MÉTODO PARA DETECÇÃO E MAPEAMENTO DE PESSOAS PARA UM ESPAÇO 2D

Vinícius Luis da Silva

Prof. Aurélio Faustino Hoppe - Orientador

1 INTRODUÇÃO

Com a ascensão da pandemia de COVID-19, inúmeros países adotaram medidas rígidas de distanciamento social, sendo o chamado *lockdown* o mais comum, ou seja, a imposição do confinamento de pessoas por parte do estado. Tais medidas vem surtindo diferentes resultados ao redor do mundo, indicando níveis de sucesso conforme aceitação ou não da população ou dos gestores públicos. No qual, cabe ressaltar que a adoção do confinamento durante um longo período causa impactos profundos na economia, levando-as a recessão nos próximos anos. Diante desta situação, muitos países vêm buscando uma reabertura gradual da economia, removendo o *lockdown* e incentivando medidas para a prevenção do COVID-19 (GODIN *et al*., 2020).

Na seção de "conselhos para o público" da Organização Mundial da Saúde (2020) estão listadas algumas precauções que devem ser tomadas para conter a pandemia do COVID-19, dentre elas estão o uso de máscaras, a constante higienização das mãos e o distanciamento social. Ainda nessa seção, está estabelecido como distanciamento social, uma medida de pelo menos 1 metro de distância entre as pessoas. Essas indicações são repassadas ao redor do mundo para governantes que tentam implementar políticas públicas para sua correta aplicação. Porém, devido à natureza dessas medidas, é muito difícil monitorar com precisão se elas realmente estão sendo aplicadas com sucesso em larga escala, criando então uma situação em que a análise da eficiência dessas medidas é feita utilizando dados imprecisos.

Neste sentido, tentativas de uso da tecnologia para a coleta automatizada e precisa de métricas relacionadas as precauções para evitar o COVID-19 vem sendo feitas ao redor do mundo. Por exemplo, de acordo com Zaruvni (2020), o governo do estado de São Paulo fechou uma parceria com as operadoras telefônicas Vivo, Claro, Oi e Tim no intuito de monitorar o distanciamento social no estado. Assim passando a utilizar a localização geográfica de aparelhos celulares de seus cidadãos para coordenar ações de combate a pandemia.

Segundo Eadicicco (2020), outra alternativa seria a utilização de câmeras de segurança em locais que normalmente tem um grande fluxo de pessoas. Permitindo a extração de informações sobre o distanciamento social nesses locais, auxiliando assim gestores a tomar decisões com a ajuda desses dados. Isso deverá aumentar a assertividade de decisões tomadas e potencialmente pode salvar vidas. Ainda é destacado que sistemas com esse intuito já estão sendo testados, com um grande destaque sendo o chamado "*Distant Assistant*" desenvolvido pela Amazon.

Ainda neste contexto, atualmente existem muitos sistemas que fazem o uso de imagens de câmeras de segurança e técnicas de aprendizado de máquina para a extração de informações. Esses sistemas flutuam em diferentes áreas de atuação, como a utilização de reconhecimento facial em câmeras de segurança com a intenção de identificar e rastrear criminosos em cidades movimentadas (SATARIANO, 2020). Até a otimização de fluxos de trânsito por meio da manipulação de luzes em semáforos (SHAVER, 2017).

Diante deste cenário, o trabalho proposto busca auxiliar o desenvolvimento de análises com base no rastreamento de pessoas em um ambiente fechado com a utilização de coordenadas cartesianas. O problema proposto será resolvido levando em consideração as medidas do local monitorado, buscando indicar distâncias reais entre pessoas. Será desenvolvido um método que deverá ser facilmente extraído para diferentes sistemas, que por sua vez podem disponibilizar uma visualização desses dados, gerar alertas de acordo com algum parâmetro, ou o que for importante para o contexto no qual ele será utilizado.

1.1 OBJETIVOS

O objetivo deste trabalho é disponibilizar um método capaz de detectar pessoas em imagens de câmeras de segurança mapeando-as para um plano 2D.

Os objetivos específicos são:

a) detectar pessoas a partir de câmeras de segurança utilizando técnicas de aprendizado de máquina;
b) estimar e transformar a detecção em coordenadas cartesianas a partir de algoritmos de visão computacional;
c) disponibilizar um mecanismo para visualização de movimentações na forma de um mapa de calor.

2 TRABALHOS CORRELATOS

Neste capítulo serão apresentados trabalhos com características semelhantes aos principais objetivos do estudo proposto. A seção 2.1 apresenta o desenvolvimento de um *framework* para a detecção e rastreamento em coordenadas 3D de múltiplas pessoas (YANG *et al.*, 2020). A seção 2.2 aborda um comparativo de algoritmos conhecidos de detecção de objetos em um cenário de cálculo do índice de distanciamento social de pessoas em tempo real (PUNN *et al.*, 2020). Por fim, a seção 2.3 discorre sobre um estudo de algoritmos de visão computacional para o rastreamento e classificação de clientes e funcionários em lojas de varejo (MUSAV *et al.*, 2020).

