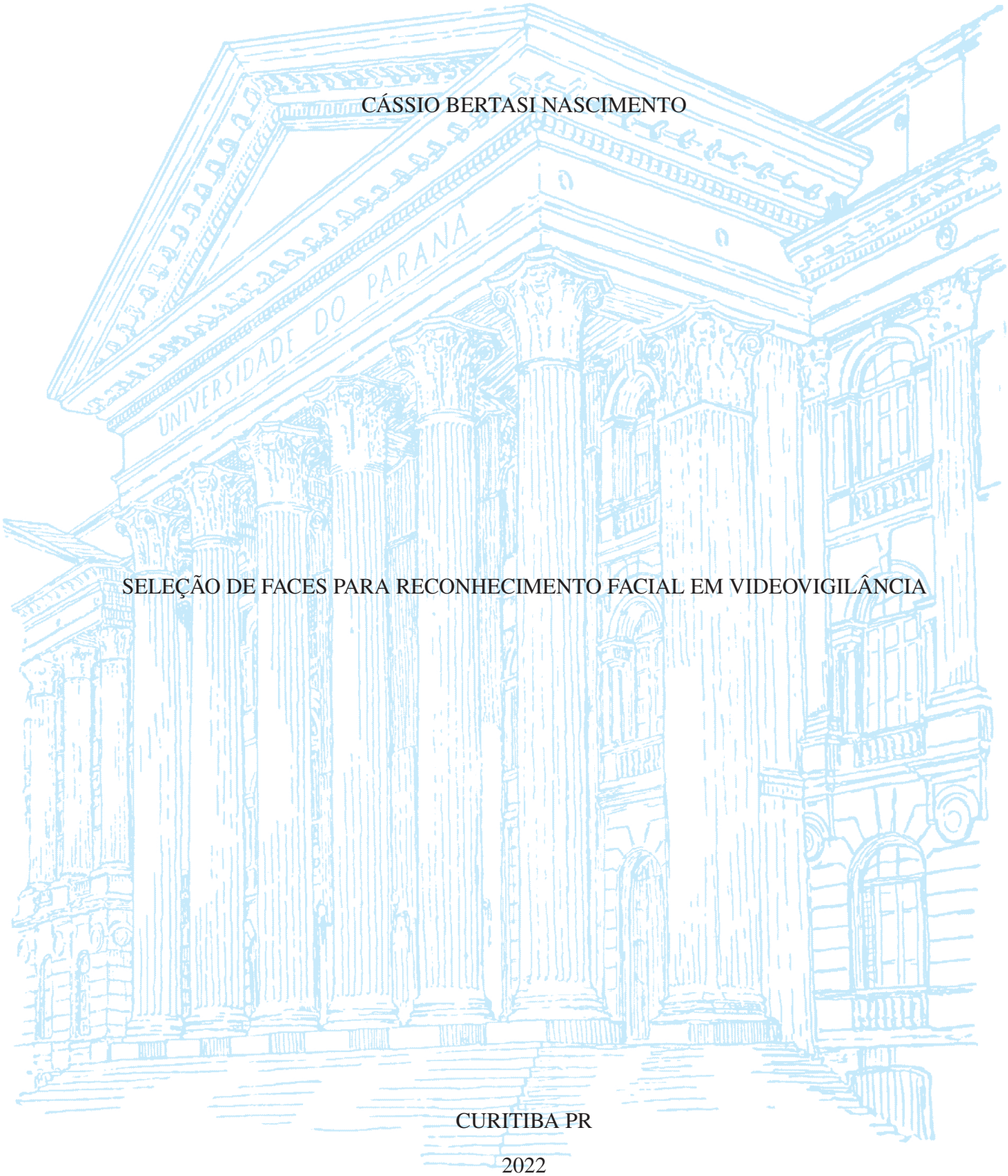
UNIVERSIDADE FEDERAL DO PARANÁ

CÁSSIO BERTASI NASCIMENTO



SELEÇÃO DE FACES PARA RECONHECIMENTO FACIAL EM VIDEOVIGILÂNCIA

CURITIBA PR

2022

CÁSSIO BERTASI NASCIMENTO

SELEÇÃO DE FACES PARA RECONHECIMENTO FACIAL EM VIDEOVIGILÂNCIA

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Informática no Programa de Pós-Graduação em Informática, Setor de Ciências Exatas, da Universidade Federal do Paraná.

Área de concentração: *Ciência da Computação*.

Orientador: Prof. Dr. Luciano Silva.

CURITIBA PR

2022

DADOS INTERNACIONAIS DE CATALOGAÇÃO NA PUBLICAÇÃO (CIP)
UNIVERSIDADE FEDERAL DO PARANÁ
SISTEMA DE BIBLIOTECAS – BIBLIOTECA DE CIÊNCIA E TECNOLOGIA

Nascimento, Cássio Bertasi
Seleção de faces para reconhecimento facial em videovigilância / Cássio Bertasi Nascimento. – Curitiba, 2022.
1 recurso on-line : PDF.

Dissertação (Mestrado) - Universidade Federal do Paraná, Setor de Ciências Exatas, Programa de Pós-Graduação em Informática.

Orientador: Luciano Silva

1. Identificação biométrica. 2. Reconhecimento facial. 3. Inteligência artificial. 4. Videovigilância. 5. Seleção de face. I. Universidade Federal do Paraná. II. Programa de Pós-Graduação em Informática. III. Silva, Luciano. IV. Título.

Bibliotecário: Elias Barbosa da Silva CRB-9/1894

TERMO DE APROVAÇÃO

Os membros da Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação INFORMÁTICA da Universidade Federal do Paraná foram convocados para realizar a arguição da Dissertação de Mestrado de **CÁSSIO BERTASI NASCIMENTO** intitulada: **Seleção de Faces para Reconhecimento Facial em Videovigilância**, sob orientação do Prof. Dr. LUCIANO SILVA, que após terem inquirido o aluno e realizada a avaliação do trabalho, são de parecer pela sua APROVAÇÃO no rito de defesa.
A outorga do título de mestre está sujeita à homologação pelo colegiado, ao atendimento de todas as indicações e correções solicitadas pela banca e ao pleno atendimento das demandas regimentais do Programa de Pós-Graduação.

CURITIBA, 17 de Agosto de 2022.

Assinatura Eletrônica
25/08/2022 18:36:46.0
LUCIANO SILVA
Presidente da Banca Examinadora

Assinatura Eletrônica
23/08/2022 14:02:45.0
HENRIQUE SERGIO GUTIERREZ DA COSTA
Avaliador Externo (HA TECNO PESQUISA E DESENVOLVIMENTO EM TECNOLOGIA LTDA)

Assinatura Eletrônica
31/08/2022 15:57:20.0
OLGA REGINA PEREIRA BELLON
Avaliador Interno (UNIVERSIDADE FEDERAL DO PARANÁ)

Dedico este trabalho aos meus pais,
a quem agradeço a paciência, o su-
porte e os exemplos de perseverança.

AGRADECIMENTOS

Ao Prof. Dr. Luciano Silva pela oportunidade, apoio e paciência na elaboração desta dissertação. Agradeço também aos meus pais, minha parceira e meus amigos por todo o amor, alegria e compreensão ao longo desta jornada.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

RESUMO

A proposta do trabalho é o desenvolvimento de um módulo, dedicado à seleção da imagem da face mais representativa da identidade de pessoas, para fins de reconhecimento facial em sistemas de vigilância por vídeo de ambientes sem restrições. Em um sistema ideal de vigilância por vídeo com reconhecimento facial, uma das etapas fundamentais é a Seleção de Faces. A Seleção de Faces combina detecção, rastreamento e aferimento de qualidade de faces para encontrar, agrupar e filtrar, respectivamente, as faces nas sequências de vídeo de acordo com métricas de qualidade, como por exemplo: orientação do rosto e nitidez da imagem. Objetiva-se que o módulo proposto seja robusto aos desafios presentes nas sequências de vídeo obtidas por sistemas de vigilância, como: múltiplas faces, baixa resolução de captura, iluminação irregular e oclusões. Para a concretização da proposta, trabalhos similares foram estudados, criou-se um *dataset* complementar de vídeos com múltiplas faces anotadas em ambientes não controlados, e o sistema proposto por Barquero et al. (2021) foi designado como *baseline*. Diante de experimentação com diferentes modelos para detecção de faces, utilização da resolução como medida de qualidade, substituição da medida de nitidez e exploração aleatória de parâmetros, obteve-se um aumento de 10.1% na métrica de precisão multi objetos (MOTP) e 9% a mais na métrica de identificação IDF1.

Palavras-chave: Seleção de faces. Rastreamento de faces. Sistema de vigilância.

ABSTRACT

This work's proposal is the development of a module, dedicated to the selection of the most representative face image of people's identities, for the purposes of facial recognition in video surveillance systems in unrestricted environments. In an ideal video surveillance system with facial recognition, one of the fundamental steps is Face Selection. Face Selection combines face detection, tracking and quality assessment to respectively find, group and filter faces in video sequences according to quality metrics such as face orientation and image sharpness. The goal is for the proposed module to be robust to the challenges present in video sequences obtained by surveillance systems, such as: multiple faces, low resolution, irregular lighting and occlusions. To carry out the proposal, similar works were studied, an additional video dataset with multiple annotated faces in uncontrolled environments was created, and the pipeline proposed by (Barquero et al., 2021) was chosen as a baseline. By means of experimentation with different face detection models, face resolution as an added quality measure, replacement of the baseline's sharpness measurement approach and random parameter search, it was achieved an increase of 10.1% in Multiple Object Tracking Precision (MOTP) metric and 9% more in the IDF1 identification metric.

Keywords: Face Selection. Face tracking. Surveillance system.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ACRÔNIMOS

$AP_{.50}$	*Average Precision at IoU = .50*
AQF	Aferimento de Qualidade de Faces
BOR	Borrões
CAMSHIFT	*Continuously Adaptive Mean Shift*
CCTV	*Closed Circuit Television*
CNN	*Convolutional Neural Network*
DTR	Detecção-Rastreio-Reconhecimento
FBTR	*Face-Based Tracklet Reconnection*
FN	Falsos Negativos
FP	Falsos Positivos
FPS	*Frames Per Second*
FV	Fora de Visão
GPU	*Graphics Processing Unit*
HSV	*Hue Saturation Value*
ID	*Identificador*
IDS	*ID-Switches*
IoU	*Intersection over Union*
MF	Múltiplas Faces
MR	Movimentos Rápidos
MOT	*Multiple Object Tracking*
MOTA	*Multiple Object Tracking Accuracy*
MOTP	*Multiple Object Tracking Precision*
MPN	*Message Passing Network*
MSCL	*Multiple Scale Convolutional Layers*
MT	*Mostly Tracked*
OCL	Oclusão
RDCL	*Rapidly Digested Convolutional Layers*
RFP	Rotação Fora do Plano
RGB	*Red Green Blue*
SSD	*Single Shot Detectors*

LISTA DE SÍMBOLOS

c_t	número de correspondências no quadro de tempo *t*
d_E	confiança de detecção para faces de registro
d_t^i	distância entre cada associação alvo-hipótese *i* no quadro de tempo *t*
d_V	confiança de detecção para faces verificáveis
fp_t	número de falsos positivos no quadro de tempo *t*
g_t	quantia de faces anotadas no quadro de tempo *t*
$IDFN$	falsos negativos para métrica IDF1
$IDFP$	falsos positivos para métrica IDF1
$IDTP$	verdadeiros positivos para métrica IDF1
λ_{det}	confiança de detecção mínima para contabilizar uma face
λ_{FBTR}	limiar mínimo de similaridade para reconexão de *tracklets*
mme_t	número de erros de correspondência no quadro de tempo *t*
m_t	número de falso-negativos no quadro de tempo *t*
n_E	limiar de nitidez para faces de registro
n_V	limiar de nitidez para faces verificáveis
Res_E	resolução mínima para faces de registro
Res_V	resolução mínima para faces verificáveis
T	limiar de Intersecção sob União para consideração de erro
T_{max}	número máximo de quadros que um *tracklet* pode estar desaparecido

SUMÁRIO

1 INTRODUÇÃO

Sistemas de vigilância por câmeras de vídeo (CCTV) exercem um papel importante na proteção de propriedades e pessoas, sendo um método situacional de prevenção de crime, que envolve a manipulação do ambiente de forma a reduzir as oportunidades de ações criminosas e aumentar a percepção de risco dos ofensores (Clarke, 1983). Agem como um deterrente de possíveis ações ilegais e fonte de evidência visual – como a identidade do culpado – para uso investigativo, de forma que, a disponibilidade de gravações oriundas de sistemas CCTV foi relacionada com aumentos substanciais na probabilidade da maioria dos crimes serem resolvidos (Ashby, 2017).

Entretanto, devido a grande quantidade de informação visual que sistemas CCTV conseguem capturar, torna-se quase impossível para um operador humano acompanhar, interpretar e consultar o volume de dados decorrente, especialmente em locais movimentados. Além disso, esquemas de CCTV que incorporam monitoramento ativo – como notificação baseada em eventos – são, no geral, mais eficazes (Piza et al., 2019). Assim, evidencia-se uma demanda de processamento inteligente das sequências de vídeo, para identificar informações relevantes, como tráfego excessivo de pessoas, identidades procuradas ou detecção de armas de fogo. O que resultaria na automação da tarefa de vigilância ou, no mínimo, em uma ferramenta auxiliar para o operador humano responsável.

Um exemplo de ferramenta voltada ao processamento efetivo dos dados visuais de sistemas CCTV com várias câmeras é o trabalho de Wang et al. (2003), exemplificado pela figura 1.1. Onde identifica-se que podem existir momentos nos quais uma ou mais câmeras não obtém informação relevante, por estarem monitorando ambientes vazios por exemplo. Nesse caso, a proposta é capaz de identificar e quantificar movimentos nas imagens de vídeo, para direcionar a atenção do operador às sequências de quadros (*frames*) com maior grau de importância, onde existem informações mais relevantes, como movimento saliente de pessoas e veículos. Eliminando efetivamente a necessidade de se atentar, processar ou armazenar sequências de vídeo em ambientes inertes.

Nesse contexto, pode-se argumentar que o foco de vigilância por vídeo é muitas vezes voltado ao monitoramento do comportamento humano suspeito, à identificação de atos criminosos e, especialmente, às identidades dos infratores. Para esse último – identidade – a face é um componente biométrico robusto e consistentemente presente em imagens de monitoramento, sendo o objeto de estudo do presente trabalho. Especificamente, o processamento de sequências de vídeo obtidas por sistemas CCTV, de forma a extrair faces que possam ser utilizadas por sistemas de Reconhecimento Facial.

