GOMES; VELHO, 2002).

Como pode ser visto nas citações acima, as funções do processamento de imagem pode ser vista como um conjunto de duas técnicas, Processamento Digital de Imagem (PDI), que é a responsável por corrigir a imagem para as análises seguintes e também a Análise Digital de Imagens (ADI), que é responsável pela análise quantitativa a partir da qual as regiões e objetos de interesse identificados na imagem serão medidos. Como pode ser observado a baixo, a figura 9 separa bem as etapas. O nome PADI (Processamento e Análise Digital de Imagens) será sempre referenciado quando envolver as duas técnicas.

Figura 9 – Sequencia de padrão PADI.



Fonte: (GOMES; VELHO, 2002).

De acordo com a figura 9, no início do processamento de imagens se dá a aquisição da imagem digital, em seguida ocorre o pré-processamento da imagem obtida, esta etapa é responsável por corrigir alguns defeitos da imagem para que as etapas subsequentes tenham êxito. Ao fim das duas primeiras etapas surge a segmentação, que fica com o trabalho de separar a imagem em partes ou objetos de interesse. Em alguns casos a segmentação é feita de forma errônea ou equivocada, e para corrigir esses casos entra a etapa de pós-processamento.

Com as 4 primeiras etapas mencionadas finalizadas, etapas qualitativas, onde o foco maior são os pixels, e as regiões da imagem, etapas de PDI, a imagem já está pronta para entrar nas etapas de ADI, a parte quantitativa deste processo, onde entra a extração de atributos e o reconhecimento de padrões e classificação. Ao finalizar as etapas de PDI a imagem está pronta para fornecer dados através da etapa de extração de atributos. Nesta etapa características dos objetos são extraídas, o que resulta na separação dos objetos por classes.

A etapa final desta parte do processamento de imagens é dada ao reconhecimento de padrões e classificação. Baseado nas informações que lhe foram passadas a descrição do objeto será atribuída através do reconhecimento. A classificação confirma um significado para cada conjunto de objetos que foram identificados. Esta sequência padrão é uma generalização o que a faz não ser imutável, o fluxograma de PADI é totalmente flexível, dependendo da complexidade do problema envolvido as etapas podem ser repetidas ou até mesmo retiradas de modo a aprimorar o processo.

De forma bem resumida foi citado as etapas necessárias para o processamento de imagem, o trabalho em questão focará no início do processo, entre as etapas de aquisição de

imagem e pré processamento, onde é reconhecido a área de interesse que será manipulada pelas etapas posteriores, nessa parte que se encontra o método para extração. Um estudo mais aprofundado de PADI pode ser encontrado na tese de mestrado escrita e defendida por (AUGUSTO, 2012), onde o mesmo detalhadamente explica sobre todas as etapas mencionadas acima e também o trabalho de conclusão de (DUTRA, 2019), onde a autora detalha de forma sucinta as etapas de pré-processamento e segmentação dando exemplo das diversas técnicas que podem ajudar nessas etapas.

4.2 Estratégias para detecção da área de interesse

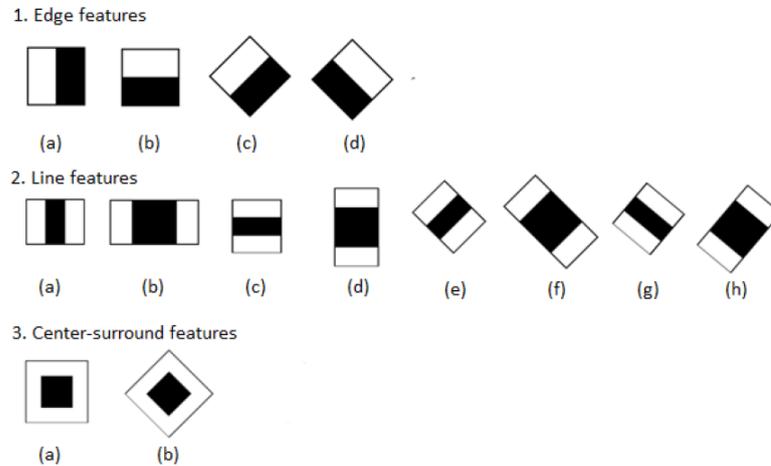
4.3 *Haar Cascade*

Algoritmo utilizado no trabalho de Samyla (2019), funciona de modo que uma árvore de decisão é treinada afim de que cada nível analise um conjunto de atributos diferentes e avalie se esses atributos pertencem ou não ao objeto de interesse, é utilizado detectores baseados em *cascade* (cascata) por isso o nome. Cada nível é chamado de estágio, e cada estágio é composto por um ou mais classificadores. Para que seja construído um classificador é necessário imagens positivas (imagens que possuem a região de interesse) e imagens negativas (imagens que não possuem a região de interesse) para se diferenciar as situações. Para que a região de interesse seja corretamente localizada é necessário se preocupar em treinar o detector.

Para gerar um bom classificador pode demandar muito tempo em alguns casos. Para efetuar o treinamento do *Haar* que foi o tipo de *feature* escolhido para o desenvolvimento do trabalho os atributos são extraídos subtraindo diferentes pixels da imagem de acordo com as “máscaras” mostradas na figura 10. O conjunto prévio de características utilizadas, a fim de varrermos as amostras em busca das características críticas, são os apresentados na figura abaixo.