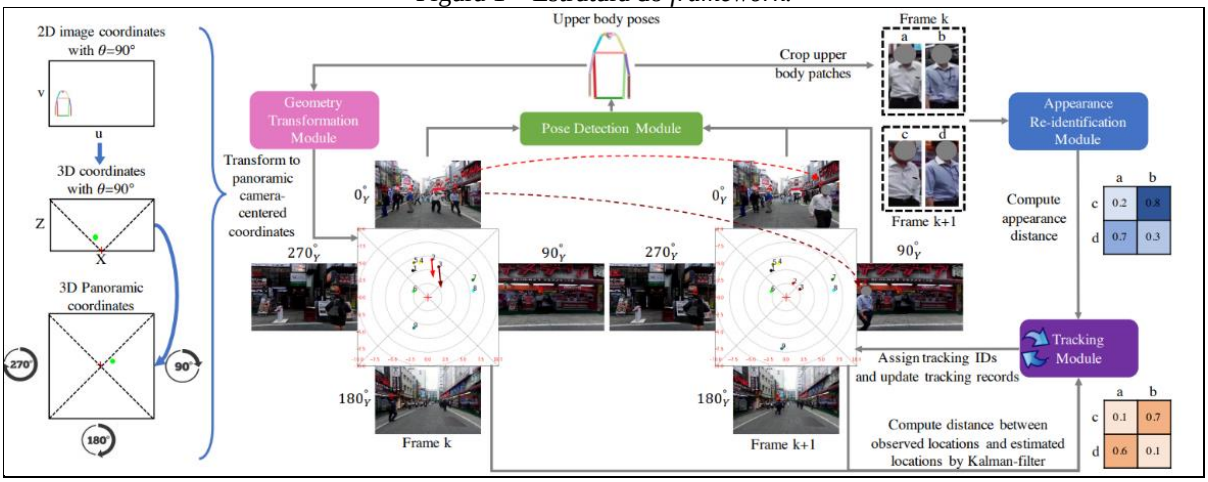
2.1 USING PANORAMIC VIDEOS FOR MULTI-PERSON LOCALIZATION AND TRACKING IN A 3D PANORAMIC COORDINATE

Yang *et al.* (2020) construíram um *framework* para obter a localização e rastreamento 3D de múltiplas pessoas a partir de vídeos panorâmicos. O *framework* proposto é separado em quatro módulos, (i) a detecção e estimativa de pose de pessoas dentro da imagem (ii) a transformação dessa detecção em uma coordenada geográfica (iii) o cálculo do custo de reincidência e (iv) o rastreamento das múltiplas pessoas detectadas.

Segundo Yang *et al.* (2020), o *framework* disponibiliza duas maneiras de se estimar as poses 2D de uma pessoa dentro de uma imagem. A primeira, chamada de *bottom-up*, utiliza o algoritmo de detecção de objetos You Only Look Once (YOLO) para localizar objetos da classe pessoa na imagem analisada e dado o resultado da detecção, é utilizado o algoritmo PifPaf para estimar suas respectivas poses 2D. Já a segunda, chamada de *top-down*, realiza a detecção e a estimativa das poses 2D através do algoritmo PifPaf. A partir dos dados gerados pelas poses estimadas nas pessoas, é feita uma transformação geométrica para o espaço 3D, utilizando a altura das pessoas como uma constante.

Yang *et al.* (2020) ressaltam que para fazer o rastreamento das pessoas detectadas são utilizados dois métodos para calcular o custo de reincidência entre um quadro e outro. Primeiramente, calcula-se o custo, utilizando a aparência assim como algoritmos de aprendizado profundo para a re-identificação de pessoas. Logo em seguida, calcula-se o custo de reincidência a partir da localização estimada da pessoa no espaço 3D, juntamente com o filtro de Kalman. Por fim, realiza-se a junção desses dois custos, no qual as instâncias de pessoas detectadas são rastreadas a partir do módulo de rastreamento das pessoas, conforme mostra a Figura 1.

Figura 1 – Estrutura do *framework*.



Fonte: Yang *et al*. (2020).

Yang *et al.* (2020) construíram uma base de imagens chamada de *Multi-person Panoramic Localization and Tracking* (MPLT), para avaliar a performance do *framework* em relação a localização 3D panorâmica e rastreamento. Além disso, também foram feitos testes utilizando a base de imagens KITTI Vision Benchmark Suite para avaliar a localização 3D de única visão e a 3D *Multi Object Tracking* (MOT) para verificar a eficiência em relação a localização 3D de única visão e rastreamento.

Segundo Yang *et al.* (2020), foram realizados testes com três limites, 0,5m, 1,0m e 2,0m na base de imagens KITTI, comparando a performance com outros três algoritmos, o Mono3D (XIAOZHI *et al.*, 2016), o SAMono (WEI *et al.*, 2018), o MonoDepth (GODARD, AODHA, BROSTOW, 2017) e o Monoloco (BERTONI, KREISS, ALAHI, 2019). Yang *et al.* (2020) ressaltam que o algoritmo possui uma boa generalização, visto que os resultados colocam o *framework* desenvolvido em segundo lugar em termos de localização 3D de única visão, conseguindo uma precisão de 22,0%, 39,4% e 63,2% dado seus respectivos

limites. Na base 3D MOT, Yang *et al.* (2020) realizaram testes utilizando a métrica *Multi Object Tracking Accuracy* (MOTA), comparando o *framework* proposto com os algoritmos previamente descritos. Eles ressaltam que o *framework* obteve melhor performance entre os algoritmos no critério de rastreamento, conseguindo uma precisão de 54,2%. Além disso, Yang *et al.* (2020) também testaram a própria base de imagens no qual foi utiliza a métrica MOTA, sendo avaliados os limites de 0,5m e 1,0m, obtendo uma precisão de 65,2% e 74,9%, respectivamente.

Yang *et al.* (2020) concluem que o *framework* é promissor, conseguindo resultados bons em bases de imagens já conhecidas e utilizadas como base para o teste de muitos outros algoritmos. Além disso, os autores comentam que devido à natureza do *framework* proposto, ele pode auxiliar na criação de aplicações que visam utilizar o entendimento de vídeos de pessoas. Eles também mencionam as perspectivas de extensão do *framework*, permitindo realizar a detecção e reconhecimento automático de atividades humanas em coordenadas 3D panorâmicas.

2.2 MONITORING COVID-19 SOCIAL DISTANCING WITH PERSON DETECTION AND TRACKING VIA FINE-TUNED YOLO V3 AND DEEPSORT TECHNIQUES

Punn *et al.* (2020) analisaram a performance de diferentes algoritmos para detecção e rastreamento de objetos em um sistema que visa calcular o índice de distanciamento social de pessoas com base em imagens de câmeras de segurança. Segundo os autores, foram utilizados três modelos de detecção de objetos para os testes de performance, o *Faster Region Based Convolutional Neural Networks* (R-CNN) (REN *et al.*, 2015), *Single shot detection* (SSD) (LIU *et al.*, 2016) e YOLOv3 (REDMON *et al.*, 2018). No rastreamento, utilizou-se o algoritmo DeepSORT (WOJKE, EWLEY, PAULUS, 2018).