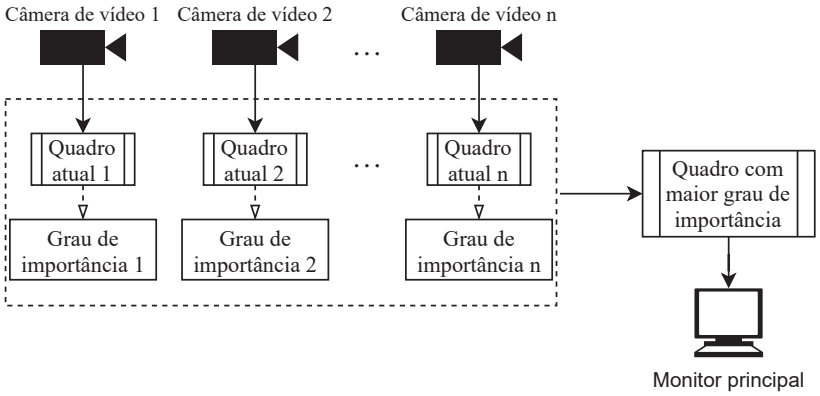


Figura 1.1: Diagrama simplificado da proposta de Wang et al. (2003), que por meio de análise de movimento (amostragem experiencial), seleciona sequências de quadros relevantes em um sistema de vigilância por vídeo para direcionar o foco de atenção.

No entanto, a análise de faces nas sequências de vídeo obtidas em ambientes não controlados, como os vigiados por sistemas CCTV, não é uma tarefa trivial devido a uma gama de desafios relacionados ao ambiente e captura de imagem, bem como ao comportamento das faces.

Quanto aos desafios relacionados ao ambiente e captura de imagem, tem-se: baixas resoluções de vídeo, geralmente utilizadas por sistemas CCTV como forma de evitar a sobrecarga da capacidade de armazenamento; brilho irregular, causado pela presença de sombras ou luz forte que distorcem as cores; e oclusões, causadas tanto pela sobreposição de faces como pela presença de outros objetos.

No que tange ao comportamento das faces, os desafios são: poses irregulares, sendo desejável que a face esteja o mais frontal possível, tanto para a detecção quanto para o reconhecimento; deformação por expressões faciais; e a utilização de adornos, como óculos, bonés e máscaras. A Figura 1.2 demonstra exemplos dos desafios por meio de recortes de imagens obtidas por sistemas CCTV.

Figura 1.2: Exemplos dos desafios encontrados ao analisar faces em sequências de vídeo obtidas em ambientes não controlados, respectivamente: brilho irregular; pose irregular da face e uso de adorno; e faces embaçadas como consequência de baixa resolução.

Em vista dos desafios citados, não é desejável empregar estratégias simplistas, como, realizar reconhecimento facial em todas as faces detectadas ao longo das sequências de vídeo. Estratégias como essa são computacionalmente inviáveis nesse cenário, dada a complexidade da tarefa de reconhecimento, assim como degradam sua acurácia (Nasrollahi e Moeslund, 2011).

Além disso, como outro exemplo de ineficácia, se esta mesma estratégia for utilizada para super resolução – entendida como a tarefa de construir uma imagem de maior resolução, a partir de uma ou mais imagens de interesse – acaba por inserir ruídos no processo de reconstrução, diminuindo a eficiência desses algoritmos (Nasrollahi e Moeslund, 2011). Enfatiza-se então, a necessidade de estratégias mais refinadas para auxiliar os algoritmos de interesse, nesse caso, reconhecimento facial.

Como forma de viabilizar a aplicação de reconhecimento facial para vigilância, propõe-se empregar seleção de faces, aqui definido como o processo de selecionar a imagem que melhor represente o rosto de cada pessoa capturada por uma câmera de vídeo. Seleção de faces é uma tarefa multidisciplinar, sendo alcançada por meio da concatenação de três outras sub tarefas: detecção de faces, rastreamento de faces e Aferimento de Qualidade de Face (AQF).

Para auxiliar na diferenciação e compreensão dos termos utilizados quando se trata do processo de seleção de faces, abaixo são listados e explicados cada um. O processo completo ideal, assim como seu resultado, é exemplificado em seguida pela Figura 1.3.

- Detecção de faces: dada uma imagem, encontrar e encaixar um retângulo envolvente em todas as faces nela presentes. O resultado desta etapa são as coordenadas das faces no plano da imagem;

- Rastreamento de faces: dada a posição inicial de uma, ou mais, face(s) em um quadro pertencente a uma sequência de vídeo, acompanhar a trajetória de cada face, quadro a quadro, mantendo um retângulo envolvente encaixado em suas posições. O resultado desta etapa são trilhas, ou rastros: o conjunto de coordenadas da face rastreada, juntamente de um identificador numérico único para cada trilha;

- Aferimento de qualidade da face: para cada face detectada, avaliar a qualidade da captura por meio da quantização de atributos, como, orientação da face (pose), nitidez da imagem, resolução do recorte da face e identificação da presença de olhos e boca. O resultado desta etapa é um valor numérico para cada face, que representa sua qualidade e pode ser utilizado para ranqueá-las;

- Seleção de faces: é a concatenação de algoritmos de detecção, rastreamento e aferimento de qualidade, para encontrar, separar e ranquear, respectivamente, todas as faces em uma sequência de vídeo. O resultado desta etapa são as melhores faces para cada pessoa encontrada na sequência. Também pode ser encontrada na literatura por outros termos, como, *face logging* (Del Bimbo et al., 2009) e reconhecimento facial em vídeo (Zheng et al., 2020).

Entendidos o funcionamento e resultados de um algoritmo para seleção de faces, a Figura 1.4 demonstra um possível fluxo de informações entre um módulo – parte de um sistema maior – de seleção de faces e um sistema CCTV, de forma a compor um sistema de vigilância inteligente para reconhecimento facial em vídeo.

Na literatura, seleção de faces se apresenta de diferentes formas, geralmente com o propósito de reconhecimento facial em vídeo. Por exemplo, o trabalho de Zheng et al. (2020) apresenta um sistema automático para reconhecimento facial em vídeo, que emprega uma série de Redes Neurais Convolucionais (CNN, do inglês: *Convolutional Neural Networks*) para as etapas de detecção de face, associação de face e reconhecimento facial.

O trabalho de Vignesh et al. (2015) desenvolve um método baseado em CNN para aferir a qualidade das faces, de forma que a rede prediga o quão apropriada é cada imagem, para ser utilizada por um algoritmo de reconhecimento facial. Para testagem da proposta, um fluxo de

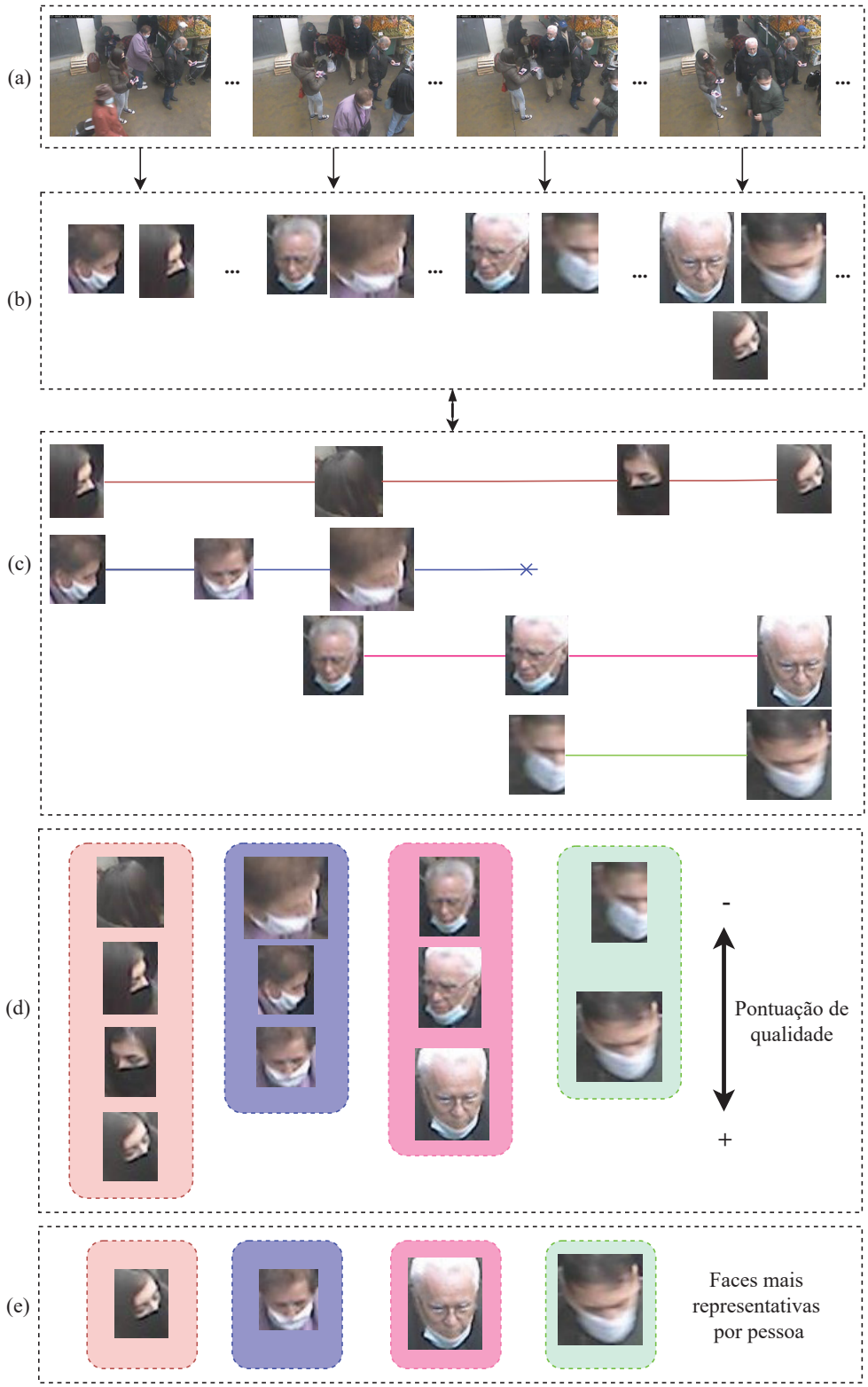


Figura 1.3: Diagrama passo a passo do processo de seleção de faces. (a) Os quadros do vídeo de entrada a ser processado; (b) Faces detectadas por um algoritmo de detecção de faces; (c) As trilhas resultantes do processo de rastreamento de faces, que associa as detecções ao longo do tempo; (d) Faces ordenadas por pontuação de qualidade, resultado do aferimento de qualidade de face; e (e) A melhor imagem que representa a face de cada pessoa.

seleção de faces é desenvolvido indiretamente, onde demonstra-se com sucesso que o processo de seleção aumenta a acurácia de reconhecimento e diminui a complexidade computacional.

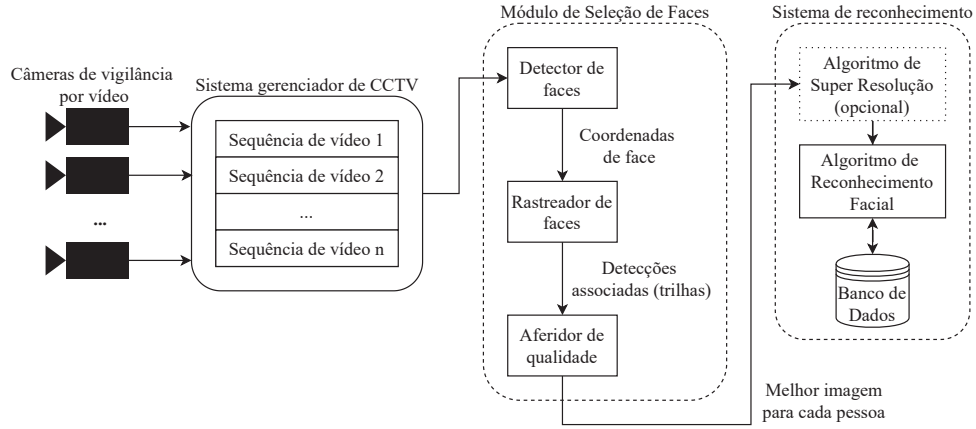


Figura 1.4: Diagrama de fluxo de um sistema de vigilância inteligente, com um módulo proposto para seleção de faces como etapa fundamental para reconhecimento.

Nasrollahi e Moeslund (2011) constatam as dificuldades de se aplicar super resolução em imagens faciais obtidas em cenários não controlados, visto que a utilização de imagens impróprias inserem ruídos no processo de reconstrução. Logo, utilizam seleção de faces para filtrar as faces obtidas. Os resultados constatam que a seleção de imagens apropriadas melhora os resultados do algoritmo de super resolução.

O presente tema é multidisciplinar, relevante e útil para diversas aplicações ao contribuir com a eficiência de processamento de vídeo, aceleração da testagem de algoritmos de super resolução e reconhecimento facial, assim como a automação de sistemas de segurança. Estudá-lo mais a fundo significa contribuir com diversas áreas, tanto com aplicações em cenários não controlados, quanto no desenvolvimento de pesquisas futuras.

1.1 OBJETIVOS

O objetivo geral desta proposta é o desenvolvimento de um método de seleção de faces aplicado ao contexto de sistemas de vigilância por vídeo, para auxiliar na extração automática de identidades pela viabilização da aplicação de reconhecimento facial em grandes quantidades de dados e em ambientes não controlados. Assim como, na criação de uma ferramenta que auxilie o operador humano na tarefa de monitoramento.

Para alcançar o objetivo geral, os passos são separados em objetivos específicos, enumerados abaixo:

1. Estudo sistemático das principais abordagens desenvolvidas para seleção de faces presentes na literatura;

2. Estudo dos algoritmos utilizados em cada etapa da seleção de faces – detecção, rastreamento e aferimento – de forma a combiná-los e avaliá-los para serem utilizados no sistema de seleção de faces desenvolvido neste trabalho;

3. Levantamento de conjuntos de dados (*datasets*) para avaliação do método proposto, assim como criação de um *dataset* próprio com características representativas de ambientes não controlados;

4. Validação da proposta com relação às suas contribuições para realização de reconhecimento facial em sequências obtidas por sistemas de vigilância por vídeo em ambientes sem restrições;

1.2 ESTRUTURA DO DOCUMENTO

O presente trabalho está organizado nos capítulos a seguir: o Capítulo 2 apresenta uma fundamentação teórica dos conceitos utilizados frequentemente neste trabalho; o Capítulo 3 discorre sobre diversos algoritmos de detecção de faces, rastreamento de faces e objetos, compila os resultados das consultas em bases de pesquisa pelos trabalhos representativos de seleção de faces, analisa os métodos utilizados, os resultados e identifica as possíveis lacunas a serem preenchidas; o Capítulo 4 apresenta o sistema de seleção de faces escolhido como *baseline*, assim como as mudanças propostas em sua implementação e as justificativas destas propostas; o Capítulo 5 apresenta o conjunto de dados utilizado, o conjunto de dados construído como suplementação, os testes neles realizados e os resultados de várias métricas para avaliação de métodos de rastreamento de múltiplos objetos em vídeo.