O *Haar Cascade*, que pode ser encontrado na biblioteca *OpenCV*, é um algoritmo de aprendizado baseado em *AdaBoost*, que seleciona um pequeno número de características visuais críticas de uma determinada imagem e as utiliza para a etapa de detecção e classificação de objetos. A figura 11 mostra um exemplo da janela deslizante sobre uma imagem. Os Haar utilizam os filtros mostrados na figura 10 para percorrer toda a imagem positiva procurando o padrão que se encaixe como mostra a figura abaixo.

Figura 10 – Filtros de Haar: Os pixels das regiões brancas são subtraídos dos pixels da região preta correspondente. Por exemplo, a máscara 1(a) apenas subtrai o valor de um pixel do seu vizinho à direita. A máscara 2(c) subtrai o pixel central dos pixels logo acima e abaixo. A máscara 3(a) subtrai um pixel central dos seus 8-vizinhos.



Fonte: (GRANATYR, 2020).

Figura 11 – Exemplos de atributos Haar calculadas sobre uma imagem.



Fonte: (GRANATYR, 2020).

4.4 YOLO (*You Only Look Once*)

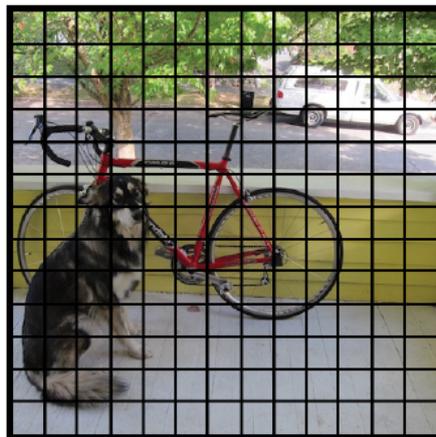
O Yolo (REDMON et al., 2016) apresenta uma estratégia que possibilita a detecção da área de interesse em tempo real analisando a imagem apenas uma vez, por isso o nome, o que lhe permite apresentar um desempenho muito superior a de outros métodos. Focado apenas na detecção o Yolo possui procedimentos de predição dividido em 4 passos simples de serem compreendido, como pode ser observado a seguir:

1. A imagem é dividida em *grid* de 13 por 13 células, como pode ser observado na 12.
2. Com a imagem dividida em 169 células começa a predição, cada uma das células fica responsável pela predição de 5 caixas de seleção. As caixas podem ter formatos e tamanhos variados e correspondem à área onde se acredita que é a área de interesse.

Os *escores* de confiança para cada caixa também são obtidos nesse passo. Ainda não há informação sobre a área de interesse e sua classificação, porém as caixas podem (ou não) conter essas áreas. As caixas com moldura mais espessas representam uma caixa com *escore* maior e as com cores diferentes representam classes diferentes. Esse segundo passo pode ser observado na figura 12.b

3. 169 células com 5 caixas cada, o que nos dá um total de 845 caixas de seleção, e para cada uma das caixas a rede faz a predição da classe. O score de confiança de que a caixa contenha um objeto é combinado com um novo *escore* que representa a probabilidade de acerto no objeto classificado. Esse passo está ilustrado na figura 12.c
4. Neste quarto e último passo as caixas com *escore* muito baixo são eliminadas e também serão eliminadas as caixas que estiverem abaixo do limiar escolhido. Após este processo são obtidas apenas as caixas com maior chance de conter o objeto e que esse objeto possa ser reconhecido pela rede. O resultado está ilustrado na figura 12.d

Figura 12 – Passo a passo da estratégia do método YOLO.



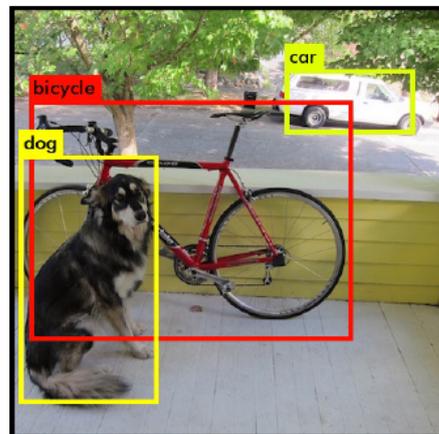
(a) Passo 1.



(b) Passo 2.



(c) Passo 3.