Punn *et al.* (2020) realizaram inicialmente uma sintonia fina para a classificação binária (pessoa ou não pessoa), utilizando a Rede Neural Artificial *Inception V2* (SZEGEDY *et al.* 2016) como rede base e o *Open Image Dataset* (OID) filtrado para conter somente amostras de pessoas reais, resultando em uma base de 800 imagens. Posteriormente, utilizaram as imagens das câmeras de segurança no modelo, que delimita as pessoas localizadas através de uma *bounding box*, ou seja, uma caixa delimitadora ao redor da detecção, cada uma com seu identificador único, sua posição e a distância da pessoa em relação a câmera. Por fim, os autores calcularam a norma L2 para cada par de detecções e com base em um limite de proximidade, apontavam a violação do distanciamento social. Os autores destacam que tal limite foi determinado a partir de testes considerando a localização espacial das pessoas em um dado quadro, ao qual variava de 90 a 170 pixels. Já em relação ao índice de distanciamento social, os autores apontam que ele foi calculado com base na formação de grupos de pessoas violando o distanciamento social, ou seja, dividiram o número de pessoas em violação pelo número de grupos detectados. A Figura 2 exemplifica o funcionamento do sistema de monitoramento.

Figura 2 – Demonstração do sistema de monitoramento proposto.

Fonte: Punn *et al.* (2020).

Segundo Punn *et al.* (2020), os resultados dos testes com diferentes detectores de objetos evidenciaram que o Faster RCNN é mais eficiente, tendo a maior *Mean Avarage Precision* (mAP) (97%). Porém, sua taxa de 3 *Frames Per Second* (FPS) foi a mais baixa entre os modelos. Fato, que segundo os autores, dificulta sua utilização em aplicações em tempo real. Os autores também apontam que o modelo mais adequado considerando performance e acurácia é o YOLO v3, que processa 23fps com um mAP de 85%. Também é destacado o tempo de treinamento de cada modelo, onde o YOLO v3 aparece em segundo lugar com 5659 segundos, perdendo somente para o SSD, com 2124 segundos. O número de iterações de cada algoritmo coloca o SSD e o YOLO v3 como líderes novamente, com um número igual a 1200 e 7560, respectivamente.

Por fim, Punn *et al*. (2020) destacam que a aplicação desenvolvida tem como objetivo ser utilizada em ambientes de trabalho, aumentando assim, a importância da acurácia e precisão dos modelos pois, caso eles resultem um número alto de falsos positivos poderão causar pânico e desconforto entre as pessoas monitoradas. Além disso, os autores demonstram preocupação com relação a privacidade das pessoas, sugerindo o desenvolvimento de uma funcionalidade para ocultar os rostos das pessoas.

2.3 TOWARDS IN-STORE MULTI-PERSON TRACKING USING HEAD DETECTION AND TRACK HEATMAPS

Musav *et al.* (2020) utilizaram técnicas de visão computacional para estabelecer comportamentos de consumo em lojas de varejo a partir do rastreamento de clientes, identificando padrões de compra, fraudes e o tráfego na loja. Para isso, foram utilizadas câmeras no teto e algoritmos de rastreamento para gerar mapas de calor das movimentações dentro da loja. Ainda segundo os autores, foram utilizados algoritmos de detecção com o foco na cabeça das pessoas rastreadas, diminuindo assim problemas de oclusão na estimativa da localização da pessoa dentro do espaço geográfico. Também foram treinados modelos para distinguir clientes de funcionários, com base nos seus padrões de movimentação. Além disso, criou-se uma base de imagens com participantes voluntários simulando alguns comportamentos típicos de clientes em lojas de varejo, assim como uma base de imagens densamente rotuladas coletadas durante 24 horas em um supermercado, conforme exibe a Figura 3.

Figura 3 – Exemplo de imagem retirada da base de imagens de participantes voluntários.

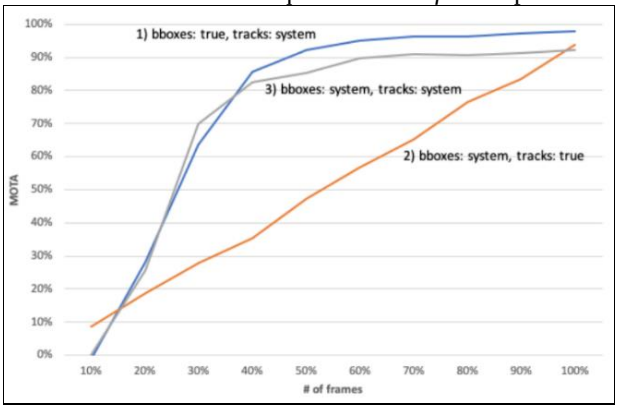
Fonte: Musav *et al.* (2020).

Musav *et al.* (2020) avaliaram a performance de dois detectores de objetos, o YOLO v3 (REDMON *et al.*, 2018) e o SSD ResNet-50 em vídeos com uma quantidade de *frames* originais progressivamente menor, com o objetivo de verificar a assertividade dos detectores em vídeos com menos *frames* por segundo. Ambos os detectores apresentam uma correlação linear em relação ao número de quadros e as métricas MOTA, no qual o YOLOv3 apresentou uma performance melhor em relação ao SSD ResNet-50.

Segundo Musav *et al.* (2020), também foram testados o algoritmo de DeepSORT para o módulo de rastreamento, adotando a estratégia IOU, ou seja, com ou sem extração de características. Os resultados apontaram que o rastreamento sem extração de características é mais adequado em vídeos com menos quadros e, que, a estratégia com extração de características precisa de vídeos com mais quadros preservados para obter uma boa performance.

Musav *et al.* (2020) também analisaram a utilização em conjunto dos algoritmos YOLO v3 e o DeepSORT sem extração de características. Eles observaram que a performance dos algoritmos foi inferior a performance do algoritmo de rastreamento de pessoas em separado. Porém, supreendentemente, a performance deles foi melhor que o algoritmo de detecção executado isoladamente. Os resultados podem ser vistos na Figura 4.