2 CONCEITOS BÁSICOS

2.1 REDES NEURAIS CONVOLUCIONAIS

O aprendizado profundo é uma subárea do aprendizado de máquina, e é representado pelo uso de redes neurais multicamada para a criação de modelos capazes de realizar predições a partir do treino em vastas quantias de dados (Alzubaidi et al., 2021). Popularmente conhecido como *Deep Learning*, o aprendizado profundo têm demonstrado taxas de erro cada vez menores e dominado o estado do conhecimento nas tarefas de reconhecimento visual, detecção de objetos e reconhecimento de fala (Goodfellow et al., 2016). Esse avanço se deve, principalmente, a grande quantia de dados disponíveis aliada a melhoras constantes de *hardware*, representadas pelo uso eficiente de arquiteturas de processamento paralelo, para execução dos algoritmos de *Deep Learning*.

As redes neurais profundas surgiram de forma bioinspirada no cérebro biológico, baseada no conceito de várias unidades computacionais (neurônios), que tornam-se inteligentes apenas por meio das interações umas com as outras (Goodfellow et al., 2016). O mesmo pode ser dito das CNNs, que imitam o córtex visual e conseguem extrair informações em diferentes níveis de abstração ao longo de suas camadas.

Na área de visão computacional, as CNNs são as mais eficientes e vastamente utilizadas para tarefas como detecção de faces (Zhang et al., 2017), reconhecimento facial (Deng et al., 2019) e classificação de imagens (Krizhevsky et al., 2017).

Atualmente existem várias arquiteturas e diferentes implementações de CNNs, porém, de forma simplificada, CNN é uma rede neural que utiliza convolução no lugar de multiplicações de matrizes em, ao menos, uma de suas camadas (Goodfellow et al., 2016). Sua composição mais básica pode ser descrita em três camadas: camada de convolução, camada de *pooling* e camada totalmente conectada. A Figura 2.1, disponível no trabalho de Alzubaidi et al. (2021), exemplifica o fluxo de uma CNN desde a imagem de entrada, passando pelas camadas anteriormente citadas, até a camada de saída que exprime o resultado do processo de classificação.

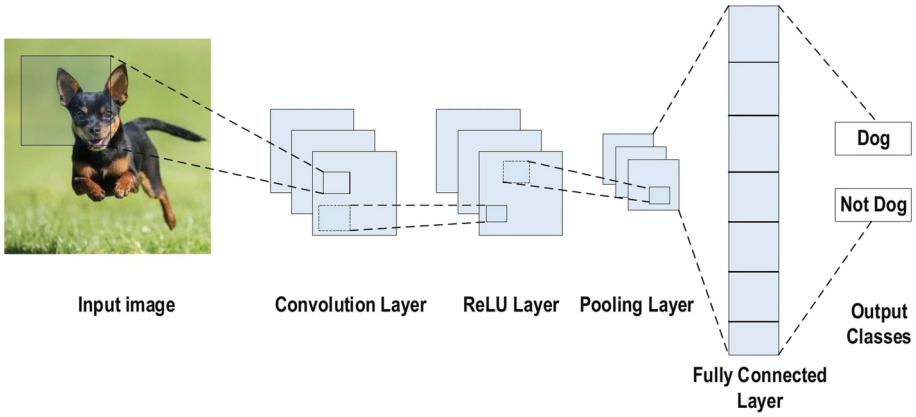


Figura 2.1: Exemplo da composição de uma CNN em suas camadas mais comuns, para classificação de imagens. Disponível em Alzubaidi et al. (2021).

A primeira e mais importante é a camada de convolução, composta de filtros convolucionais (*kernels*), cada qual uma matriz composta de pesos. Os pesos de um *kernel* podem ser inicializados aleatoriamente e são atualizados ao longo do treino para extraírem atributos

discriminativos da imagem de entrada. A operação de convolução ocorre com o deslizar do filtro ao longo da imagem, da esquerda para a direita, em intervalos padronizados. Para cada intervalo de deslize (*stride*) do filtro, calcula-se o produto escalar entre o *kernel* e a região da imagem coberta por ele, que consiste na multiplicação membro a membro e soma dos produtos em um único valor escalar (Alzubaidi et al., 2021). A operação se repete até atingir o canto inferior direito da imagem de entrada. O resultado deste processo é uma matriz menor, denominada mapa de atributos. A Figura 2.2 apresenta um exemplo dos cálculos de uma operação de convolução.

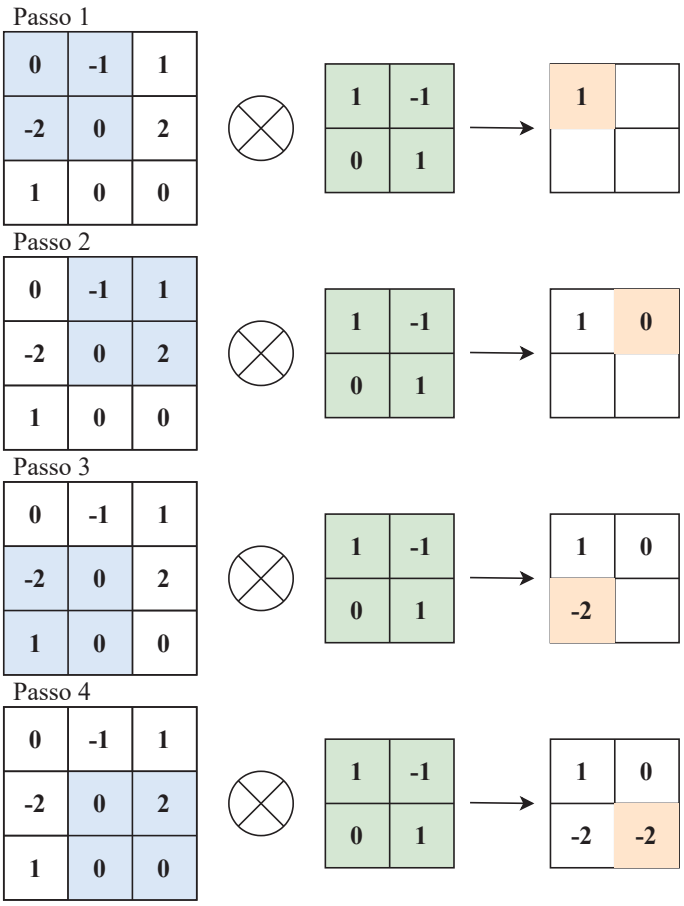


Figura 2.2: Exemplo dos cálculos efetuados em cada passo de uma convolução, com *stride* de 1. A região azul é a posição do *kernel*, em verde o *kernel* e em laranja o resultado da operação de convolução naquele passo. O resultado do exemplo é um mapa de atributos de dimensões 2x2.

Apesar das operações de convolução naturalmente diminuírem o tamanho da entrada, muitas vezes é recomendado a utilização de camadas de *pooling*, que reduzem ainda mais os mapas de atributo, enquanto concorrentemente mantém a maioria da informação espacial dominante (Alzubaidi et al., 2021). Existem várias operações de *pooling*, onde as mais utilizadas são as de *pooling* mínimo, *pooling* máximo, *pooling* médio e *pooling* médio global. Por exemplo, o *pooling* máximo resulta no valor máximo correspondente à porção da imagem coberta pelo *kernel* de *pooling*, de forma análoga o *pooling* mínimo resulta no menor valor da região. A Figura 2.3 demonstra o resultado de diferentes operações de *pooling* em uma imagem de exemplo.

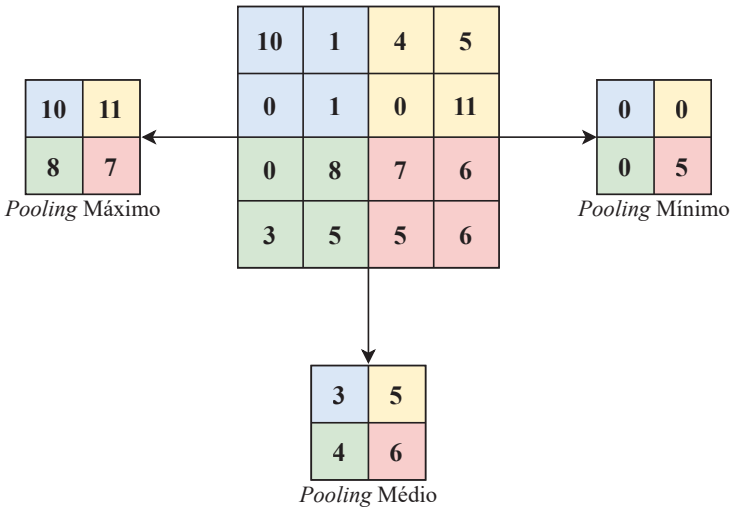


Figura 2.3: Exemplo dos cálculos de *pooling* máximo, mínimo e médio em uma região de dimensão 4x4.

Para realizar a classificação é comum que as últimas camadas sejam totalmente conectadas, isto é, todos os neurônios desta camada estão conectados com os neurônios da camada anterior. A entrada da camada totalmente conectada é o resultado do redimensionamento – transformação de uma matriz n-dimensional em um único vetor com uma dimensão – dos mapas de ativação das camadas de convolução anteriores. A utilização deste tipo de camada permite que a rede aprenda combinações não lineares dos atributos e abstrações obtidas pelas camadas de convolução. A saída da camada totalmente conectada é a saída final da CNN.

Uma arquitetura de CNN bem estabelecida na literatura é a VGG16 (Simonyan e Zisserman, 2015). Sua entrada é uma imagem RGB de tamanho fixo 224 x 224 *pixels*, seguida de 13 camadas de convolução e 3 totalmente conectadas. Os filtros utilizados nas camadas de convolução são de dimensão 3 x 3 e eram menores que o de arquiteturas anteriores, porém demonstraram resultados similares a filtros maiores enquanto aproveitou-se do tamanho reduzido de parâmetros, tornando-a ligeiramente mais eficiente. No geral a arquitetura obteve resultados significativos nas tarefas de localização e classificação, mas sua quantia excessiva de aproximadamente 140 milhões de parâmetros a torna uma rede computacionalmente cara (Alzubaidi et al., 2021). A Tabela 2.1 detalha a arquitetura da VGG16 por camada.

Tabela 2.1: Arquitetura detalhada da rede VGG16, disponível em Simonyan e Zisserman (2015), página 3.

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 **LRN**	conv3-64 **conv3-64**	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 **conv3-128**	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 **conv1-256**	conv3-256 conv3-256 **conv3-256**	conv3-256 conv3-256 conv3-256 **conv3-256**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 **conv1-512**	conv3-512 conv3-512 **conv3-512**	conv3-512 conv3-512 conv3-512 **conv3-512**
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

2.2 RASTREAMENTO DE MÚLTIPLOS OBJETOS

Esta seção se atenta a detalhar o que é a tarefa de Rastreamento de Múltiplos Objetos (MOT, do inglês: *Multiple Object Tracking*), seus conceitos importantes e termos comumente utilizados para descrever diferentes abordagens. Destaca-se que o tema desta dissertação trata especificamente do rastreamento de múltiplas faces, que é uma subtarefa de MOT, assim, versar sobre MOT de forma ampla auxilia na abstração dos desafios de rastreamento e inclui todos os conceitos necessários para sua compreensão.

MOT é a tarefa de localizar, identificar e obter trajetórias para cada objeto de interesse em um vídeo que pode conter múltiplos objetos da mesma classe, ou de classes diferentes. Alguns exemplos de objetos comumente alvos de rastreamento são pessoas, veículos, jogadores em campo e animais (Luo et al., 2021).

Com o avanço das CNNs nas tarefas de visão computacional, em especial na área já bem desenvolvida de detecção de objetos, é compreensível que o próximo passo para se obter mais informações de vídeos seja o rastreamento (Park et al., 2021), fazendo com que, recentemente, MOT tenha recebido cada vez mais atenção graças ao seu potencial comercial e acadêmico.

Visto que a tarefa de MOT se propõe a rastrear múltiplos objetos concorrentemente, alguns desafios são mais consideráveis nesta modalidade de rastreamento, especificamente, oclusão e troca de identificador (ID). Oclusão ocorre quando um objeto “A” tem sua forma encoberta, parcialmente ou totalmente, por outro objeto “B” no plano da imagem. Neste caso é comum dizer que “A” foi ocluso por “B”. Quando oclusão ocorre é difícil predizer a posição do objeto ocluso, muitas vezes sendo necessário algum método de reconexão utilizando a informação

do objeto em quadros anteriores (Park et al., 2021), como características extraídas por uma CNN, modelos de cor, ou predição de movimento baseado em velocidade.

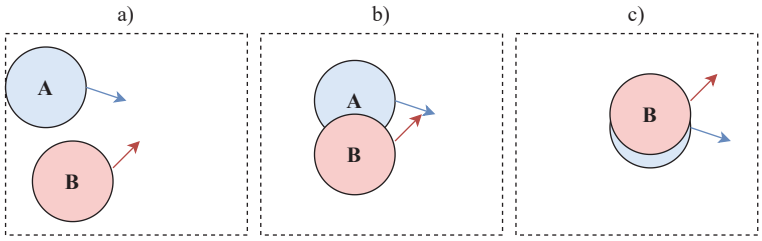


Figura 2.4: Exemplo de oclusão conforme objetos se movem na imagem. a) A e B estão visíveis, mas movendo-se em direção um ao outro; b) A está levemente ocluso por B; c) A está fortemente ocluso por B.

No entanto, quando a predição ou associação do objeto ocluso falha, ocorre a troca de ID. Para entender a troca de ID, primeiramente introduz-se o conceito de *tracklet*: associação de detecções pertencentes ao mesmo objeto, em um curto período de tempo, que representam trajetórias curtas e confiáveis (Brasó e Leal-Taixé, 2020). Na Figura 2.5 exemplifica-se a troca de ID quando o objeto rastreado move-se de forma abrupta, distanciando-se da predição feita pelo rastreador para seu *tracklet* correspondente. Nesse caso, o *tracklet* é encerrado e assim que o objeto for detectado novamente, um novo *tracklet* é iniciado com um novo ID.

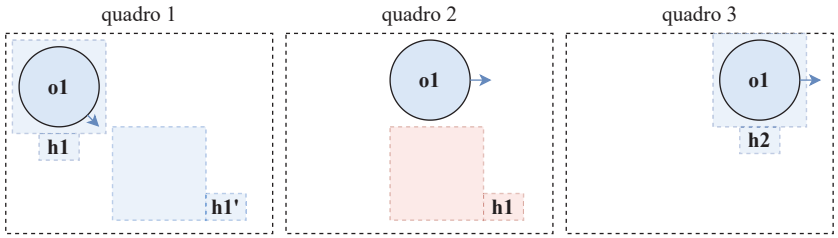


Figura 2.5: Exemplo de troca de ID quando o rastreador perde a posição do objeto. No quadro 1 o objeto o1 é rastreado pelo *tracklet* h1 e tem sua posição predita em h1'; no quadro 2 o objeto o1 muda de direção, de forma que h1 não consegue continuar rastreando sua posição; no quadro 3 o *tracklet* h1 é finalizado e o objeto o1 é detectado novamente, porém com um novo ID e agora rastreado pelo novo *tracklet* h2.