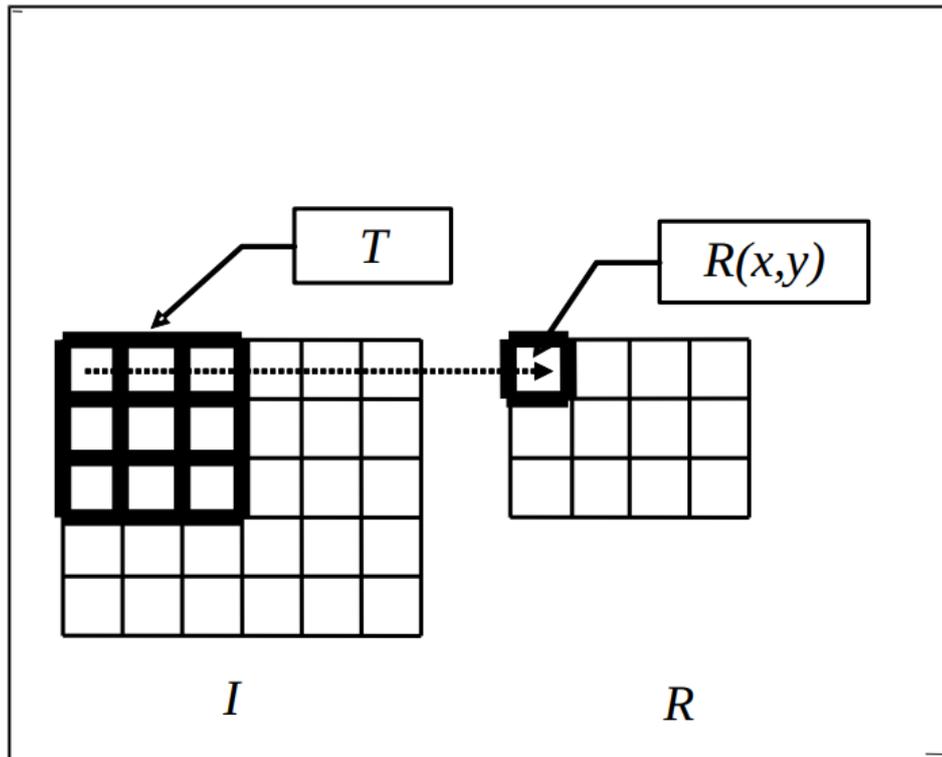
(d) Passo 4.

Fonte: (HOLLERWEGER, 2019).

4.5 *Templating Matching*

Presente na biblioteca OPENCV um método simples de ser implementado, flexível, eficiente e de baixo custo de processamento. Como pode ser observado na figura 13 *Templating Matching* percorre a imagem I comparando template T com cada posição (x,y) de I . O resultado da comparação é armazenado em $R(x,y)$. A saída R é menor que a entrada I (USP, 2020).

Figura 13 – Estratégia do método *Templating Matching*.



Fonte: (USP, 2020).

O método realiza uma convolução da máscara sobre toda a imagem e retorna a posição onde existe a menor diferença e o valor de tal diferença, semelhante ao cálculo da diferença ponto a ponto entre as imagens. A máscara possui as mesmas dimensões da imagem e a saída depende do tamanho das entradas, se a imagem de entrada for do tamanho $(W \times H)$ e a imagem do modelo for do tamanho $(w \times h)$, a imagem de saída terá o tamanho de $(W-w + 1, H-h + 1)$ (OPENCV, 2020).

Para estimar a diferença entre a máscara e a imagem de teste durante a convolução, existem diversas métricas. A técnica mais comum é a correlação e suas variações e a diferença quadrática média. O método torna-se mais complexo ao exigir invariância de mudança de brilho, contraste, rotação, mudança de escala e mudança de ponto de vista, o que pode impossibilitar um bom reconhecimento da área de interesse em alguns casos. Na figura 14 pode-se observar o método em funcionamento. Onde lhe foi passado um modelo onde o objetivo era identificar as moedas presentes no cenário, e como pode ser visto, o objetivo foi alcançado.

Figura 14 – Moeda sendo identificada usando do método *Templating Matching*.

Fonte: (USP, 2020).

4.6 Métodos Exatos

Quando a área de interesse possui características bem definidas e constantes, um método exato é uma boa alternativa. Além de ser recomendado em casos como estes, estes métodos podem servir de auxílio para outros métodos mais complexos. Eles baseiam-se em características particulares da área de interesse tais como: raio, altura, perímetro, comprimento, rotação e etc. As características podem ser combinadas para classificar a área de interesse com mais precisão, identificando corretamente a qual classe pertence.

Estes métodos baseiam-se em operações matemáticas simples sobre as imagens, que apresenta um baixo custo de processamento, o que os tornam atraente para serem utilizados em algumas determinadas situações evitando o uso desnecessário de métodos mais complexos. Para que os métodos sejam efetivos em sua detecção da área de interesse é necessário que o ambiente seja previamente conhecido, sua utilização deve ocorrer em ambientes controlados pois as características utilizadas para a detecção são relativas e variam de acordo com alguns aspectos, como posicionamento, distância e ângulo em relação a câmera, uma sugestão é que a câmera seja fixa.

Como pode ser observado na figura 15, utilizou-se métodos exatos para classificar algumas roupas, entre camisas e camisetas, e como pode ser observado o resultado foi satisfatório.

Figura 15 – Identificação de camisa e camiseta por métodos exatos.



Fonte: (PINTO, 2015).

5 Especificação do Processo

“As palavras fogem quando precisamos delas e sobram quando não pretendemos usá-las.”
Carlos Drummond de Andrade

5.1 Desenvolvimento

Como foi mencionado no Capítulo 2, existem diversos métodos para efetuar a extração da área de interesse em uma imagem. Porém cada um deles se adapta melhor a um determinado cenário. Em um dos trabalhos citados no capítulo, foi utilizado o *Haar Cascade* para localizar o bico dos jatos pulverizadores. Com a aplicação deste método a autora obteve uma taxa de acerto superior a 80%, ou seja, ele localizou com sucesso a área de interesse na maioria dos casos. Neste caso *Haar Cascade* conseguiu extrair corretamente a área de interesse, entretanto ao ser implementado no estudo de caso proposto o mesmo não teve o mesmo desempenho. Aplicou-se esse método no estudo de caso proposto, uma folha de resposta de uma prova de múltipla escolha como mostra a figura abaixo.