Figura 4 – Resultado dos testes com os algoritmos em isolamento e em conjunto. O eixo y indica a performance MOTA. O eixo x indica a quantidade de *frames* presentes.



Fonte: Musav *et al.* (2020)

Segundo Musav *et al.* (2020), a performance do algoritmo que prediz se uma trajetória formada é de um cliente ou de um funcionário apresentou uma taxa de erro de 31%, sendo considerada muito alta. A partir disso, os autores removeram as trajetórias curtas, aumentando assim o nível de acurácia, chegando em 93% de taxa de acertos. Por fim, os autores concluem indicando que o resultado obtido pelo modelo de análise de comportamento de pessoas rastreadas mostra-se promissor.

3 PROPOSTA DO MÉTODO

Esse capítulo tem como objetivo apresentar a justificativa para a elaboração deste trabalho, assim como os requisitos e metodologias que serão adotadas. Também serão apresentadas breves revisões bibliográficas das principais áreas de estudo que serão exploradas.

3.1 JUSTIFICATIVA

No Quadro 1 é possível observar um comparativo métodos escolhidos para a resolução dos problemas propostos pelos respectivos trabalhos, onde as linhas representam as características destacadas e as colunas representam os trabalhos.

Quadro 1 – Comparativo entre trabalhos correlatos

Características	Yang *et al.* (2020)	Punn *et al.* (2020)	Musav *et al.* (2020)
Cenário / objetivo	Estimar a posição 3D de pessoas utilizando um ponto de vista único	Calcular o distanciamento social entre pessoas	Detectar e rastrear pessoas em lojas de varejo
Tipo da câmera	Câmera 360º	Câmera de segurança	Câmera de segurança posicionada no teto
Extração de localização geográfica	Sim	Não	Sim
Algoritmo de detecção de objetos	YOLO e PifPaf	Faster RCNN, SSD e YOLO v3	YOLO v3 e SSD ResNet-50
Algoritmo de rastreamento de pessoas	Algoritmo próprio	Deep SORT	Deep SORT e Deep SORT IOU

Fonte: elaborado pelo autor.

A partir do Quadro 1 é possível identificar que os trabalhos utilizam câmeras posicionadas de maneira diferente no cenário espacial explorado. Yang *et al.* (2020) propõem o uso de câmeras 360º posicionadas no centro do espaço monitorado, Punn *et al.* (2020) utilizam câmeras de segurança sem nenhum tipo de restrição e Musav *et al.* (2020) propuseram o uso de uma única câmera posicionada no teto do espaço monitorado.

Todos os trabalhos possuem objetivos diferentes, com Yang *et al.* (2020) e Musav *et al.* (2020) propõem maneiras diferentes de se olhar para o problema de encontrar a localização geográfica de inúmeras pessoas em um espaço geográfico. Yang *et al.* (2020) partem do princípio de que uma máquina possa entender seus arredores, por isso a escolha de uma única câmera 360º localizada em um eixo y relativamente igual ao dos objetos rastreados. Já Musav *et al.* (2020) focam em uma solução para geração de mapas de calor em ambiente internos movimentados, tendo como objetivo a análise desses dados para aplicação no varejo. Já Punn *et al.* (2020) propõem uma solução um pouco diferente para um problema que está relacionado a localização de pessoas em um espaço geográfico, a medição de um índice de distanciamento social. Para isso não é utilizado

em si uma coordenada, porém são analisados dados extraídos das imagens que se relacionam ao espaço geográfico, como a medida de distância entre uma pessoa e outra.

Também pode ser observado que, com exceção do trabalho de Punn *et al*. (2020), são utilizadas câmeras em posições pré-definidas, com o objetivo de extrair as coordenadas das pessoas detectadas. Dessa forma, torna-se o diferencial deste trabalho, pois ele propõe a implementação de um método para o mapeamento geográfico de pessoas através de câmeras de segurança, sem depender de ângulos específicos. Portanto, pretende-se contribuir para a área de segurança e monitoramento de pessoas, disponibilizando um método de fácil implantação em sistemas já existentes, visto que, não será necessário nenhum tipo de localização da câmera com o objetivo de facilitar a identificação das coordenadas cartesianas. Além disso, este trabalho também propõe o desenvolvimento de uma aplicação web, que tem como objetivo facilitar a visualização dos dados através de mapas de calor. Essa aplicação será desenvolvida de uma maneira modular para se encaixar em arquiteturas de software mais complexas. Contudo, a aplicação final poderá ser utilizada no ramo comercial, podendo facilitar por exemplo, a identificação de potenciais clientes dentro de lojas de varejo, ou até mesmo, auxiliar na manutenção do distanciamento social, podendo ser uma ferramenta importante de apoio ao controle de pandemias, como por exemplo da COVID-19.

3.2 REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO

Os requisitos estão divididos em duas partes, uma relacionada a aplicação *web* que permite o cadastro de vídeos e a visualização do mapa de calor gerado para o ambiente monitorado, e outra relacionada ao método de detecção e mapeamento geográfico para um plano 2D.

A aplicação web deverá:

a) permitir que o usuário cadastre o plano 2D monitorado pelas imagens de câmeras de segurança, com suas devidas medidas (Requisito Funcional - RF);
b) permitir que o usuário faça o *upload* dos vídeos que serão usados para detectar e mapear as pessoas para o plano 2D (RF);
c) permitir que o usuário faça o cadastro da localização das câmeras que capturam as imagens fornecidas (RF);
d) permitir ao usuário visualizar o ambiente 2D na forma de um mapa de calor (RF);
e) permitir ao usuário visualizar a quantidade de pessoas rastreadas (RF);
f) utilizar a linguagem de programação Python juntamente com o *framework* Flask para o desenvolvimento da API rest (Requisito Não Funcional - RNF);
g) utilizar um banco de dados não relacional para guardar os dados gerados (RNF);
h) utilizar o framework VueJS para o desenvolvimento da página web que permitirá a interação do usuário (RNF).