Ademais, a troca de ID pode ocorrer quando dois ou mais objetos, rastreados por *tracklets* diferentes, cruzam-se. Dependendo do modo de associação e da similaridade entre os objetos, é possível que seus *tracklets* sejam trocados. Diferente do primeiro exemplo de troca de ID, neste caso é mais difícil reconectar os *tracklets* a seus objetos originais. A Figura 2.6 demonstra um exemplo desta falha de rastreamento.

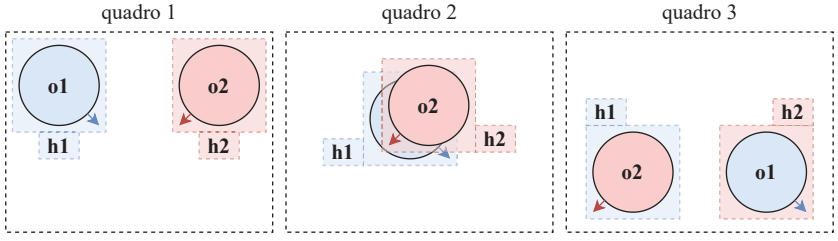


Figura 2.6: Exemplo de troca de ID quando dois objetos, o1 e o2, respectivamente rastreados pelas hipóteses h1 e h2, cruzam-se. No quadro 3 os *tracklets* trocam de objeto ao invés de serem desativados.

Existem várias formas de realizar o rastreamento, um dos paradigmas mais dominante atualmente na literatura é o de rastreamento por detecção (*tracking-by-detection*). Neste paradigma, ao invés de tentar predizer a posição do objeto no quadro seguinte, um detector é utilizado para obter as posições de todos os objetos da cena, e em seguida o rastreamento é delegado a uma estratégia de associação de dados, que tenta conectar todas as detecções pertencentes ao mesmo objeto (Luo et al., 2021).

Uma das formas de realizar *tracking-by-detection* é associar sequencialmente as posições dos objetos detectados em um quadro, com as posições detectadas no quadro seguinte por meio dos valores de Interseção sob União (IoU, do inglês: *Intersection over Union*). IoU é uma métrica de distância entre dois retângulos envolventes, onde o valor da área de interseção é dividido pela área de união dos dois retângulos, a Figura 2.7 apresenta uma visualização da IoU. Essa métrica também é comumente utilizada para medir outros valores de qualidade como precisão e acurácia de um detector ou rastreador (Bernardin e Stiefelhagen, 2008) (a utilização da IoU para métricas de qualidade é tratada no Capítulo 5).

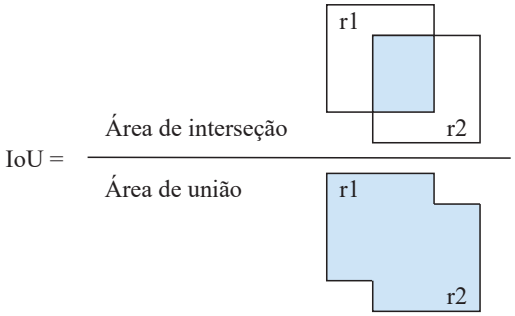


Figura 2.7: Cálculo do valor de IoU entre dois retângulos envolventes, r1 e r2.

Um exemplo de *tracking-by-detection* é demonstrado no trabalho de Bewley et al. (2016), onde as detecções são associadas quadro a quadro por meio do método Húngaro (Kuhn, 1955) aplicado nos valores de IoU dos retângulos envolventes. O método Húngaro é uma forma de resolver o problema de associação, onde uma matriz de custos é gerada com os valores de IoU entre as detecções de quadros consecutivos. Em seguida, o método aloca os pares de detecção com o menor custo possível (Park et al., 2021). A Figura 2.8 representa uma visualização do método Húngaro aplicado nas detecções de dois objetos em dois quadros consecutivos.

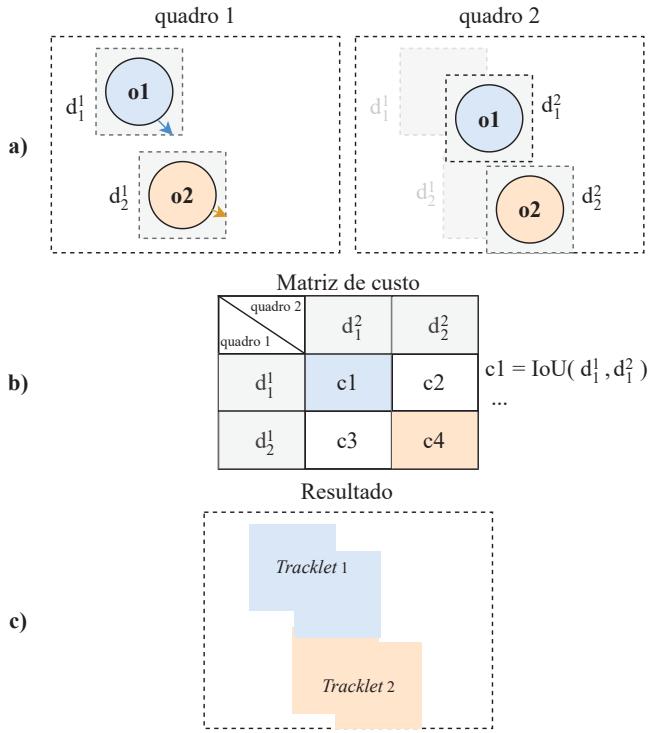


Figura 2.8: a) Dois objetos são detectados em dois quadros consecutivos; b) Matriz de custo gerada pelo cálculo de IoU entre todas as detecções do quadro 1 e todas as detecções do quadro 2, c1 e c4 representam a melhor associação resultante do método Húngaro; c) Resultado final da associação das detecções, que origina dois *tracklets*.

O trabalho de Bewley et al. (2016) demonstrou que é possível atingir resultados de rastreamento comparáveis ao estado da arte utilizando-se de métodos de associação, como IoU, e que a dependência do detector neste contexto é considerável. No geral, o uso de um detector robusto melhora consideravelmente os resultados de rastreamento, porém, é necessário atentar-se aos números de falso-negativos e falso-positivos, visto que esses erros geram trocas de ID frequentemente e degradam os resultados do rastreador (Park et al., 2021).

3 TRABALHOS RELACIONADOS

O desafio de rastreamento e seleção de múltiplas faces é complexo porque envolve uma série de algoritmos na confecção de um fluxo que seja capaz de detectar, associar, rastrear e avaliar múltiplas faces em vídeo. Entretanto, ao longo da literatura existem mais trabalhos voltados a detecção de faces e MOT (geralmente os objetos alvos de rastreio são pessoas, ao invés de faces). Explorar o rastreamento de múltiplas faces também requer estudar várias técnicas separadamente.

A seção 3.1 disserta sobre métodos para detecção de faces. A escolha do detector é o primeiro passo da implementação de um fluxo para rastreamento bem sucedido, visto que a qualidade das detecções influencia intensamente métodos de rastreio por detecção.

No que tange rastreamento, a seção 3.2 apresenta métodos para várias modalidades de rastreamento, como o de múltiplas pessoas, múltiplos objetos e para face singular (quando há apenas uma face na imagem). São escassas as bases de dados e trabalhos voltados ao rastreamento de múltiplas faces em vídeos de vigilância (Barquero et al., 2021), por conseguinte, é relevante estudar os métodos propostos para diferentes tarefas, em busca de derivar estratégias de associação que possam ser testadas ou modificadas para rastrear múltiplas faces, como realizado por Liu e Cai (2006) onde um rastreador de objetos genérico é modificado para rastrear múltiplas faces em vídeo.

É importante constatar que quando métricas de velocidade são apresentadas neste capítulo, tal qual quadros por segundo (FPS, do inglês: *Frames Per Second*), estas não representam uma avaliação unificada de todos os métodos em um único sistema e apenas denotam os resultados expostos em seus trabalhos originais.

Por último, a seção 3.3 elenca trabalhos voltados ao rastreamento de faces em cenários de videovigilância, separa suas propostas nas três etapas necessárias para seleção de faces – detecção, rastreamento e aferimento de qualidade – e discorre sobre o que se pode verificar dos algoritmos utilizados em cada etapa. Ao fim deste capítulo, pretende-se que o leitor esteja informado do estado do conhecimento sobre seleção de faces, assim como as questões passiveis de estudo que justificam a presente dissertação.

3.1 DETECÇÃO DE FACES

A detecção de faces é o primeiro passo da solução para uma série de problemas na área de visão computacional. Alinhamento de face, reconhecimento facial, reconstrução e rastreio são exemplos de problemas que dependem de um modelo de detecção robusto, capaz de oferecer resultados mesmo diante de desafios como: oclusão, variabilidade intra-classe, iluminação, baixa resolução, deformação e expressão.

Detecção de faces se deu marcadamente em dois períodos. O primeiro foi marcado por classificadores cujos atributos, restrições e modelos eram muitas vezes desenvolvidos a mão e aplicados como uma janela deslizante ao longo da imagem. Posteriormente, métodos que utilizam aprendizado de máquina demonstraram resultados cada vez melhores. Possivelmente o trabalho mais conhecido dessa categoria é o trabalho de Viola e Jones (2004) que treina um detector em cascata utilizando características de Haar e aprendizado Adaboost.

Outros métodos tradicionais, além do paradigma cascata, são os modelos de partes deformáveis (Felzenszwalb et al., 2010; Yan et al., 2014; Mathias et al., 2014). Esta abordagem busca resolver o problema de variação intra-classe característicos da tarefa de detecção, por meio

do treino de modelos multiescala deformáveis que representam as várias partes e visões possíveis do objeto alvo.

Atualmente, com o advento das redes neurais profundas e seu desempenho elevado nas tarefas de visão computacional, os modelos de detecção são em grande parte resultados de experimentação com CNNs. Os trabalhos iniciais se davam através do treino de redes com paradigmas originalmente desenvolvidos para a tarefa de detecção de objetos, como *Faster-RCNN* e *Single Shot Detectors* (SSD).

A proposta apresentada por Zhu et al. (2017) trabalha com o *framework Faster-RCNN* e apresenta a *Contextual Multi-Scale Region-based CNN*, que implementa informação contextual corporal para a detecção de faces, inspirado na intuição do sistema visual humano.

No que diz respeito aos SSD, o modelo *FaceBoxes* desenvolvido por Zhang et al. (2017), trata o fato da eficiência computacional das CNNs ser limitada em aparelhos que não dispõem de hardware altamente paralelizável, como, por exemplo, uma unidade de processamento gráfico (GPU, do inglês: *Graphics Processing Unit*). Assim voltam-se os esforços à criação de um modelo que apresente um processamento mais rápido em arquiteturas com baixo número de núcleos.

A arquitetura de *FaceBoxes* consiste dos *Rapidly Digested Convolutional Layers* (RDCL) e dos *Multiple Scale Convolutional Layers* (MSCL). Os RDCL rapidamente encolhem o tamanho da imagem de entrada ao definir uma série de passos (*strides*) grandes e os MSCL são responsáveis por tratar a alta variação de tamanho das faces presentes nas imagens. Para imagens de resolução VGA, *FaceBoxes* executa a 20 FPS em um único núcleo de um processador Intel Xeon E5-2660v3@2.60GHz e a 125 FPS quando acelerado por uma GPU Titan X Pascal.

Comumente, quando se trata da velocidade de processamento de CNNs em vídeo, a denominação "tempo real" informa que o modelo processa, no mínimo, a 30 FPS. Outra denominação é "tempo quase real" (*semi real-time*), neste caso a velocidade mínima é de 15 FPS.

Deng et al. (2020) baseiam sua proposta em uma configuração de pirâmide de atributos, como a apresentada por Lin et al. (2017). Ademais, apresentam uma função de perda multitarefa (*multi-task loss*) para regressão de pontos, que é possível devido a anotações extras de cinco pontos fiduciais em 103100 faces do *dataset WIDER FACE* (Yang et al., 2016), às quais atribui-se uma melhora na capacidade de detecção nas imagens do subconjunto difícil deste mesmo *dataset*.

Assim, a função de perda multitarefa a ser minimizada contém, além dos termos usuais de minimização do erro de classificação e regressão dos retângulos envolventes (*bounding boxes*), outros dois termos: regressão de pontos fiduciais das faces e regressão densa de pixels. O modelo final, *RetinaFace*, é um detector de faces em tempo real para cenários densos e não controlados.

No trabalho de Zhang et al. (2020) argumenta-se que apesar dos avanços significativos na tarefa de detecção resultante das CNNs, dois aspectos são passíveis de melhora: regressão dos retângulos envolventes e eficiência de classificação, ou seja, diminuição de falso-positivos. Apresenta-se o detector *RefineFace*, composto de cinco módulos que auxiliam na detecção de faces em várias escalas e regressão robusta. O método também apresenta velocidade em tempo real.

3.2 RASTREAMENTO

Voltado à criação de uma interface de usuário perceptiva para computadores, Bradski (1998) utiliza-se do algoritmo não paramétrico para escalada de gradientes *Mean Shift*, alterando-o para tratar das mudanças nas distribuições de probabilidade que ocorrem em vídeo, resultando no algoritmo *Continuously Adaptive Mean shift* (CAMSHIFT), um rastreador de modelo de cor que redimensiona sua forma a melhor encaixar o retângulo envolvente na face do alvo.

O CAMSHIFT rastreia o objeto por intermédio da distribuição de probabilidade das suas cores. Essa distribuição é obtida através do encaixe de um retângulo envolvente ao redor do objeto no plano da imagem e em seguida, no espaço de cores composto dos canais Matiz, Saturação e Valor (HSV, do inglês: *Hue, Saturation and Value*), extrai-se um histograma unidimensional do canal Matiz correspondente a área de interesse na imagem. O histograma é então salvo e utilizado como uma tabela de consulta para mapear os *pixels* do vídeo para uma probabilidade daquele *pixel* corresponder a superfície do alvo.

No contexto de rastreadores baseados em cor, Gejguš e Šperka (2003) realizam a segmentação dos *pixels* que correspondem a cores de pele, por meio da normalização das cores cromáticas vermelha e verde do espaço de cores RGB e com o auxilio do modelo estocástico de cor de pele descrito por Jie Yang e Waibel (1996). Uma elipse é desenhada e encaixada ao redor da segmentação pelo processo apresentado por Sobottka e Pitas (1996) e sua probabilidade de representar uma face é verificada pela razão entre altura e largura da região da elipse. Por fim, o rastreador KLT (Lucas e Kanade, 1981) é utilizado na imagem segmentada. O método apresentou, à época, processamento em tempo quase real.

Entretanto, por dependerem de um modelo de cor, tanto CAMSHIFT quanto o KLT deterioram-se com a alteração dos valores de brilho da imagem, não realizando rastreio em cenas muito escuras, claras ou com variações na iluminação. Esse fato deve ser considerado quando se trata de vigilância em ambientes não controlados. Além disso, nenhuma dessas abordagens cita a manutenção de um identificador único para cada objeto, visto que sua utilização inicial era para rastreio de um único alvo.