Como pode ser observado na figura abaixo, as áreas de interesse encontram-se isoladas, as áreas a serem extraídas são: a identificação do aluno que realizou a prova e o gabarito, onde o mesmo marcou suas respostas.

Figura 16 – Ambiente do estudo de caso. Folha de resposta..



**CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS VII - UNIDADE TIMÓTEO**

AVALIAÇÃO SOMATIVA

Curso: Desenvolvimento de Sistemas
Disciplina: Matemática
Professor: ABC
Aluno: XYZ

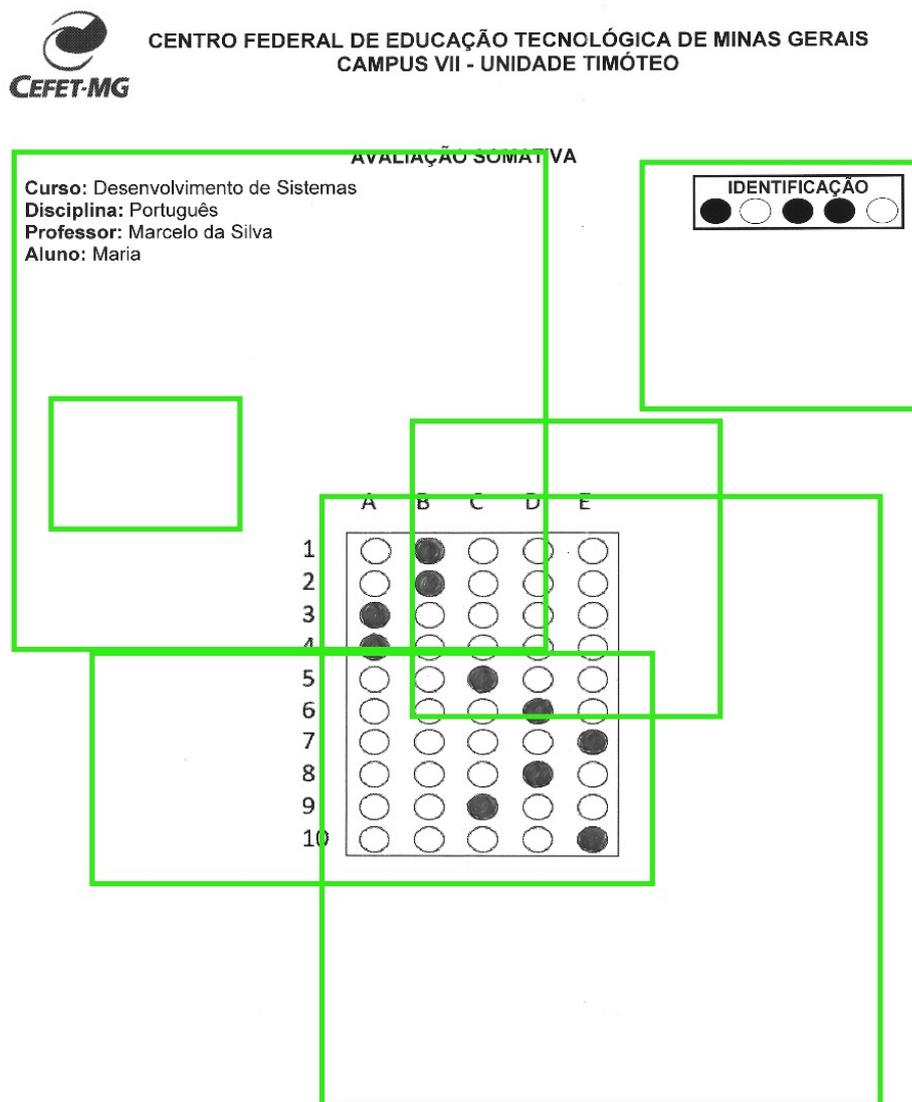


	A	B	C	D	E
1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
2	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
4	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
8	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
9	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
10	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fonte: Elaborada pelo autor.

Utilizando o *Haar Cascade* observou-se que na maioria dos casos a área de interesse não era devidamente localizada, outro ponto negativo do método no ambiente proposto era o grande numero de falso positivo (área que o método julga estar identificando corretamente) como mostra a imagem abaixo.

Figura 17 – Falsos positivos obtidos com o *Haar Cascade*.

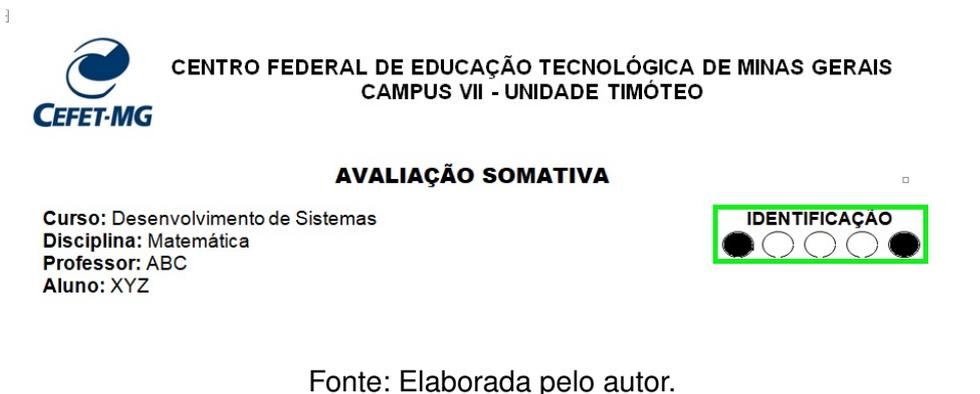


Fonte: Elaborada pelo autor.