O método de detecção e mapeamento das pessoas para um plano 2D deverá:

a) utilizar técnicas de aprendizado de máquina para a detecção de pessoas (RF);
b) utilizar um algoritmo de *tracking* para o rastreamento das pessoas (RF);
c) utilizar um algoritmo para transformar os *bouding boxes* detectados em coordenadas cartesianas (RF);
d) ser desenvolvido na linguagem de programação Python (RNF);
e) utilizar o Keras para o auxílio na utilização de um algoritmo de detecção de objetos e o OpenCV para auxiliar na utilização de um algoritmo de rastreamento de objetos (RNF).

3.3 METODOLOGIA

O trabalho será desenvolvido observando as seguintes etapas:

a) coleta de imagens de câmeras de segurança: coletar as imagens que serão usadas nos testes do método proposto. Serão pesquisadas opções de bases de dados públicas e também será avaliada a coleta de imagens em um ambiente específico controlado;
b) rotulação das imagens coletadas: rotular as imagens que serão utilizadas, permitindo assim que o algoritmo possa ter sua assertividade testada;
c) pesquisa e escolha do algoritmo de detecção de objetos: pesquisar os principais algoritmos de detecção de objetos, escolhendo o mais adequado para o desenvolvimento do trabalho;
d) pesquisa e escolha do algoritmo de rastreamento de objetos: pesquisar os principais algoritmos de rastreamento de objetos, escolhendo o mais adequado para o desenvolvimento do trabalho;
e) levantamento de formas para calcular a localização das pessoas detectadas: pesquisar métodos para se mapear a posição geográfica das pessoas detectadas nas imagens para o espaço 2D proposto;

f) desenvolvimento do método: desenvolver o método de detecção e mapeamento geográfico de pessoas detectadas para um plano 2D utilizando a linguagem de programação Python e a biblioteca Keras juntamente com o OpenCV;
g) testes da detecção e mapeamento 2D: em paralelo ao desenvolvimento, verificar a assertividade do método proposto a partir do percentual de objetos rastreados e mapeados corretamente, e caso necessário, alterar os requisitos para atender o problema a ser resolvido;
h) elicitação de requisitos da aplicação web: detalhar e reavaliar os requisitos propostos para a aplicação e, se necessário, especificar outros a partir das necessidades observadas ao longo do trabalho;
i) especificação: elaborar os diagramas de casos de uso e de classes de acordo com a Unified Modeling Language (UML), utilizando a ferramenta Astah;
j) desenvolvimento: a partir do item (i) desenvolver a aplicação web que será utilizada para permitir ao usuário cadastrar vídeos para análise e visualizar os resultados obtidos pelo método proposto, sendo utilizado a linguagem de programação Python com o *framework* Flask e um banco de dados não relacional para o servidor, e o *framework* VueJS para o desenvolvimento da página web;
k) testes da aplicação web: elaborar testes para validar a usabilidade da aplicação.

As etapas serão realizadas nos períodos relacionados no Quadro *2*.

Quadro 2 – Cronograma de atividades a serem realizadas

	2020				2021											
	nov.		dez.		jan.		fev.		mar.		abr.		maio		jun.	
etapas / quinzenas	1	2	1	2	1	2	1	2	1	2	1	2	1	2	1	2
coleta de imagens de câmeras de segurança																
rotulação das imagens coletadas																
pesquisa e escolha do algoritmo de detecção de objetos																
pesquisa e escolha do algoritmo de rastreamento de objetos																
levantamento de formas para calcular a localização das pessoas detectadas																
desenvolvimento do método																
testes da detecção e mapeamento 2D																
elicitação de requisitos da aplicação web																
especificação																
desenvolvimento																
testes da aplicação web																

Fonte: elaborado pelo autor.

4 REVISÃO BIBLIOGRÁFICA

Neste capítulo serão descritos os principais assuntos que serão estudados ao longo do desenvolvimento do projeto. A seção 4.1 apresenta uma introdução ao tema de detecção de objetos, que é parte fundamental do método proposto, fazendo um breve resumo de alguns dos principais métodos utilizados para abordar problemas desse sentido. A seção 4.2 introduz o tempo de rastreamento de objetos, identificando os pontos principais de como esses algoritmos funcionam, e também falando de limitações desse campo de estudos.

4.1 DETECÇÃO DE OBJETOS

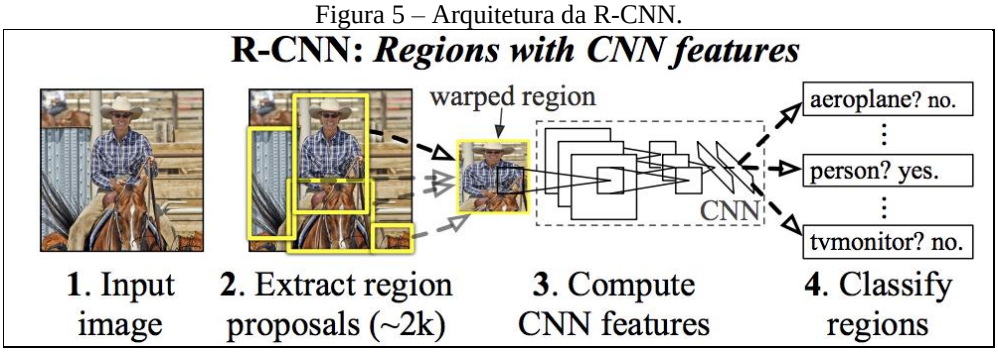
Brownlee (2019) descreve detecção de objetos, como a tarefa conjunta de encontrar um objeto em uma dada imagem, desenhar uma *bouding box* ao seu redor e classifica-lo. Segundo o autor, essa tarefa é geralmente realizada por algoritmos baseados em *Convolutional Neural Networks* (CNN), como o R-CNN, Fast R-CNN e Faster R-CNN. Mas, também pode ser desempenhada por algoritmos da família *You Only Look Once* (YOLO), que embora seja menos preciso que algoritmos como o Faster R-CNN, possui uma performance muito alta, conseguindo fazer a detecção de 45 quadros por segundo.