Apesar da sua vulnerabilidade quanto à iluminação, devido a sua velocidade de processamento, o CAMSHIFT foi utilizado em projetos posteriores. Liu e Cai (2006) modificam o CAMSHIFT original, com a inicialização de uma janela de busca na face do alvo e uma janela acessória logo abaixo desta, resultando no *Dual Searching Window Based Face-Tracking Algorithm*. Para cada quadro seguinte são realizadas operações morfológicas no canal matiz da janela alvo para eliminar cabelo, sobrancelhas, olhos e boca, de forma a aumentar a área de cor de pele. Esse plano frontal é então separado do fundo para evitar interferências de cores similares.

Utilizando das alterações anteriores, os mesmos autores extendem essa abordagem e apresentam o *CAMSHIFT based multiple faces match tracking* para rastreio de múltiplas faces (Qiang Liu et al., 2007). Os resultados são avaliados em quatro sequências de vídeo de resoluções 320x240 e 640x480, que apresentam de 2 a 4 faces frontais simultaneamente no mesmo quadro.

Em outro exemplo de uso do CAMSHIFT, Vadakkepat et al. (2008), relatam o desafio da construção de um módulo para auxiliar um robô a detectar e seguir um ser humano. Realiza-se a detecção através de um modelo de cor de pele, desenvolvido com auxílio de uma base de 100 imagens de pessoas de diferentes grupos étnicos, e de operações morfológicas para segmentação da região do rosto. Onde após a detecção, selecionou-se o CAMSHIFT para rastreio.

Bewley et al. (2016) , desenvolvem um rastreador multiobjeto focado em simplicidade e eficiência para aplicações em tempo real. Explorou-se os avanços em detecção de objeto providos pelas CNNs, e optou-se por ignorar problemas decorrentes de oclusão, visto que o tratamento explícito desses erros adiciona complexidade indesejada. O método resultante – SORT – utiliza-se da *Faster Region* CNN (Ren et al., 2017).

SORT se baseia na descoberta de que a qualidade das detecções impactam diretamente na performance do rastreador e que, com um detector robusto, é possível atingir resultados similares ao de rastreadores mais complexos, por meio apenas de métodos de associação quadro a quadro. Nessa abordagem a tarefa de encontrar o objeto nos quadros é delegada a um detector, com o rastreador sendo responsável apenas por conectar a identidade de uma detecção em um quadro i com o mesmo alvo detectado no quadro subsequente $i + 1$.

Nesse caso, para propagar a identidade de um alvo para o quadro seguinte, SORT utiliza-se do Filtro de Kalman quando há uma caixa delimitadora associada ao alvo, ou um modelo de velocidade constante caso contrário. O passo de associação utiliza IoU como medida de distância e o Método Húngaro de combinação (Kuhn, 1955) para conectar cada detecção a uma trilha existente. O resultado é um componente de rastreio em tempo real e métricas comparáveis ao de rastreadores com modelos de associação mais complexos.

Bergmann et al. (2019) argumenta que um detector pode ser convertido a um rastreador através da utilização dos seus algoritmos já presentes de regressão e refinamento de retângulos envolventes. O método proposto, *Tracktor*, não requer treino específico em conjuntos de rastreio e executa de forma *online*. Demonstrou resultados de Acurácia de Rastreamento Multiobjeto (MOTA, do inglês: Multiple Object Tracking Accuracy) iguais a 44.1%, 54.4% e 53.5% nas *benchmarks* 2D MOT 2015 (Leal-Taixé et al., 2015), MOT16 e MOT17 (Milan et al., 2016), respectivamente. As métricas de desempenho, como a MOTA, são tratadas detalhadamente na Seção 5.1.

Similarmente ao *Tracktor* que realiza detecção e rastreio de formas conjuntas, *CenterTrack* (Zhou et al., 2020) também condiciona a tarefa de rastreio a dois quadros consecutivos, visa utilizar a robustez dos métodos de CNN e diminui o tempo de inferência por escolher não tratar situações de erro explicitamente.

CenterTrack (Zhou et al., 2020) aplica um modelo de detecção em um par de quadros consecutivos juntamente a um mapa de calor resultante de detecções prévias. Os objetos alvo são rastreado como pontos, permitindo que a saída também proporcione um vetor de deslocamento, que associado a um algoritmo de correspondência é o suficiente para a manutenção da identidade do alvo. O resultado é um rastreador *online*, que reportou execução em tempo quase real e uma MOTA igual a 67.8% no *dataset* MOT17 (Milan et al., 2016).

Recentemente, *tracking-by-detection* destacou-se como um dos paradigmas mais utilizados na maioria das *benchmarks*. Esse paradigma trabalha o desafio de MOT em dois passos: (I) Realiza-se a detecção, geralmente através de um classificador como uma CNN; e (II) conecta-se as detecções ao longo do tempo. Geralmente tratado como um problema de particionamento de grafos.

O trabalho de Arachchilage e Izquierdo (2020) emprega *tracking-by-detection* por meio de agrupamento. O método *Adaptive Aggregation based Agglomerative Clustering*, utiliza CNNs para a detecção, extração e codificação dos atributos de cada imagem. O processo de codificação auxilia na representação utilizada para agrupamento e na diminuição do vasto espaço de busca que resultaria das informações redundantes de um vídeo.

Em seguida, as detecções são agrupadas em pequenas trilhas referentes ao percurso de um único objeto, de forma que, ao final, são conectadas umas as outras por operações de união (*merge*). Resultados de acurácia de agrupamento do método AAAC são de 99%, 100% e 98.53% MOTA nos *datasets* BBT0101, QMUL e Buffy0502, respectivamente.

Ainda no contexto de *tracking-by-detection*, Brasó e Leal-Taixé (2020) propõem uma rede denominada *Message Passing Network* (MPN) que realiza seu aprendizado diretamente no grafo que representa a tarefa de rastreamento. Seja $G = (V, E)$ um grafo não dirigido, onde cada vértice $i \in V$ representa uma detecção e o conjunto de arestas E é construído de tal forma que todos os pares de detecção em *frames* diferentes estão conectados.

O objetivo da MPN é aprender uma função para propagar as informações embutidas nos vértices e arestas ao longo de G. De tal forma que ao fim das operações de propagação, a variável binária $y_{(i,j)}$ presente em cada aresta seja igual a 1 se os vértices que essa aresta conecta forem: pertencentes a mesma trajetória e temporalmente consecutivos dentro desta trajetória. E 0 caso contrário.

3.3 SELEÇÃO DE FACES

Esta seção discorre sobre os trabalhos constatados, suas características assim como pontos passíveis de aprimoramento. Primeiramente os trabalhos são enumerados e seus temas principais são listados. Depois, elenca-se os os métodos e algoritmos empregados em cada uma das três etapas do processo de seleção de faces. Trata também das semelhanças entre os trabalhos, assim como os conjuntos de dados disponíveis para teste. Por último, pontos passíveis de estudo e aprimoramento são apresentados.

A Tabela 3.1 elenca os trabalhos em ordem anual de publicação e relaciona os temas principais discutidos em cada um. Percebe-se que a tarefa de obter e selecionar as melhores faces por identidade se apresenta sob diferentes perspectivas.

Tabela 3.1: Trabalhos selecionados e seus temas principais, ordenados por ano e enumerados.

N°	Citação	Temas principais
1	Chen et al. (2007)	Pontuação de faces
2	Nasrollahi e Moeslund (2008)	Aferimento de qualidade de faces
3	Del Bimbo et al. (2009)	*Logging* de faces
4	Mau et al. (2010)	Reconhecimento facial e Seleção de faces
5	Nasrollahi e Moeslund (2010b)	Super resolução e *Logging* de faces
6	Nasrollahi e Moeslund (2010a)	Super resolução e *Logging* de faces
7	Nasrollahi e Moeslund (2011)	Super resolução
8	Bagdanov et al. (2012)	*Logging* de faces
9	Kim et al. (2014)	Aferimento de qualidade de faces e Reconhecimento facial
10	De Marsico et al. (2014)	*Logging* de faces
11	Momin e Jere (2015)	*Logging* de faces
12	Vignesh et al. (2015)	Aferimento de qualidade de faces
13	Hannane et al. (2015)	Indexação de vídeo
14	LI et al. (2017)	Reconhecimento facial efetivo
15	Qi et al. (2018)	Extração de *key-frame*
16	Cai e Gan (2019)	Agrupamento de faces
17	Zheng et al. (2020)	Reconhecimento facial em vídeo
18	Barra et al. (2020)	Estimativa de pose da cabeça
19	Barquero et al. (2021)	Rastreamento multiface

Por exemplo, Nasrollahi e Moeslund (2010b) utilizam seleção de faces como pré-processamento para algoritmos de super resolução. Primeiramente, avaliam a qualidade das imagens faciais com respeito a pose, nitidez, brilho e resolução para gerar o que chamam de registro intermediário, uma coleção das melhores faces ordenadas por qualidade. Em seguida, a melhor imagem do registro intermediário, juntamente das imagens mais semelhantes a ela, são separadas em um novo registro chamado registro refinado. O estudo demonstra que utilizar o registro refinado como entrada para o algoritmo de super resolução diminui os erros inseridos na reconstrução e gera resultados visivelmente mais nítidos.

Kim et al. (2014) investigam aferimento de qualidade de face como uma ferramenta de pré-processamento para reconhecimento facial. Descrevem um fluxo em cascata com 3 critérios de qualidade: pose (alinhamento), borrão (como decorrência de movimentos bruscos) e brilho.

De forma que, aproximadamente 36% das imagens de face obtidas foram rejeitadas, contribuindo com eficiência e em torno de 5% de ganho de acurácia em reconhecimento.

Hannane et al. (2015) propõem um sistema para indexar vídeos de vigilância de acordo com atributos faciais. Primeiramente, o vídeo é sumarizado em *key-frames* – quadros mais relevantes para a tarefa –, nos quais executa-se detecção de faces. As faces detectadas passam por um processo de extração de atributos faciais, que ao fim são utilizados para indexação. Os testes realizados no conjunto de vídeos *Chokepoint* (Wong et al., 2011) demonstraram sua eficácia para sumarização, de tal forma que apenas 17.56% do conjunto original foi mantido. Quando realizadas operações de consulta a partir de imagens de face não indexadas (imagens não presentes no conjunto de treino), o método apresentou precisão e *recall* de 89.83% e 92.35%, respectivamente.

As Figuras 3.1 e 3.2 apresentam os algoritmos utilizados pelos trabalhos selecionados nas etapas de detecção e rastreamento de faces, respectivamente, de forma a analisar quais os algoritmos mais comuns para cada etapa.

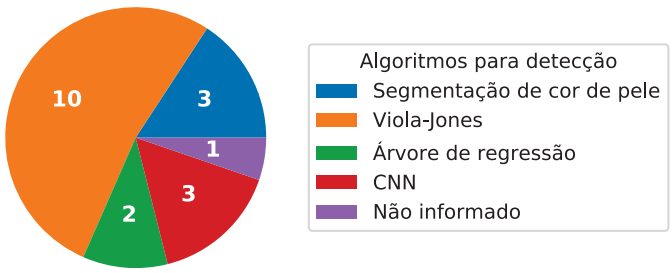


Figura 3.1: Gráfico demonstrando a proporção de algoritmos utilizados pelos trabalhos selecionados para a detecção de faces em vídeos.

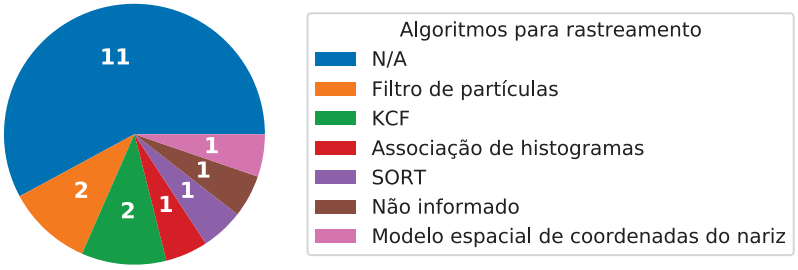


Figura 3.2: Gráfico demonstrando a proporção de algoritmos utilizados pelos trabalhos selecionados para o rastreamento de faces em vídeos.

Quanto aos detectores de face utilizados, percebe-se a prevalência do *framework* de detecção Viola-Jones (Viola e Jones, 2004), que apesar de ser um método bem estabelecido e rápido, já é antigo e também é conhecido por ser propenso a falso-positivos. Ademais, o recente desempenho ímpar de CNNs nas tarefas de detecção e classificação demanda testes de modelos atuais para validação da escolha do detector. Os trabalhos mais recentes já demonstram a preferência de CNNs para detecção.

O gráfico da Figura 3.2 demonstra que, dos 19 trabalhos selecionados, apenas 8 utilizam alguma forma de rastreamento (um dos quais não informa o algoritmo utilizado, mas afirma

a realização de rastreamento). Quando se trabalha com cenários de vigilância, a ausência de um método de associação das detecções – como um rastreador – implica na incapacidade do sistema em tratar múltiplas faces concorrentemente em uma mesma sequência, logo uma lacuna importante é destacada.

Outro ponto a ser destacado é a escassez de *datasets* voltados especificamente para a tarefa de rastreio multiface em cenários de vigilância (Barquero et al., 2021). Os *datasets* frequentemente mais citados para rastreamento são: PETS2009 (Ferryman e Shahrokni, 2009), PETS2014 (Patino e Ferryman, 2014), PETS2017 (Patino et al., 2017), CAVIAR (Fisher, 2003), LTDT (Camps et al., 2014), i-LIDS (i-LIDS Team, 2006), ETISEO (Nghiem et al., 2007), VIRAT (Oh et al., 2011), CVBASE (Pers e Magee, 2006), VOT (Kristan et al., 2013) e MOTChallenge (Leal-Taixé et al., 2015).

Entretanto, os conjuntos citados referem-se a tarefas como: rastreamento de pessoas, rastreamento de veículos, análise de comportamento humano para detecção de eventos perigosos e re-identificação de pessoas. Outros conjuntos que se dispõem para o estudo de faces, não oferecem anotações (*groundtruth*) para rastreio simultâneo de múltiplas faces, por exemplo: 300-VW (Shen et al., 2015) oferece anotações de pontos fiduciais (pontos localizados ao redor da boca, olhos e nariz) para uma pessoa por sequência; o já citado *Chokepoint* (Wong et al., 2011) apesar de ser gravado em um cenário de vigilância relevante ao tema, oferece anotações de *groundtruth* que atentam-se apenas à posição dos olhos.

Para avaliar métodos de rastreio para múltiplas faces, precisa-se de um conjunto que: anote as posições de todas as faces presentes em cada quadro, seja por retângulo envolvente ou máscara de segmentação, e que todos os retângulos envolventes que anotam a mesma face estejam identificados com um número único.

Barquero et al. (2021) apresentam um trabalho recente que também afirma a falta de *datasets* para a tarefa de rastreio multiface em cenários não controlados. Assim, desenvolvem um *dataset* com 8 minutos e 54 segundos totais de vídeos anotados com posições de faces e destacam-se como um trabalho que propõe um conjunto para o estudo de múltiplas faces em cenários não controlados, cujas anotações do *dataset* estão disponíveis para o público, o que é importante para reprodutibilidade e avanço da pesquisa.