A utilização deste método não obteve resultados satisfatório, sendo assim, optou-se por uma nova abordagem, utilizar métodos exatos, sendo que o mesmo demanda menos recurso computacionais e tende a ser mais rápido que os outros algoritmos, tendo em vista

também que o ambiente não é algo que necessita de muita complexidade. Como pode ser observado nas duas figuras abaixo, é mostrado uma simulação de como o algoritmo deveria se portar diante do corpo de prova identificando corretamente a área de IDENTIFICAÇÃO do aluno e o GABARITO, onde o aluno marcou suas respostas. As bordas da área de interesse são demarcadas com a cor verde, informando que todo aquele contorno limita a área de interesse desejada.

Figura 18 – Reconhecimento da área de interesse - Identificação do aluno.



Fonte: Elaborada pelo autor.

Figura 19 – Reconhecimento da área de interesse- Gabarito.

	A	B	C	D	E
1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
2	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>
4	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
5	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
6	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
8	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
9	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
10	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

Fonte: Elaborada pelo autor

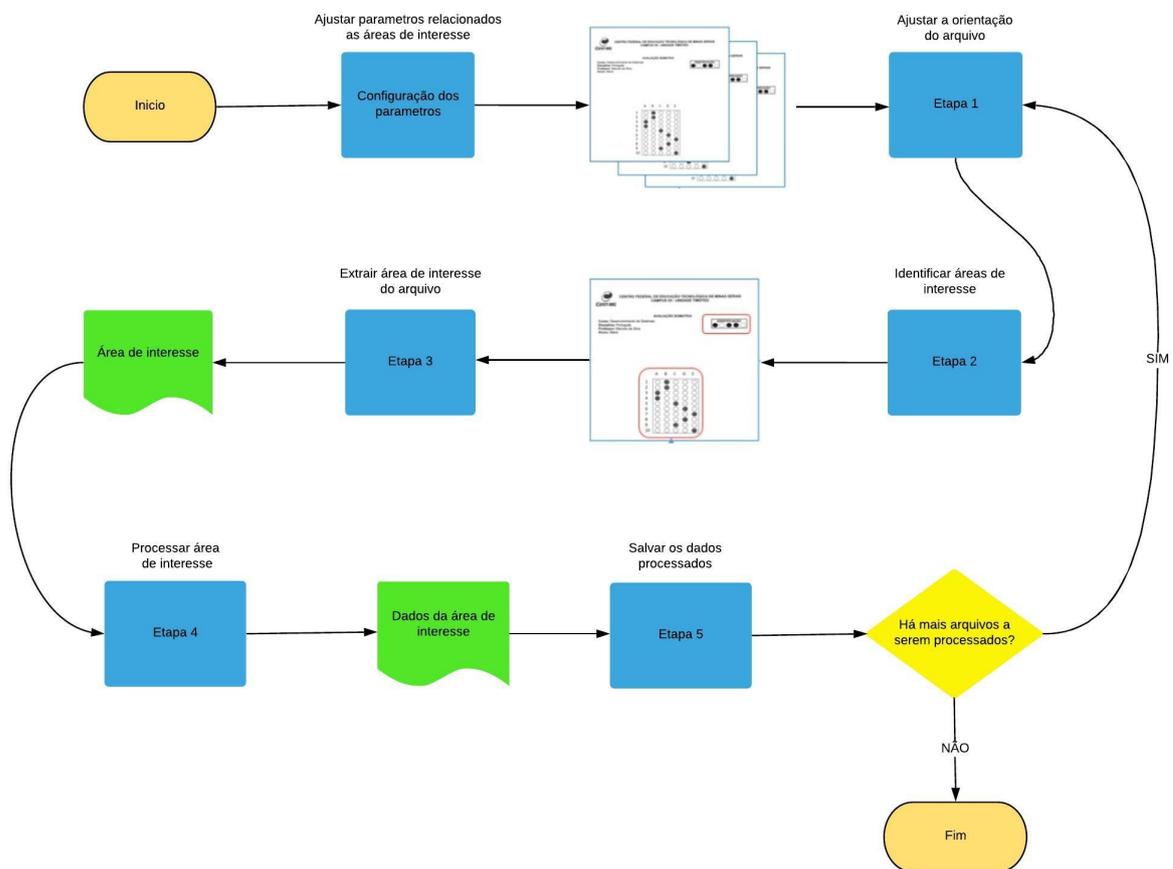
Após alguns estudos elegeu-se um método presente na biblioteca *OpenCV*, o *findContours* que busca os contornos na imagem através de operações matemáticas. Contornos são uma ferramenta útil para a análise de formas e detecção e reconhecimento de objetos (OPENCV, 2020). Através de alguns parâmetros como quantos lados tem e tamanho total

do contorno podemos reconhecer diversas figuras geométricas. Como as áreas de interesse neste estudo de caso possuem características bem definidas e constantes, ambiente ideal para métodos exatos, julgou-se ser uma boa escolha. Uma outra vantagem do método escolhido é que ele demanda de pouco poder computacional, é mais rápido e eficiente em ambientes como esse.