Segundo Brownlee (2019), a maior parte das recentes evoluções para algoritmos de detecção de objetos tornou-se possível graças a *ImageNet Large Scale Visual Recognition Challenge* (ILSVRC), que é uma competição anual que classifica algoritmos de acordo com sua performance em diferentes tipos de problemas relacionados a visão computacional. Essa competição tem como objetivo promover avanços individuais para diversos tipos de problemas, tendo como intuito criar algoritmos que possam ser usados de forma mais ampla.

Brownlee (2019) destaca que a R-CNN foi um dos primeiros casos de sucesso relacionados a tarefa de detecção de objetos, sendo utilizado atualmente como base para diversos outros algoritmos, como o Fast R-CNN

e Faster R-CNN. Esse modelo é composto por três módulos, o primeiro (i) chamado de *Region Proposal*, onde são geradas e extraídas regiões de interesse, ou seja, *bounding boxes*, o segundo (ii) chamado de *Feature Extractor*, onde são extraídas as características de cada região de interesse utilizando uma *Deep Convolutional Neural Network*, e o terceiro (iii) chamado de *Classifier*, onde cada região é classificada em uma das classes conhecidas, utilizando um *Support Vector Machine* (SVM). A Figura 5 apresenta a arquitetura da R-CNN.

Figura 5 – Arquitetura da R-CNN.

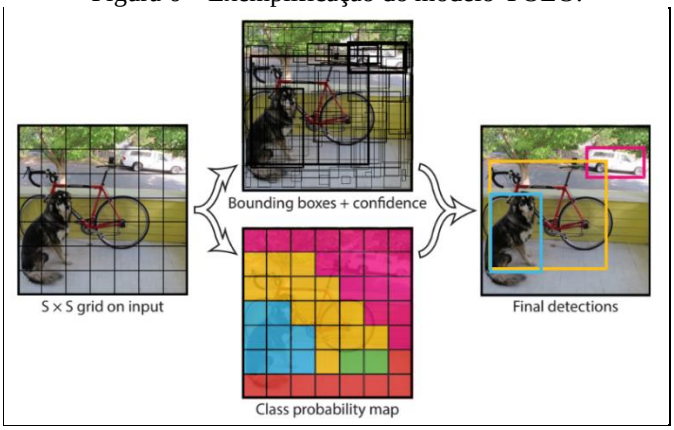


Fonte: Brownlee (2019).

Segundo Brownlee (2019), outro algoritmo para a detecção de objetos muito popular é o YOLO. Mesmo sendo menos preciso, é muito mais veloz do que algoritmos da família R-CNN. Ele utiliza somente uma CNN que recebe como entrada uma imagem e retorna como saída inúmeras *bouding boxes*, cada uma com seu respectivo rótulo e classe.

De acordo com Brownlee (2019), o modelo YOLO divide a imagem em uma grade de células, onde cada célula é responsável pelas detecções de objetos. Essa saída contém a largura e altura de cada objeto, as coordenadas dele na imagem e o nível de confiança da detecção. Além disso, cada cédula é responsável por indicar a classe do objeto. Ao final, as *bounding boxes* e classes são combinados em uma única saída, assim, utilizando um nível de confiança mínimo, os objetos são detectados. A Figura 6 exemplifica as duas saídas e sua combinação.

Figura 6 – Exemplificação do modelo YOLO.



Fonte: Brownlee (2019).

Segundo Brownlee (2019), assim como o R-CNN, o YOLO também serviu como base para modelos mais eficientes. O autor destaca o YOLOv2, que trouxe inúmeras mudanças de treinamento e arquiteturais para o modelo, como o uso de normalização por lote e de entradas com resoluções superiores. E, o YOLOv3, que trouxe como melhoria a detecção de características mais profundas.

4.2 RASTREAMENTO DE OBJETOS

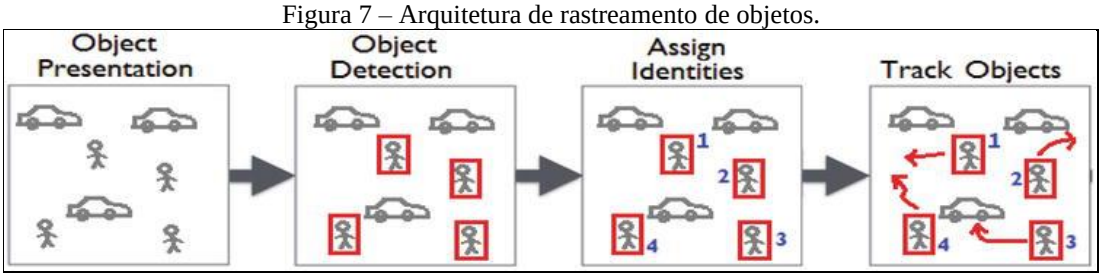
Segundo Shah (2020), o rastreamento de objetos significa estimar a posição de um objeto em uma cena de acordo como informações anteriores. Em um alto nível de abstração, existem principalmente dois tipos de rastreamento de objetos, *Single Object Tracking* (SOT) e *Multi Object Tracking* (MOT). O SOT se refere a algoritmos que visam detectar e rastrear um único objeto em uma cena ao longo do tempo, enquanto o MOT é o ato de detectar em uma cena um ou mais objetos, de uma ou mais categorias e rastreá-los ao longo do tempo.

De acordo com Shah (2020), em algoritmos de rastreamento de objetos únicos a aparência do objeto é conhecida anteriormente. Já para o rastreamento de múltiplos, primeiro é necessário realizar a detecção, identificando os alvos que podem entrar ou sair da cena. Ainda segundo o autor, a maior dificuldade encontrada

em algoritmos que trabalham com múltiplos objetos ocorre devido as oclusões e interações entre objetos que possuem aparência similar, logo, aplicar apenas algoritmos que rastreiam um único objeto normalmente não garante altos índices de precisão.