Visto que essas anotações são escassas, propõe-se também a suplementar o conjunto de Barquero et al. (2021). A discussão sobre os conjunto de dados utilizados fica reservada ao Capítulo 5.

Além disso, Barquero et al. (2021) também desenvolvem um sistema para rastreamento de múltiplas faces, composto de quatro módulos. O objetivo da proposta é obter rastros mais longos com a utilização de um módulo para reconexão dos *tracklets*. As faces são detectadas com a CNN *Faceboxes* (Zhang et al., 2017), associadas quadro a quadro e rastreadas com o rastreador KCF (Henriques et al., 2015). Para reconectar *tracklets* fragmentados, utiliza-se a CNN Arcface (Deng et al., 2019) nas faces de melhor qualidade para gerar representações faciais de cada *tracklet*. Por último, estas representações faciais são comparadas, umas com as outras, e os *tracklets* são reconectados quando representam a mesma pessoa.

De forma homóloga, Zheng et al. (2020) realizam rastreamento em vídeo, porém com o foco em reconhecimento de faces em vídeos irrestritos. Isto significa que, o objetivo neste caso foi associar os *tracklets* obtidos de cada vídeo, com uma identidade em uma galeria de imagens. As faces são detectadas com uma CNN multiescala, rastreadas com o método SORT (Bewley et al., 2016) e as representações de face são geradas extraindo atributos profundos de cada face detectada e atribuindo-as pesos para valorizar amostras com maior qualidade.

No que tange aferimento de qualidade de face, a abordagem mais comum é selecionar um conjunto de "*scores*", que são pontuações utilizadas pelos autores para traduzir a qualidade

de uma imagem. Exemplos de *scores* comuns são: resolução do recorte da face, nível de nitidez, quantia de *pixels* cor de pele, medida de confiança do detector utilizado e orientação, ou pose, da face.

Por exemplo, Barra et al. (2020) utilizam apenas um *score* de pose da face, obtido pelo modelo desenvolvido para estimativa de pose da cabeça. Cai e Gan (2019) servem-se de 3 *scores*: valor de confiança da detecção de face, a resolução do recorte obtido e uma medida do quão embaçada está a imagem.

Chen et al. (2007) extraem 8 *scores* de qualidade, que são normalizados em um intervalo de $[0, 1]$ e utilizados como entrada para uma rede neural de uma camada, que fica responsável por produzir uma pontuação final para cada detecção.

No geral, os *scores* são normalizados e combinados em um único valor que traduz a qualidade de cada imagem e serve para ranqueá-las em melhores ou piores.

Concluída a análise dos trabalhos selecionados, pode-se afirmar que:

1. Há necessidade da realização de testes com outros detectores além do *framework* Viola-Jones, preferivelmente CNNs;

2. Percebe-se o uso de poucas propostas para rastreamento. Visto que o presente trabalho envolve a aplicação de seleção de faces no contexto de vigilância, comumente irrestrito e populoso, é imprescindível utilizar rastreamento para associação das detecções. O que exige, assim como com os detectores, o estudo e emprego de métodos atuais;

3. A criação de um *dataset* para avaliar rastreamento de múltiplas faces em ambientes não controlados está justificada.

4 METODOLOGIA

Este capítulo apresenta o sistema utilizado como *baseline* e utilizado para rastreamento e seleção de faces, assim como as alterações propostas e motivações por trás das mesmas. O sistema escolhido foi desenvolvido por Barquero et al. (2021) e pode ser descrito como um sistema de verificação baseado em ranque para rastreio de faces a longo prazo em ambientes populosos (LTFT, do inglês *Long-Term Face Tracking*).

A Seção 4.1 apresenta os detalhes da arquitetura composta de quatro módulos do sistema LTFT, seguida dos detalhes de implementação. Finaliza-se na Seção 4.2, onde discute-se as mudanças propostas a serem realizadas no fluxo do LTFT.

4.1 ARQUITETURA DO SISTEMA LTFT

O sistema LTFT é um fluxo para processamento de vídeos, composto de quatro módulos responsáveis pela aquisição das faces nas imagens, associação e propagação da identidade de uma face detectada no quadro anterior para o quadro seguinte, aferimento de qualidade de face e reconhecimento de faces.

4.1.1 Módulo de Rastreamento

Após iniciados os *tracklets*, este módulo é responsável por continuar predizendo as posições da face para cada *tracklet* onde não houve uma correspondência realizada pelo módulo de associação de dados descrito na subseção 4.1.2.

Isto é, seja $\mathcal{T}_t$ o conjunto de *tracklets* ativos no quadro t e N_{t+1} o número de faces detectadas no quadro $t + 1$. Para cada *tracklet* $T_i \in \mathcal{T}_t$ que não puder ser associado com nenhuma das novas N_{t+1} detecções, T_i será marcado como desaparecido e terá sua posição predita nos próximos quadros pelo rastreador KCF (Henriques et al., 2015).

O módulo de rastreamento irá continuar predizendo a posição de cada T_i desaparecido, até que o módulo de associação de dados conecte T_i a uma nova detecção, ou desative o *tracklet* T_i após T_{max} quadros marcado como desaparecido.

4.1.2 Módulo de Associação de Dados

Em métodos de rastreamento por detecção (*tracking-by-detection*), um detector de faces é utilizado para encontrar as faces em todos os quadros da sequência de vídeo e um método de associação fica responsável por propagar a identidade de uma detecção para os quadros seguintes.

No caso do LTFT, o detector de faces utilizado é o modelo *Faceboxes* (Zhang et al., 2017), e a associação entre os *tracklets* ativos e as detecções é feita por meio do método Húngaro de associação (Kuhn, 1955) aplicado aos valores de IoU das posições dos *tracklets* no quadro anterior, com as detecções no quadro atual.

Para cada associação cujo valor de IoU for acima de um limiar λ_{IOU}, o módulo de associação ira atualizar a posição do *tracklet* correspondente com o novo retângulo envolvente. Para detecções sem associação a nenhum *tracklet*, um novo *tracklet* é gerado e considerado uma nova identidade. Por fim, para *tracklets* que não foram correspondidos, o módulo de rastreamento continua predizendo sua posição, sendo que depois de T_{max} quadros sem nenhuma correspondência, o *tracklet* é considerado como inativo e deixa de ser computado.

4.1.3 Módulo para Reconexão de *Tracklets* Baseado em Faces

Fragmentações são um problema comum na área de rastreamento de objetos e ocorrem quando um objeto de interesse passa por uma oclusão, resultando na perca da posição deste objeto e, quando este volta a aparecer, é tratado como um objeto diferente com outro identificador. Este módulo é responsável por reconectar *tracklets* perdidos com *tracklets* que rastreavam a mesma identidade, porém que foram gerados como resultado de fragmentação.

Para reconectar dois *tracklets* com identificadores diferentes, o módulo de reconexão gera representações das faces de cada *tracklet* utilizando o modelo ArcFace (Deng et al., 2019) para reconhecimento facial. As representações são então comparadas e, caso sejam similares o suficiente, são marcadas para reconexão.

As representações de cada *tracklet* são geradas por meio de aferimento de qualidade das faces detectadas. O sistema LTFT (Barquero et al., 2021) utiliza três medidas de qualidade: pontuação de confiança de detecção oferecida pelo detector de faces, ângulos de posição da cabeça e medida de nitidez do recorte da face. Os ângulos de posição da cabeça são obtidos pelo modelo 3DDFA (Zhu et al., 2019) para estimativa de pose e alinhamento de face. A medida de nitidez é obtida pela implementação do filtro Laplaciano modificado descrito por Nikitin et al. (2014).

De acordo com as medidas de qualidade obtidas, existem três possíveis categorias em que uma face detectada pode se encaixar, são elas:

- Faces de registro: faces com maior qualidade visual, usadas para registrar as identidades, seus limiares de qualidade de confiança de detecção, ângulos de pose e nitidez são, respectivamente, 0.95, ±25° e 0.9.

- Faces de verificação: faces que possuem qualidade o suficiente para produzir resultados confiáveis para reconhecimento, no processo de reconexão as faces verificáveis são comparadas com as faces de registro. Os limiares de confiança de detecção, ângulos de pose e nitidez para essas faces são, respectivamente, 0.8, ±60° e 0.75.

- Faces descartáveis: devido à sua baixa qualidade estas faces não são consideradas, seu uso para o reconhecimento pode gerar representações não confiáveis e, consequentemente, a reconexão de *tracklets* cujas identidades rastreadas são diferentes.

Após o passo de associação de dados, o módulo de reconexão de *tracklets* verifica a qualidade de todas as faces detectadas no quadro atual, se não forem descartáveis uma representação é obtida com o modelo de reconhecimento (Deng et al., 2019) e armazenada no *tracklet* correspondente como uma representação de registro ou de verificação.

Em seguida, sendo $\mathcal{T}$ o conjunto de todos os *tracklets* obtidos até o momento, para todo *tracklet* $T_k \in \mathcal{T}$ que foi detectado e atualizado no quadro atual, obtém-se todos os *tracklets* $T_i \in \overline{\mathcal{T}}$ onde $\overline{\mathcal{T}} = \mathcal{T} \setminus \{T_k\}$. Para cada *tracklet* T_i obtido, computa-se a média das representações de suas faces de registro, $\overline{E_{T_i}}$. Para os *tracklets* T_k, a média das representações de suas faces verificáveis, $\overline{V_{T_k}}$, também é calculada.

Adiante, seja S a função de similaridade entre duas representações do modelo de reconhecimento facial, define-se T^R como o *tracklet* candidato de ranque R, isto é, o *tracklet* na posição R após todos os candidatos terem sido ordenados de maior a menor similaridade a T_k, como descrito na Equação 4.1.

$$T^R = \underset{T_i \in \overline{\mathcal{T}} \setminus \{T^j\}_{1 \leq j < R}}{argmax} \quad S(\overline{E_{T_i}}, \overline{V_{T_k}}). \tag{4.1}$$

Para que o *tracklet* T_k seja reconectado ao T^1 (o *tracklet* de ranque 1, obtido como resultado da ordenação explicada no páragrafo anterior), ainda é necessário que duas condições sejam verdadeiras. Primeiramente, o valor de similaridade entre T_k e T^1 deve ser maior que um limiar λ_{FBTR}, como exposto na equação 4.2, onde $0 \leq \lambda_{FBTR} \leq 1$,

$$S(\overline{E_{T^1}}, \overline{V_{T_k}}) \geq \lambda_{FBTR}. \tag{4.2}$$

Não obstante, o valor de similaridade entre T_k e T^1 também deve ser maior que a média dos próximos C maiores valores considerando uma margem de $1/\epsilon$:

$$S(\overline{E_{T^1}}, \overline{V_{T_k}}) \geq \frac{1}{\epsilon} \cdot \frac{1}{C} \sum_{r=2}^{C+1} S(\overline{E_{T^r}}, \overline{V_{T_k}}), \tag{4.3}$$

onde $0 < \epsilon \leq 1$ e $C \in \mathbb{N} \geq 1$.

Desse modo, sempre que T^1 verificar as duas condições impostas pelas Equações 4.2 e 4.3 os *tracklets* T_k e T^1 são marcados para reconexão. Um par (T_k, T^1) é gerado e armazenado em uma lista de pares marcados para reconexão, que serão em seguida enviados ao módulo de correção.

4.1.4 Módulo de Correção

O módulo de correção atua sob os pares associados pelo módulo de reconexão, onde para cada par (T_k, T^1) este módulo muda o identificador de T_k para o de T^1 e no processo transfere todo o histórico de posições e detecções de T_k para T^1.

A figura 4.1 do trabalho de Barquero et al. (2021) exemplifica o benefício de utilizar o processo de reconexão e correção, evitando a troca de identificadores quando dois ou mais *tracklets* rastreiam a mesma face na sequência de vídeo.

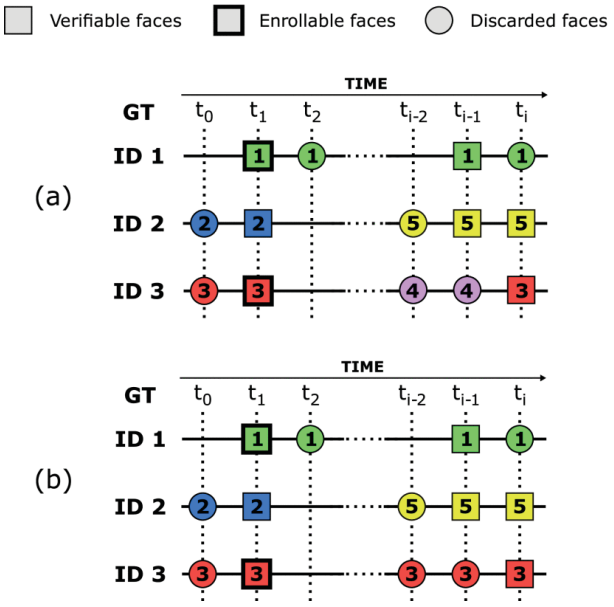


Figura 4.1: (a) Resultados de rastreamento sem o módulo de correção. (b) Resultados com o módulo de correção. Os Números dentro das formas coloridas se referem ao identificador numérico atribuído ao *tracklet* pelo sistema LTFT. Em (b), como no quadro t_1 a face de ID 3 detectada é de registro, isso permite a reconexão do *tracklet* T_4 com o T_3 no quadro t_i e também atualiza o identificador em todos os quadros anteriores. Figura disponível em Barquero et al. (2021), página 5.

4.2 MUDANÇAS PROPOSTAS

Nesta seção são descritas as mudanças propostas que visam expandir o sistema original e obter resultados melhores baseados em métricas que refletem acurácia, precisão e número de fragmentações nas tarefas de rastreamento e seleção de faces.

4.2.1 Detectores Utilizados

Como visto anteriormente, métodos de *tracking-by-detection* dependem fortemente do desempenho do detector (Park et al., 2021). Neste caso, o modelo selecionado por Barquero et al. (2021) para detecção de faces no sistema LTFT foi a rede *Faceboxes*.

Faceboxes (Zhang et al., 2017) é uma rede SSD, isto é, o modelo é capaz de predizer as posições dos objetos de interesse em um passo completo da imagem pela rede neural. A rede *Faceboxes* em específico, foi desenvolvida com a proposta de ser executável em tempo real por unidades de processamento central (CPU), sendo um modelo rápido, capaz de processar vários quadros por segundo.

Entretanto, devido as suas camadas de convolução iniciais, denominadas camadas de convolução rapidamente digeridas, o tamanho espacial da entrada é diminuído celeremente com uma série de passos (*strides*) consideráveis nas camadas de convolução e de *pooling*. Naturalmente, com a rápida diminuição do tamanho da imagem de entrada, o modelo apresenta dificuldades na detecção de faces pequenas. Uma tentativa de amenizar este detalhe é feita com camadas de convolução para múltiplas escalas, porém, o modelo apresenta dificuldades em *datasets* com alta variação na escala e pose de faces, tal qual o subconjunto denominado difícil do *dataset WIDER FACE* (Yang et al., 2016).