5.2 Etapas

Após a escolha do método foi necessário traçar as metas do processo, sendo assim foi definido quais os passos serão seguidos até a conclusão do objetivo proposto nesse trabalho, na figura 20 podemos ver quais passos foram definidos e especificados.

Figura 20 – Etapas a serem seguidas para concretizar o objetivo proposto neste trabalho.



Fonte: Elaborada pelo autor

Como é mostrado na figura acima, o processo é iniciado com a configuração dos parâmetros como pasta a ser lida, ajuste de imagem, o redimensionamento, entre outras correções. É informado a quantidade de áreas de interesse a serem identificadas na imagem, no nosso caso de estudo proposto a quantidade de objetos de interesses são dois como já foi dito anteriormente, o gabarito, e também a identificação do aluno que efetuou a prova.

Ao iniciar o processamento da imagem, na Etapa 1, é necessário saber se a imagem

se encontra na orientação correta, para que a extração e o processamento das áreas extraídas ajam conforme o planejado. Para que a orientação esteja de forma correta, foi adicionado um pequeno quadrado na parte superior esquerda do estudo de caso, podemos notá-lo ao observar a figura 21. Ao processar a imagem e o algoritmo identificar que esse quadrado se encontra presente neste campo do estudo de caso, ele assegura que a orientação está correta, caso não encontre ele rotaciona a imagem em 180 graus para continuar o processo.

Figura 21 – Exemplo de caso de teste alterado para identificação da orientação correta.



Fonte: Elaborada pelo autor

Com a imagem em orientação correta seguimos para a etapa 2 e etapa 3, onde ocorre a identificação e a extração das áreas de interesse. A identificação das áreas é feita através do método já mencionado anteriormente *findContours*, ele reconhece todos os contornos do estudo de caso, porém através da configuração de parâmetros ele só irá retornar os contornos com quantidades de pixels superiores a 2 mil, quantidade que abrange as duas áreas de

interesse. Os 32 gabaritos foram dispostos em duas orientações, a normal e de cabeça para baixo, o que nos dá um total de 64 testes, e em todos eles foram identificadas e extraídas corretamente as áreas como podemos ver um exemplo na figura a seguir

Figura 22 – Reconhecimento correto das áreas de interesse- Gabarito e Identificação do Aluno.

CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS
CAMPUS VII - UNIDADE TIMÓTEO

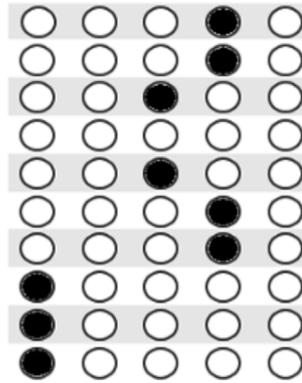
AVALIAÇÃO SOMATIVA

Curso: Desenvolvimento de Sistemas
 Disciplina: Matemática
 Professor: ABC
 Aluno: XYZ

IDENTIFICAÇÃO

	A	B	C	D	E
1	○	○	○	●	○
2	○	○	○	●	○
3	○	○	●	○	○
4	○	○	○	○	○
5	○	○	●	○	○
6	○	○	○	●	○
7	○	○	○	●	○
8	●	○	○	○	○
9	●	○	○	○	○
10	●	○	○	○	○

Figura 23 – Extração correta da área de interesse - Gabarito.



Fonte: Elaborada pelo autor

Figura 24 – Extração correta das áreas de interesse - Identificação do Aluno.



Fonte: Elaborada pelo autor

Seguindo para as etapas finais, etapas onde é efetuado processamento das imagens obtidas para a obtenção dos dados ali presente. Através dos dados auferidos nesta etapa e conferidos com os reais dados, o autor pode julgar se a identificação e a extração das áreas de interesse foram feitas com precisão. Como o caso de teste se trata de uma prova, ao identificar o aluno e corrigir a prova com base em um gabarito já existente o autor julga que o teste foi concluído com sucesso.

A identificação é feita com base em potência de 2, como a identificação é composta de 5 lacunas a serem preenchidas apenas, as mesmas ao serem preenchidas simbolizam o número a ser elevado, de 0 a 4, aumentando o número de lacunas, aumenta-se também o número de identificações, mas para estes testes usaremos apenas 32 identificações. No gabarito, as questões erradas serão marcadas com um círculo vermelho, as corretas com um círculo verde, ao marcar mais de uma questão o autor optou por marcar a questão com uma linha vermelha e ao não marcar nenhuma questão com uma linha branca.

Após o caso de teste passar por todas as etapas temos o resultado como mostra a figura 25, caso tenha mais arquivos a serem processados as etapas se iniciam novamente, se não, é finalizado o programa e todos os resultados são salvos na pasta definida pelo autor.

Figura 25 – Processamento das áreas extraídas .