Shah (2020) destaca que o crescimento do uso de algoritmos de *Deep Learning*, para realizar a detecção de objetos, resultou na melhora da precisão dos algoritmos de rastreamento. Isso ocorreu porque a maioria deles segue o esquema de *Tracking by detection*, onde primeiro são detectados os objetos da cena, e posteriormente são realizadas as predições de onde o objeto deve aparecer na sequência. A Figura 7 apresenta um exemplo de arquitetura para rastreamento de objetos.

Figura 7 – Arquitetura de rastreamento de objetos.



Fonte: Shah (2020).

Petrovicheva (2020) explica o funcionamento dos algoritmos de *tracking* tendo como exemplo o problema de rastrear múltiplos pedestres ao longo de um vídeo. Segundo a autora, tais algoritmos utilizam de maneira geral um vetor de números que de alguma maneira descrevem a aparência de uma pessoa a partir de múltiplas detecções. Contudo, os vetores relacionados a detecção de uma mesma pessoa devem ser muitos similares, enquanto vetores de pessoas distintas devem ser diferentes.

Segundo Petrovicheva (2020), existem duas maneiras de se implementar algoritmos de detecção de pedestres aplicados ao conceito de rastreamento. A primeira delas é utilizando a detecção de faces, onde é feito um recorte do rosto da pessoa detectada, e a partir disso, são criados os vetores utilizados para o rastreamento. A autora destaca também que o lado negativo dessa abordagem é que os vetores criados não podem, por exemplo, ser utilizados para a re-identificação de pessoas de costas. Já a segunda maneira seria a detecção do corpo inteiro e o uso dele para a criação desses vetores, que, ao contrário da detecção de faces, consegue realizar a re-identificação de pessoas de costas. Porém, não é possível realizar a re-identificação de pessoas durantes longos períodos, por exemplo, se a pessoa sai da cena e retorna com uma roupa completamente diferente.

Petrovicheva (2020) destaca que durante o *matching* de detecções em vídeos deve-se observar alguns pontos. O primeiro diz respeito ao quanto os vetores que descrevem a aparência das detecções se parecem. O segundo corresponde a distância entre os centros das *bounding boxes* detectadas. Neste caso, assume-se que em um vídeo com uma alta taxa de *frames*, uma pessoa não irá simplesmente se movimentar entre pontos muito distantes de um *frame* para o outro. E, o terceiro tem a ver com a diferença entre os tamanhos das *bounding boxes*, onde também se admite que esse tamanho não deve sofrer muitas alterações ao longo do vídeo.

De acordo com Shah (2020), embora nos últimos anos tenham sido feitos grandes avanços nessas áreas, ela ainda acompanha inúmeros desafios que devem impulsionar seu estudo no futuro. Como por exemplo, diferenças em fatores dos *frames* capturados (iluminação, aparência etc.), movimentações não lineares, resoluções baixas, objetos similares na cena e cenários com muitos objetos. Além disso, antes de se iniciar os algoritmos, são necessárias informações sobre quais objetos devem ser rastreados, no qual, em muitos cenários isso não é possível.

REFERÊNCIAS

BERTONI, Lorenzo; KREISS, Sven; ALAHI, Alahi; **Monoloco: Monocular 3d pedestrian localization and uncertainty estimation**. In: IEEE/CVF International Conference on Computer Vision (ICCV), 2019, Seoul. **Proceedings…** Seoul: IEEE, 2019. p. 6861-6871.

BROWNLEE, Jason. **A Gentle Introduction to Object Recognition With Deep Learning**. [S.I.], [2019]. Disponível em <https://machinelearningmastery.com/object-recognition-with-deep-learning/>. Acesso em: 02 out. 2020.

EADICICCO, Lisa. **Smart AI-powered cameras that can tell how close you are to other people may be the answer to maintaining social distancing as the US reopens**. [S.I.], [2020]. Disponível em <https://www.businessinsider.com/ai-surveillance-cameras-used-for-social-distancing-coronavirus-us-reopening-2020-6>. Acesso em: 12 out. 2020.

GODARD, Clément; AODHA, Oisin Mac; BROSTOW, Gabriel J.; Unsupervised monocular depth estimation with left-right consistency. In: Conference on computer vision and pattern recognition (CVPR), 2017, Honolulu. **Proceedings…** Honolulu: IEEE, 2017. p. 270-279.

GODIN, Mélissa. **These European Countries Are Slowly Lifting Coronavirus Lockdowns. Here's What That Looks Like.** [S.I.], [2020]. Disponível em < https://time.com/5822470/countries-lifting-coronavirus-restrictions-europe/>. Acesso em: 12 out. 2020.

LIU, Wei *et al*. SSD: Single Shot MultiBox Detector. In: European conference on computer vision, 2016, Amsterdam. **Proceedings…** Amsterdam: Springer, 2016. p. 21-37.

MUSAV, Aibek *et al.* **Towards in-store multi-person tracking using head detection and track heatmaps**. [S.I.], [2020]. Disponível em <https://www.researchgate.net/publication/341477855_Towards_in-store_multi-person_tracking_using_head_detection_and_track_heatmaps>. Acesso em: 30 set. 2020.

ORGANIZAÇÃO MUNDIAL DA SAÚDE. **Coronavirus disease (COVID-19) advice for the public**. [S.l.], 2020. Disponível em: < https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public>. Acesso em 03 out. 2020.

PETROVICHEVA, Anna. **Multiple Object Tracking in Realtime**. [S.I.], [2020]. Disponível em <https://opencv.org/multiple-object-tracking-in-realtime/>. Acesso em: 21 nov. 2020.

PUNN, Narinder Singh *et al.* **Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques**. [S.I.], [2020]. Disponível em <https://ui.adsabs.harvard.edu/abs/2020arXiv200501385S>. Acesso em: 30 set. 2020.

REDMON, Joseph *et al.* **YOLOv3: An Incremental Improvement**. [S.I], [2018]. Disponível em <https://arxiv.org/abs/1804.02767>. Acesso em: 09 out. 2020.