Por conseguinte, é interessante analisar outros modelos de detecção, que possam substituir *Faceboxes* e possivelmente aprimorar os resultados de rastreamento. Modelos como DSFD (Li et al., 2019), RetinaFace (Deng et al., 2020), SRN (Chi et al., 2019), YOLO5Face (Qi et al., 2021) e TinaFace (Zhu et al., 2020) foram selecionados para esta etapa.

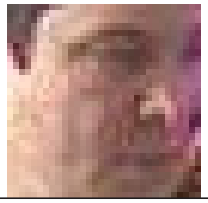
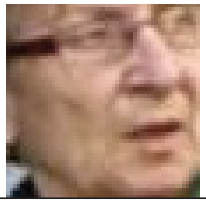
4.2.2 Aferimento de Qualidade das Faces

Barquero et al. (2021) utilizam em seu trabalho, três *scores* para aferir a qualidade das faces detectadas: confiança de detecção, pose e nitidez. A confiança de detecção é um valor no intervalo $[0, 1]$ oferecido pela CNN para cada face detectada; a pose é obtida com o modelo CNN 3DDFA (Zhu et al., 2019), e é um conjunto de três valores que representam a rotação angular da face em torno dos três eixos; a nitidez é calculada como a média da resposta do filtro Laplaciano modificado de Nikitin et al. (2014). A Tabela 4.1 demonstra quatro recortes de face e seus três *scores* respectivos para medida da qualidade de cada detecção.

No entanto, uma medida de qualidade recorrente em outros trabalhos é a resolução dos recortes de face. A adoção de faces com baixa resolução pode inserir ruídos no processo de detecção de pose e de reconhecimento facial. Por exemplo, Boom et al. (2006) argumentam que retângulos envolventes com resoluções menores que 32x32 começam a degradar os resultados de reconhecimento, e Marciniak et al. (2015) reduziram artificialmente a resolução de imagens de faces e demonstraram queda de acurácia na tarefa de reconhecimento facial.

Além disso, a medida de nitidez utilizada atribui valores muito pequenos (consistentemente menores que 0.1 para um intervalo esperado de $[0, 1]$) para maioria das faces detectadas e dificulta o processo de reconexão de *tracklets* descrito na Subseção 4.1.3. Assim, propõe-se a substituição do filtro Laplaciano modificado (Nikitin et al., 2014) pelo método proposto por Nasrollahi e Moeslund (2011) e descrito pela Fórmula 4.4,

Tabela 4.1: Medidas de qualidade de faces, utilizadas no sistema LTFT.

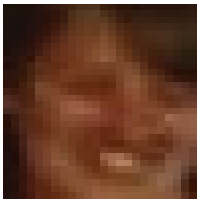
Recorte de Face				
Confiança	1.0	0.91	1.0	1.0
Ângulos (em º)	[-32.36, 10.33, -4.34]	[-54.06, 4.95, 2.04]	[-31.02, -2.22, 1.85]	[-3.48, 13.77, -0.5]
Nitidez	0.0125	0.0124	0.0188	0.031

$$Sh_{X_i} = avg(abs(X_i - lowpass(X_i))), \tag{4.4}$$

onde X_i é o i-ésimo recorte de face obtido na sequência de vídeo, Sh_{X_i} é a medida de nitidez da imagem X_i, avg é a média, abs é a função de valor absoluto e $lowpass$ um filtro de média simples de tamanho 3x3.

A Tabela 4.2 demonstra exemplos da diferença entre as duas abordagens para pontuação da nitidez de uma face, todas as faces são redimensionadas para 120x120 antes da execução dos respectivos algoritmos.

Tabela 4.2: Comparação das medidas de nitidez.

Recorte de Face				
Medida de Nitidez Original (Nikitin et al., 2014)	0.043	0.031	0.009	0.022
Medida de Nitidez Proposta (Nasrollahi e Moeslund, 2011)	104.38	79.12	25.91	93.31

4.2.3 Enriquecimento do Conjunto de Dados

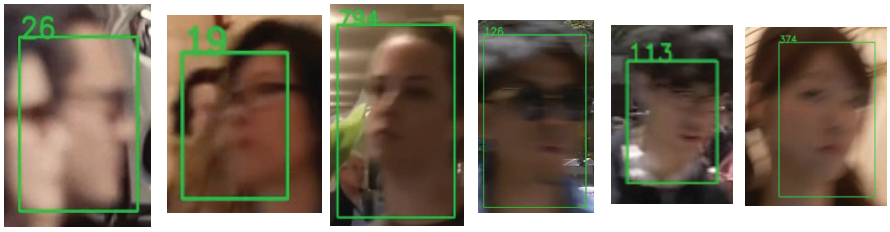
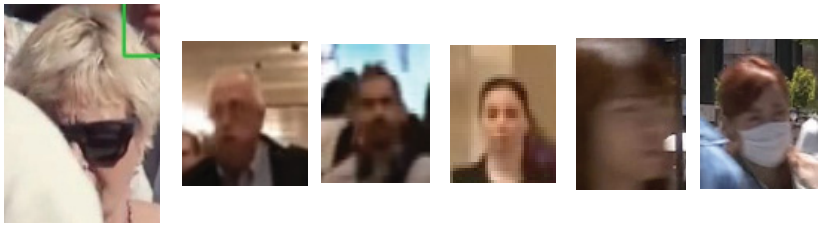
Como discutido na Seção 3.3, e apontado por Barquero et al. (2021), são escassos os *datasets* publicamente disponíveis para avaliação e desenvolvimento de algoritmos e sistemas para rastreamento de múltiplas faces, em especial, nos cenários de vigilância. Logo, Barquero et al. (2021) propõem seu próprio *dataset*, composto de 10 vídeos e com um total de 8 minutos e 54 segundos de duração.

A composição e detalhes do *dataset* de Barquero et al. (2021) são discutidos mais a fundo no Capítulo 5, entretanto, alguns pontos são aqui destacados como justificativa da necessidade de anotar mais vídeos para enriquecer o conjunto de dados de teste.

Primeiramente, o processo de anotação do *dataset* se deu por meio de anotações semi-automáticas. Neste caso, o modelo para detecção *Faceboxes* (Zhang et al., 2017) foi utilizado para obter retângulos envolventes ao redor das faces. Em seguida, as detecções foram verificadas e anotadas com identificadores numéricos para representar o percurso de cada face. Porém, o mesmo modelo *Faceboxes* é utilizado como detector no sistema LTFT, assim, é necessário a introdução de novos vídeos para analisar o desempenho de *Faceboxes* em um *dataset* menos tendencioso.

Em segundo lugar, é possível notar a presença de ruídos nas anotações. Exemplos de falsos positivos, erros de precisão e falsos negativos são demonstrados na Tabela 4.3, corroborando com a proposta de fornecer mais vídeos para o conjunto de testes. Logo, propõe-se anotar manualmente um subconjunto de vídeos com as posições e identificadores para cada face. Os detalhes sobre o conjunto suplementar desenvolvido cabe ao Capítulo 5.

Tabela 4.3: Exemplos de falhas de anotação no *dataset* proposto por Barquero et al. (2021).

Erros de Precisão	
Falsos Negativos	
Falsos Positivos	

5 EXPERIMENTOS E RESULTADOS

Este Capítulo trata dos detalhes que envolvem a aplicação da metodologia proposta. A Seção 5.1 apresenta as métricas selecionadas para avaliação do sistema de rastreamento. Precisamente, as duas *Clear MOT Metrics*: *Multiple Object Tracking Accuracy* (MOTA) e *Multiple Object Tracking Precision* (MOTP) (Bernardin e Stiefelhagen, 2008); a IDF1 (Ristani et al., 2016); falsos negativos, falsos positivos e outras métricas adicionais são utilizadas.

Em seguida, a Seção 5.2 apresenta o conjunto de dados desenvolvido por Barquero et al. (2021). E também aborda o processo de criação do *dataset* proposto para suplementá-lo, os vídeos utilizados, o *dataset* de origem desses vídeos e como foram rotulados para avaliação de um sistema para rastreamento e seleção de faces.

A Seção 5.3 descreve os experimentos realizados para seleção dos melhores métodos de detecção de faces. A Seção 5.4 estuda a variação de parâmetros no *baseline*. A Seção 5.5 discute os resultados obtidos com a substituição do detector, substituição do método de aferimento de nitidez e adição da resolução das faces como medida de qualidade (como proposto anteriormente na Seção 4.2.2). Todos os experimentos foram realizados na plataforma Google Colab Pro, em uma máquina equipada com uma CPU Intel(R) Xeon(R) em 2.00GHz e uma placa gráfica NVIDIA Tesla P100.

5.1 MÉTRICAS

A tarefa de MOT exige métricas específicas para avaliar as capacidades de cada algoritmo. Esta seção apresenta as métricas IDF1, *Multiple Object Tracking Precision* (MOTP) e *Multiple Object Tracking Accuracy (MOTA)* estabelecidas na literatura e utilizadas em *benchmarks* populares como MOT *Challenge* (Milan et al., 2016).

5.1.1 Métricas de Acurácia e Precisão Para Rastreamento Multiobjeto

Bernardin e Stiefelhagen (2008) propõem em seu trabalho as *Clear MOT Metrics*, compostas pelas MOTP e MOTA. Ambas são antecedidas de um processo de correspondência entre hipóteses – propostas pelo rastreador – e objetos-alvo, cujas posições são conhecidas de antemão por um arquivo verdade (*groundtruth*).

Esse processo de correspondência utiliza a medida de distância IoU, representada visualmente pela Figura 5.1. Essa medida expressa o quão bem a hipótese proposta pelo rastreador cobre a posição verdadeira do alvo. Geralmente define-se também um limiar T de forma que valores de IoU maiores ou iguais a T são considerados uma correspondência bem sucedida, ou um erro caso contrário.

A métrica MOTA considera todos os erros de correspondência que um rastreador realiza ao longo de todos os quadros e traduz a habilidade do rastreador de manter a trajetória do objeto rastreado, independentemente da precisão com a qual a trajetória é mantida. A Figura 5.2 demonstra um caso de erro (*mismatch*) e um de acerto, dependendo se a medida de distância é menor ou não que um limiar T.

Calculados os erros e acertos de correspondência hipótese-alvo e sejam (no quadro de tempo t) m_t o número de falsos negativos, fp_t o número de falsos positivos, mme_t o número de

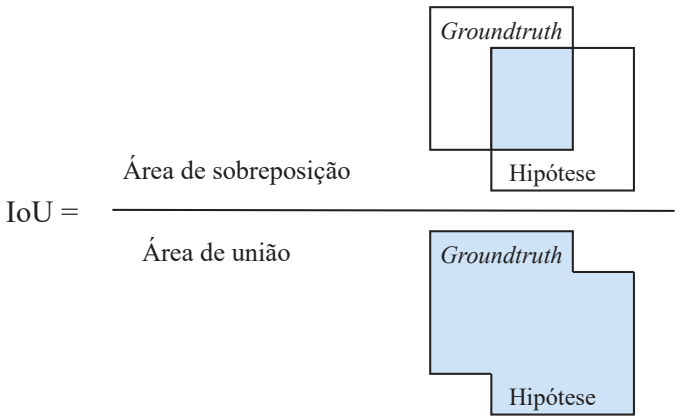


Figura 5.1: Representação visual da medida de distância *Intersection over Union* (IOU).

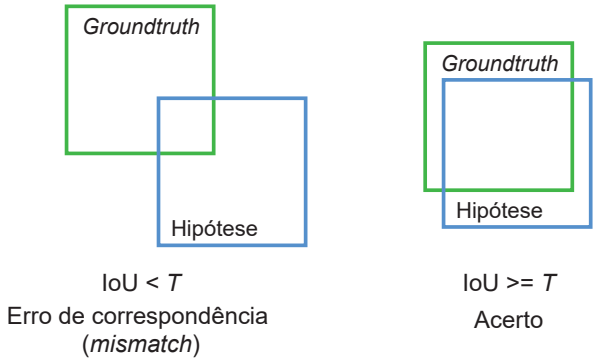


Figura 5.2: Demonstração de erro de correspondência, quando a distância IoU é menor que o limiar T, e de acerto, quando a distância IoU é maior ou igual ao limiar T.

erros de correspondência (*mismatches*) e g_t o total de objetos-alvo, a MOTA é expressa abaixo pela equação 5.1.

$$MOTA = 1 - \frac{\sum_t (m_t + fp_t + mme_t)}{\sum_t g_t} \tag{5.1}$$

Se por um lado a acurácia traduz a proporção de acertos do rastreador avaliado, a precisão traduz a capacidade de um rastreador estimar precisamente as posições dos objetos-alvo e é exemplificada pela Figura 5.3.

Nesse contexto, a MOTP representa o erro total nas posições estimadas para os pares hipótese-alvo ao longo de todos os quadros, normalizado pelo número de correspondências feitas. Sendo d_t^i a distância entre cada associação alvo-hipótese i no quadro de tempo t e c_t o número de correspondências no quadro de tempo t, a MOTP é representada pela equação 5.2.

$$MOTP = \frac{\sum_{i,t} d_t^i}{\sum_t c_t}. \tag{5.2}$$

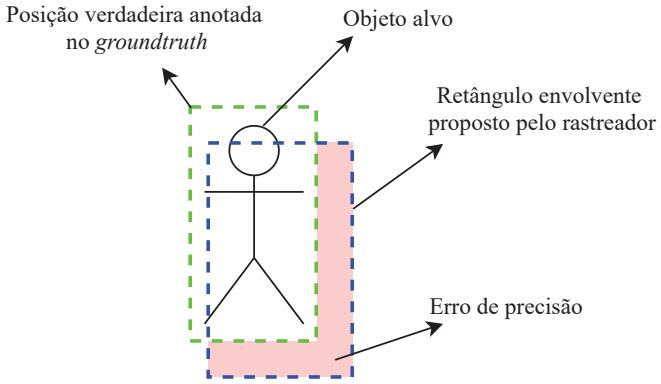


Figura 5.3: A precisão diz respeito ao quão bem o detector ou rastreador conseguem predizer a posição do alvo, a área vermelha representa o erro de precisão.

5.1.2 Pontuação IDF1

Proposta por Ristani et al. (2016) , a IDF1 objetiva medir a performance pela quantia de tempo em que o rastreador corretamente identifica o alvo e não pela frequência em que os erros de correspondência occorem, diferentemente do que ocorre em MOTA.

O primeiro passo nessa abordagem, assim como com as métricas *Clear MOT* (Bernardin e Stiefelhagen, 2008), é corresponder os objetos-alvo com as hipóteses computadas pelo rastreador. Entretanto, diferentemente das *Clear MOT*, a correspondência das hipóteses com os alvos, nesse caso, não ocorre a cada quadro t, e sim por meio da construção de um grafo bipartido que associa uma trajetória verdade (anotada no *groundtruth*) a apenas uma trajetória computada, minimizando o número de erros de associação ao longo de todos os quadros disponíveis.