Fonte: Elaborada pelo autor

Como pode ser observado na figura acima o teste foi concluído com êxito, identificando corretamente todas as questões, incluindo as nulas e as em branco, como pode ser observado na identificação temos o primeiro e a quarto círculo demarcado, sendo assim temos:

$$2^0 + 2^3 = 1 + 8 = 9$$

Com essas informações podemos afirmar que a prova pertence ao aluno 9. No canto direito logo abaixo a identificação podemos ver que a identificação foi feita de forma correta assim também como a correção da prova do aluno 9, que teve a nota 0 baseados em suas respostas e no gabarito correto previamente passado.

O estudo de caso contou com 32 provas, enumeradas de Prova (0) a Prova (32) as provas ficam dispostas em uma pasta como mostra a figura 26, onde o método fica encarregado de carregá-las fazer a detecção e extração da área de interesse.

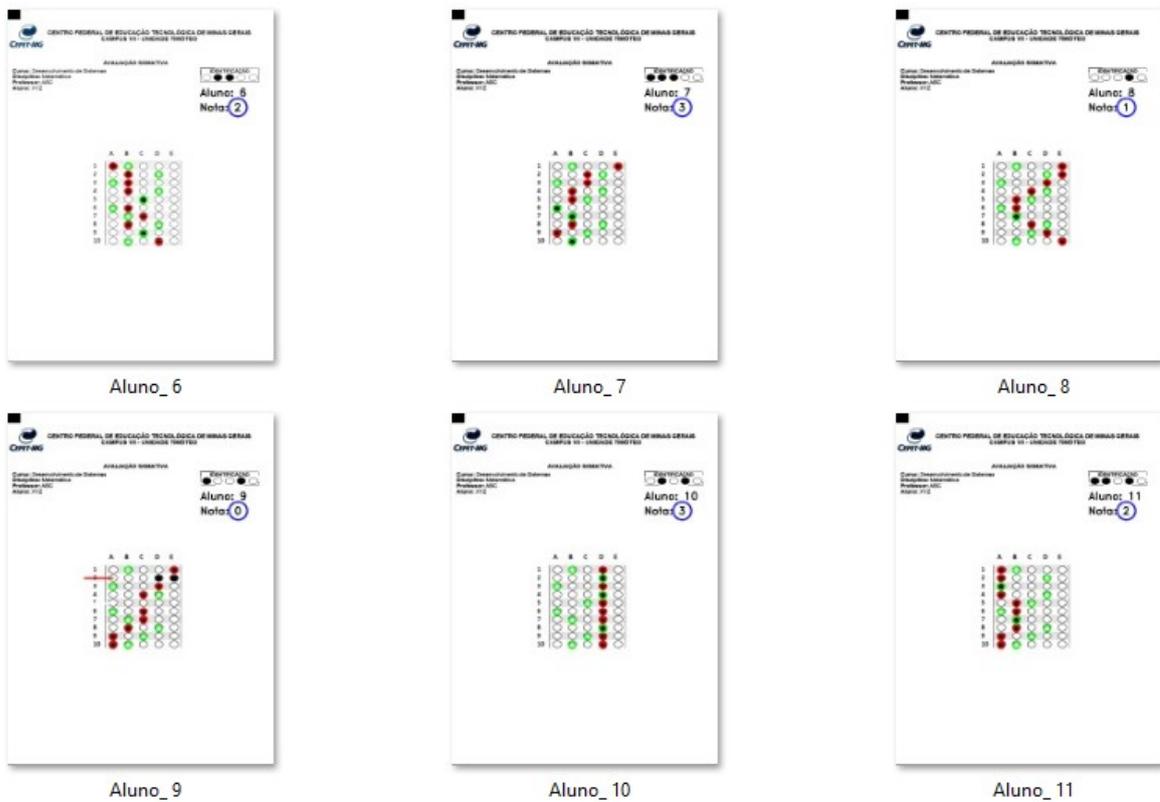
Figura 26 – Provas antes do processamento



Fonte: Elaborada pelo autor

Com a extração da área de interesse concluída os dados ali presentes são processados, e após as informações extraídas, como quais questões foram marcadas e a quem pertence a prova, as provas são salvas assim que corrigidas. Com a correção finalizada e as provas salvas podemos analisar os resultados na pasta de saída, como mostra a figura 27, podemos ver que foi concluído com sucesso as etapas. Em média nos testes, cada prova demorava 1 segundo para ser analisada, detectado a área de interesse, extraída as informações e ser salva corrigida.

Figura 27 – Provas depois do processamento



Fonte: Elaborada pelo autor

6 Conclusão

*“Não é a força mas a constância dos bons resultados
que conduz os homens à felicidade.”
Friedrich Nietzsche*

A partir dos estudos no referencial teórico e nas pesquisas realizadas, este trabalho propôs aplicar um método e analisar sua eficiência na detecção da área de interesse em uma imagem digital. Para realização da atividade proposta, aplicou-se um método exato, com isso, a tarefa foi realizada com êxito. Desse modo, o objetivo geral foi alcançado.

Os objetivos específicos também foram alcançados uma vez que após elencados os métodos, foi possível escolher um que sobressai em ambientes bem definidos e constantes. Utilizando um método que rastreia os contornos em uma imagem, foi possível detectar corretamente região de interesse.