REN, Shaoqing *et al.* Faster r-cnn: Towards real-time object detection with region proposal networks. In: Neural information processing systems (NIPS), 2015, Montreal. **Proceedings…** Montreal: Advances in Neural Information Processing Systems, 2015. p. 91-99

SATARIANO, Adam. **London Police Are Taking Surveillance to a Whole New Level.** [S.I.], [2020]. Disponível em <https://www.nytimes.com/2020/01/24/business/london-police-facial-recognition.html>. Acesso em: 11 out. 2020.

SHAH, Deval. **The Surveillance phenomenon you must know about : Multi Object Tracking**. [S.I.], [2020]. Disponível em <https://medium.com/visionwizard/object-tracking-675d7a33e687>. Acesso em: 03 out. 2020.

SHAVER, Katherine. **'Smart' traffic signals soon will change themselves in Maryland.** [S.I.], [2017]. Disponível em <https://www.washingtonpost.com/local/trafficandcommuting/smart-traffic-signals-soon-will-change-themselves-in-maryland/2017/10/25/d4f57058-b9bf-11e7-9e58-e6288544af98_story.html>. Acesso em: 11 out. 2020.

SZEGEDY, Christian *et al.* Performance Rethinking the inception architecture for computer vision. In: Conference on computer vision and pattern recognition (CVPR), 2016, Las Vegas. **Proceedings…** Las Vegas: IEEE, 2016. p. 2818-2826.

WEI, Dan *et al.*; Structured attention guided convolutional neural fields for monocular depth estimation. In: Conference on computer vision and pattern recognition (CVPR), 2018, Salt Lake City. **Proceedings…** Salt Lake City: IEEE, 2018. p. 3917-3925.

WOJKE, Nicolai; BEWLEY Alex; PAULUS Dietrich; Simple online and realtime tracking with a deep association metric. In: International conference on image processing (ICIP), 2018, Beijing. **Proceedings…** Montreal: IEEE, 2018. p. 3645-3649.

XIAOZHI, Chen *et al.*; Monocular 3d object detection for autonomous driving. In: Conference on computer vision and pattern recognition (CVPR), 2016, Las Vegas. **Proceedings…** Las Vegas: IEEE, 2016. p. 2147-2156.

YANG, Fan *et al.* Using Panoramic Videos for Multi-Person Localization and Tracking In A 3D Panoramic Coordinate. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020, Las Vegas. **Proceedings…** Las Vegas: IEEE, 2020. p. 1863-1867.

ZARUVNI, Reinaldo. **São Paulo inicia monitoramento de celulares para conter coronavírus**. [S.I.], [2020]. Disponível em <https://www.tecmundo.com.br/ciencia/151977-paulo-inicia-monitoramento-celulares-conter-coronavirus.htm>. Acesso em: 09 nov. 2020.

ASSINATURAS

(Atenção: todas as folhas devem estar rubricadas)

Assinatura do(a) Aluno(a): _____

Assinatura do(a) Orientador(a): _____

Assinatura do(a) Coorientador(a) (se houver): _____

Observações do orientador em relação a itens não atendidos do pré-projeto (se houver):

FORMULÁRIO DE AVALIAÇÃO – PROFESSOR AVALIADOR

Acadêmico(a): Vinícius Luis da Silva_____

Avaliador(a): Miguel Alexandre Wisintainer _____

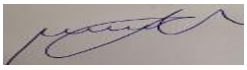
	ASPECTOS AVALIADOS[1]	atende	atende parcialmente	não atende
ASPECTOS TÉCNICOS	1. INTRODUÇÃO O tema de pesquisa está devidamente contextualizado/delimitado?	x		
	O problema está claramente formulado?	x		
	1. OBJETIVOS O objetivo principal está claramente definido e é passível de ser alcançado?	x		
	Os objetivos específicos são coerentes com o objetivo principal?	x		
	2. TRABALHOS CORRELATOS São apresentados trabalhos correlatos, bem como descritas as principais funcionalidades e os pontos fortes e fracos?	x		
	3. JUSTIFICATIVA Foi apresentado e discutido um quadro relacionando os trabalhos correlatos e suas principais funcionalidades com a proposta apresentada?	x		
	São apresentados argumentos científicos, técnicos ou metodológicos que justificam a proposta?	x		
	São apresentadas as contribuições teóricas, práticas ou sociais que justificam a proposta?	x		
	4. REQUISITOS PRINCIPAIS DO PROBLEMA A SER TRABALHADO Os requisitos funcionais e não funcionais foram claramente descritos?	x		
	5. METODOLOGIA Foram relacionadas todas as etapas necessárias para o desenvolvimento do TCC?	x		
	Os métodos, recursos e o cronograma estão devidamente apresentados e são compatíveis com a metodologia proposta?	x		
	6. REVISÃO BIBLIOGRÁFICA (atenção para a diferença de conteúdo entre projeto e pré-projeto) Os assuntos apresentados são suficientes e têm relação com o tema do TCC?	x		
	As referências contemplam adequadamente os assuntos abordados (são indicadas obras atualizadas e as mais importantes da área)?	x		
ASPECTOS METODOLÓGICOS	7. LINGUAGEM USADA (redação) O texto completo é coerente e redigido corretamente em língua portuguesa, usando linguagem formal/científica?	x		
	A exposição do assunto é ordenada (as ideias estão bem encadeadas e a linguagem utilizada é clara)?	x		

PARECER – PROFESSOR AVALIADOR:
(PREENCHER APENAS NO PROJETO)

O projeto de TCC ser deverá ser revisado, isto é, necessita de complementação, se:

- qualquer um dos itens tiver resposta NÃO ATENDE;
- pelo menos **5 (cinco)** tiverem resposta ATENDE PARCIALMENTE.

PARECER: (x) APROVADO ( ) REPROVADO

Assinatura:  _____ Data: 30/11/2020 _____

[1] Quando o avaliador marcar algum item como atende parcialmente ou não atende, deve obrigatoriamente indicar os motivos no texto, para que o aluno saiba o porquê da avaliação.