O desenvolvimento desse projeto apresentou algumas dificuldades, principalmente na hora de detectar de forma correta somente as áreas de interesse, o que influenciava diretamente na extração das informações presentes na área. Entretanto, ao achar a configuração correta dos parâmetros, obteve-se os resultados esperados.

Como trabalhos futuros para a extensão desse projeto é possível citar:

- Implementar a leitura de QrCodes ou Código de barras, para melhorar a identificação do aluno e seus atributos.
- Implementar em um APP
- Utilizar programação paralela para melhorar o tempo de correção
- Avaliar a taxa de acerto em uma base de dados maior

Os experimentos realizados mostraram que, utilizando visão computacional e técnicas de processamento digital de imagens, podemos automatizar tarefas rotineiras e/ou repetitivas, que no trabalho em questão era a correção de provas. A identificação das áreas de interesse em todos os 32 casos de testes foram concluídas com êxitos. Com a identificação correta das áreas, as provas puderam ser corrigidas e identificadas de forma precisa em 100% dos casos de testes, com um tempo médio de 0,40 segundos por prova. Ressaltando que a quantidade de casos de testes podem ser aumentados ou diminuídos, alterando a quantidade de lacunas no campo de identificação.

Findando este trabalho, inúmeros problemas e situações podem ser beneficiados através de recursos computacionais. A tipificação de um método de natureza simples, flexível, eficiente e de baixo custo operacional é relevante para incentivar e auxiliar trabalhos futuros

em suas resoluções. Em vista disso a abordagem deste trabalho pode ser aplicada em outros casos que envolvam a detecção da área de interesse em uma dada imagem.

Referências

- ALMEIDAR, L. E. P. M. *Estudo de caso em Reconhecimento Automático de Placas Veiculares usando linguagem de programação Python*. 2018 — Universidade Federal da Paraíba, 2018. Citado na página 16.
- AUGUSTO, K. S. *Identificação automática do grau de maturação de pelotas de minério de ferro*. 2012. Dissertação (Mestrado) — PUC-Rio, 2012. Citado na página 22.
- CONCI, A.; AZEVEDO, E.; LETA, F. R. *Computação gráfica vol. 2*. [S.l.]: Elsevier, 2009. Citado nas páginas 13 e 14.
- DUTRA, S. de S. *Utilização do Algoritmo Haarcascade e Técnicas de Processamento Digital de Imagens para Avaliação de Jatos Pulverizadores*. 2019 — CEFET-MG, 2019. Citado nas páginas 14, 15 e 22.
- GOMES, J.; VELHO, L. *Computação gráfica: imagem*. [S.l.]: IMPA, 2002. Citado nas páginas 20 e 21.
- GONZALEZ, R.; WOODS, R. *Processamento Digital de Imagens. Book*. [S.l.]: São Paulo: Pearson Prentice Hall, 2010. Citado nas páginas 10, 11, 13 e 20.
- GRACIANO, A. B. V. Rastreamento de objetos baseado em reconhecimento estrutural de padrões. *Universidade de Sao Paulo, Sao Paulo, Brasil*, 2007. Citado na página 11.
- GRANATYR, J. *Reconhecimento facial com Python e OpenCV*. 2020. Disponível em: <<https://www.udemy.com/reconhecimento-facial-com-python-e-opencv/learn/lecture/12839868overview>>. Citado na página 23.
- HOLLERWEGER, M. M. *Aplicação de Visão Computacional no Auxílio ao Levantamento de Defeitos em Pavimento Rodoviário*. 2019 — Universidade Federal de Santa Catarina, 2019. Citado nas páginas 15, 16 e 24.
- OPENCV. *Open Source Computer Vision Library*. 2020. Disponível em: <<https://docs.opencv.org/3.4/>>. Citado nas páginas 16, 25 e 31.
- ORTH, A. Desenvolvimento e implementação de um sistema de reconhecimento automático de peças mecânicas em uma célula flexível de manufatura. projeto rap. In: *Projeto RAP Laboratório de Automação Industrial, UFSC. Congresso da Sociedade Brasileira de Computação, SBC*. [S.l.: s.n.], 1998. Citado na página 10.
- PEDRINI, H.; SCHWARTZ, W. R. *Análise de imagens digitais: Princípios. Algoritmos e Aplicações*, Thomson, 2008. Citado na página 10.
- PINTO, P. R. *Utilizando Métodos Exatos no Reconhecimento e Rastreamento de Rbjetos*. 2015 — CEFET, 2015. Citado nas páginas 16, 17 e 27.
- REDMON, J. et al. You only look once: Unified, real-time object detection. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2016. p. 779–788. Citado nas páginas 15 e 23.
- SILVA, B. R. d. A. I. Sistema de contagem automática de objetos utilizando processamento digital de imagens em dispositivos móveis. *Master's thesis. Universidade do Estado do Rio Grande do Norte*, 2014. Citado na página 12.

SZEGEDY, C. et al. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016. Citado na página 15.

USP. *Reconhecimento facial com Python e OpenCV*. 2020. Disponível em: <<http://www.lps.usp.br/hae/apostila/tmatch-ead.pdf>>. Citado nas páginas 25 e 26.

WANGENHEIM, A. v.; COMUNELLO, E. Visão computacional. *Seminário Introdução à Visão Computacional*, p. 1, 2005. Citado na página 10.