Os parâmetros $IDTP$, $IDFP$ e $IDFN$ são então descobertos como resultado dessa correspondência e representam o número de verdadeiros positivos, falsos positivos e falsos negativos, respectivamente. O cálculo da IDF1 é representado pela equação 5.3 abaixo.

$$IDF1 = \frac{2IDTP}{2IDTP + IDFP + IDFN} \tag{5.3}$$

5.1.3 Métricas Suplementares

Além das métricas anteriormente descritas, também serão utilizadas medidas de Falsos Positivos (FP), Falsos Negativos (FN), *ID-Switches* (IDS) e *Mostly Tracked* (MT).

FP é a quantia de erros feitos pelo rastreador, corresponde ao número de vezes onde uma face foi predita erroneamente. FN é o número de vezes onde uma face aparece na sequência, porém não foi contabilizada ou rastreada pelo método proposto. IDS é o número de vezes em que o sistema troca erroneamente o identificador numérico de um *tracklet*. Por fim, MT é o número de *tracklets* que foram rastreados corretamente por, no mínimo, 80% da sua trajetória total.

5.2 CONJUNTOS DE DADOS

Com a finalidade de analisar cada uma das etapas necessárias para o funcionamento de um sistema para seleção de faces – detecção, rastreamento e aferimento de qualidade –, é imprescindível a utilização de um *dataset* que represente a tarefa de vigilância em questão. Isso exige que: (1) os vídeos representem os cenários não controlados, comumente vigiados por sistemas de monitoramento por vídeo; (2) múltiplas faces estejam presentes simultaneamente em

algum momento das sequências de vídeo; e (3) as anotações contenham um identificador para cada face e a mesma esteja anotada por retângulos envolventes a cada quadro em que apareça no vídeo.

Esta Seção trata do *dataset* proposto por Barquero et al. (2021) e do *dataset* suplementar desenvolvido para esta dissertação.

5.2.1 *Dataset*: Rastreamento Facial em Cenários Populosos

Barquero et al. (2021) identificam a escassez de *datasets* voltados à avaliação de métodos para rastreamento multiface em ambientes não controlados, e propõem um *dataset* composto de 10 vídeos em diferentes resoluções, com faces anotadas por meio de retângulos envolventes e identificadores numéricos únicos para cada identidade. As anotações foram obtidas de forma semi-automática com a utilização do detector Faceboxes (Zhang et al., 2017) e, em seguida, com ajuste manual e anotação dos identificadores para cada retângulo envolvente.

Duas das sequências anotadas foram obtidas do *dataset ChokePoint* (Wong et al., 2011), que foi projetado originalmente para experimentos de reconhecimento facial, subtração de plano de fundo e detecção de faces. As sequências foram filmadas por meio de três câmeras posicionadas acima de dois portais, gravadas em uma taxa de 30 quadros por segundo com resolução de 800x600 *pixels*.

As sequências do conjunto *ChokePoint* apresentam variações de iluminação, ocorrência de oclusão por sobreposição de faces e alteração no tamanho da face capturada conforme a pessoa se aproxima da câmera. Ademais, esse conjunto possui uma variação significativa de aparência física das pessoas filmadas, como, por exemplo: sexo, etnia, barba, idade e uso de acessórios como óculos.

Entretanto, as anotações originalmente disponibilizadas atentam apenas à posição dos olhos de cada sujeito. Portanto, Barquero et al. (2021) selecionaram as sequências de vídeo com maior densidade de pessoas, especificamente, as sequências P2E_S5 e P2L_S5, exemplificadas pela Figura 5.4 e denotadas na Tabela 5.1 como Choke1 e Choke2, respectivamente.

(a) (b)

Figura 5.4: Exemplos de quadros anotados de duas sequências do *dataset ChokePoint*. (a) Quadro da sequência P2E_S5 - Câmera 2, onde percebe-se um caso de sobreposição parcial de faces e uma variação de iluminação conforme as faces se aproximam da câmera; (b) Quadro da sequência P2L_S5 - Câmera 2, onde nota-se uma face completamente oclusa e, consequentemente, não anotada nesse quadro específico.

Além disso, oito vídeos são obtidos na plataforma YouTube. O resultado é um *dataset* com 8 minutos e 54 segundos de vídeos anotados com a posição de múltiplas faces. A Figura 5.5 apresenta dois quadros anotados de dois vídeos diferentes. A Tabela 5.1 detalha as características

de cada sequência anotada, onde a coluna de densidade refere-se a média de faces detectadas em cada quadro.

(a) (b)

Figura 5.5: Exemplos de quadros anotados de duas sequências do *dataset* LTFT, obtidas na plataforma de vídeos YouTube. (a) Quadro da sequência Bengal, onde percebe-se múltiplas faces de tamanhos diferentes em um cenário propício a oclusões; (b) Quadro da sequência Terminal2, onde nota-se uma oclusão parcial e, novamente, variação de tamanho das faces devido a distância.

Tabela 5.1: Descrição das sequências de vídeo e anotações do *dataset* desenvolvido por Barquero et al. (2021). A coluna de densidade se refere ao número médio de faces detectadas por quadro.

Nome	Resolução	Duração	BBoxes	Densidade	N° IDs
Choke1	800x600	1’24"	7964	4.0	24
Choke2	800x600	1’11"	8710	4.8	26
Terminal1	1920x1080	1’18"	13722	5.9	148
Terminal2	1920x1080	1’15"	11551	5.2	140
Terminal3	1920x1080	26"	4255	5.5	59
Terminal4	1920x1080	35"	6756	6.6	126
Sidewalk	1920x1080	27"	8433	13.0	34
Bengal	1920x1080	40"	6953	6.9	36
Street	1920x1080	1’8"	4883	2.9	31
Shibuya	3840x2160	30"	8058	9.0	91

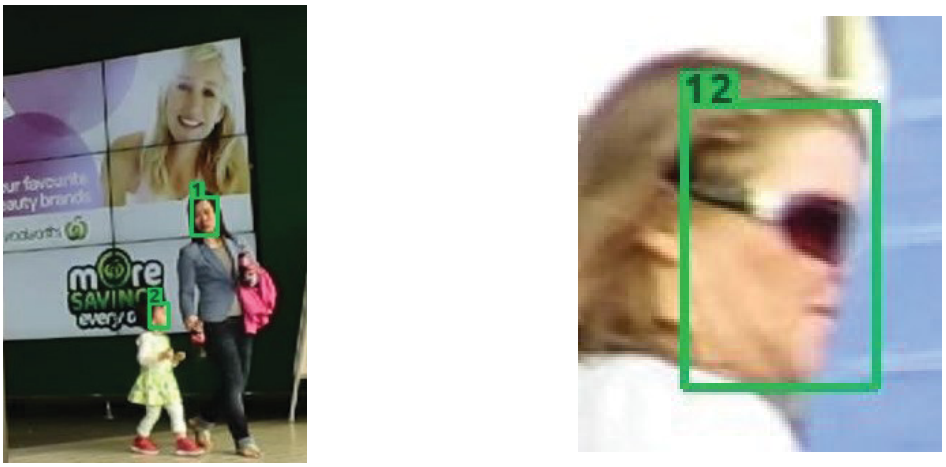
5.2.2 *Dataset* Proposto

Visando enriquecer o conjunto de dados que será utilizado para avaliação do sistema resultante, propõe-se anotar mais vídeos selecionados do *dataset* MOT17 (Milan et al., 2016). Os parágrafos a seguir discorrem, nessa ordem, sobre: o processo de anotação; apresentação das sequências anotadas e sua origem; e, por fim, são apresentadas as características gerais do *dataset* proposto, como, resolução dos vídeos, número de retângulos envolventes (*bounding boxes*) e quais desafios cada sequência apresenta.

Primeiramente, a anotação se deu pelo uso da ferramenta *Computer Vision Annotation Tool* (CVAT) (Sekachev et al., 2020), onde as posições das faces foram manualmente rotuladas com retângulos envolventes e identificadores numéricos únicos em um total de três vídeos.

As regras para anotação foram: não são anotadas representações artificias de faces, tais quais reflexos, desenhos, sombras ou imagens em painéis publicitários; a caixa delimitadora

envolve a face da forma mais compacta possível; em casos de oclusão parcial, estima-se a extensão do objeto de acordo com outras informações, como, sombras ou quadros anteriores; as trajetórias se iniciam o mais cedo possível e terminam o mais tarde possível; são anotados todos os objetos de interesse que aparecem e podem ser identificados ou estimados pelo anotador. A Figura 5.6 demonstra exemplos das regras citadas para anotação.



(a) (b)

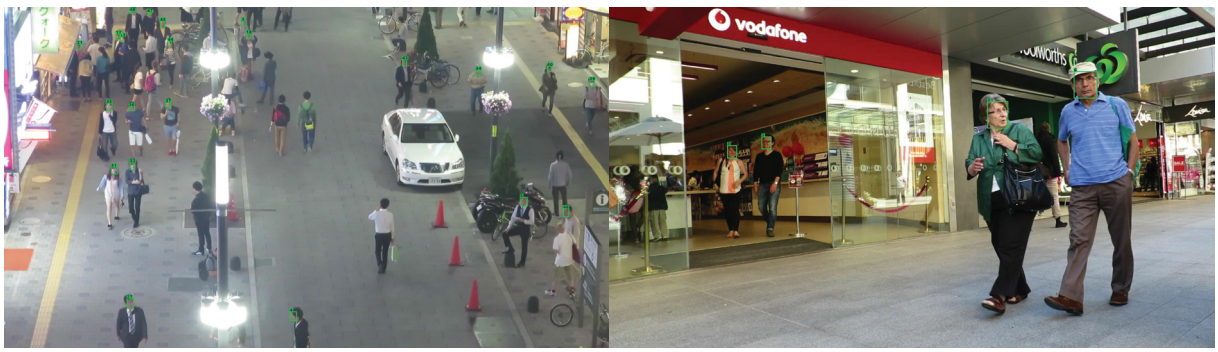
Figura 5.6: Recortes de imagens anotadas onde ocorrem: (a) Duas faces anotadas e uma não anotada por ter sido representada artificialmente em um painel publicitário; e (b) Face anotada apesar de oclusão parcial.

Além disso, para avaliar métodos de rastreamento, não apenas deve-se anotar a posição dos objetos de interesse – representada pelo retângulo envolvente –, como também é necessário atribuir um identificador único para a mesma face ao longo de toda sequência. Para demonstrar esse aspecto das anotações, a Figura 5.7 elenca, quadro a quadro, um caso onde a face anotada deixa de ser visível devido a oclusão total e rotação acentuada. Entretanto, nas instâncias em que a face volta a ser visível novamente, sua posição é anotada e o mesmo identificador numérico é mantido.

Figura 5.7: Demonstração de como as faces são anotadas ao longo de seu percurso, de forma que, apesar de oclusões totais e rotações extremas, a face volta a ser anotada com o mesmo identificador caso reapareça.

No que diz respeito ao *dataset* de origem destas sequências, o conjunto MOT17 (Milan et al., 2016) foi originalmente elaborado para o desenvolvimento de rastreadores de objetos, tais quais pessoas e veículos. O conjunto é dividido em dois subconjuntos de treino e teste, com 7 sequências de vídeo em cada.

Das 14 sequências disponíveis, 3 foram selecionadas para anotação: MOT17-01, MOT17-04 e MOT17-09. Dois critérios foram utilizados para a seleção dessas 3 sequências: os ambientes gravados em cada uma são diferentes uns dos outros e a câmera encontra-se fixa em uma base estática, assim como câmeras de sistemas de vigilância por vídeo (existem sequências onde a câmera está anexada a um veículo ou à cabeça de uma pessoa em movimento, logo essas sequências não foram consideradas). A figura 5.8 apresenta dois quadros anotados como exemplo.



(a) (b)

Figura 5.8: Exemplos de quadros anotados de duas sequências do *dataset* MOT17. (a) Quadro anotado da sequência MOT17-04, onde a câmera se encontra em uma posição elevada e com várias fontes de luz espalhadas ao longo da cena; (b) Quadro anotado da sequência MOT17-09, onde a câmera está abaixo da altura dos olhos e as pessoas na rua movem-se livremente.

A tabela 5.2 demonstra informações descritivas das sequências e suas anotações. Ao todo, o conjunto dispõe de 22575 faces anotadas por retângulos envolventes (denotado na tabela

por BBoxes), 2025 quadros ao todo e 81 pessoas diferentes a serem rastreadas. As colunas de "Menor BBox" e "Maior BBox" representam a área, em *pixels*, do menor e do maior retângulo envolvente presente em cada sequência.

Tabela 5.2: Dados das sequências de vídeo e anotações.

Sequência	Resolução	N° *Frames*	BBoxes	Menor BBox	Maior BBox	N° IDs
MOT17-09	1920x1080	525	1805	180	25496	19
MOT17-01	1920x1080	450	3833	88	2213	14
MOT17-04	1920x1080	1050	16937	106	991	48
TOTAL	–	**2025**	**22575**	–	–	**81**

As tabelas 5.3 e 5.4, a seguir, são complementares. A primeira, inspira-se nos atributos de *dataset* apresentadas por Lin et al. (2019) e descreve as características evidenciadas durante o processo de anotação.

Tabela 5.3: Descrição das características evidenciadas nas sequências anotadas.

Característica	Descrição
Oclusão (OCL)	Há momentos em que a face anotada está parcialmente oclusa
Movimentos Rápidos (MR)	Notável distância percorrida pela face anotada em poucos quadros
Rotação Fora do Plano (RFP)	A face gira de forma a sair do plano da imagem
Borrões (BOR)	A face anotada está embaçada devido a movimento rápido, iluminação ou baixa resolução
Múltiplas Faces (MF)	A sequência apresenta, em algum momento, mais de uma face anotada presente ao mesmo tempo
Fora de Visão (FV)	Alguma face anotada sai do plano da imagem, seja por oclusão total ou por sair do campo de visão da câmera

Em seguida, a Tabela 5.4 associa as características descritas pela Tabela 5.3 a cada sequência anotada. É possível notar que todas as sequências apresentam em algum momento mais de uma face presente simultaneamente e em todas ocorre oclusão das faces. Essas características justificam a criação do conjunto apresentado, visto que as sequências representam os desafios encontrados em vídeos obtidos por sistemas de vigilância, nomeadamente: múltiplas faces simultaneamente presentes, rotação livre das faces e ocorrência de oclusões.

Tabela 5.4: Relacionamento das características presentes em cada sequência, os pontos assinalam que a sequência apresenta a característica.

Sequência	OCL	MR	RFP	BOR	MF	FV	Origem
MOT17-09	•	–	•	–	•	•	MOT17
MOT17-01	•	–	•	•	•	•	MOT17
MOT17-04	•	–	•	•	•	•	MOT17

Quanto as deficiências deste conjunto, o tamanho total de quadros rotulados é pequeno. Por exemplo, se comparado ao conjunto *MobiFace* (Lin et al., 2019) para rastreamento facial em dispositivos móveis, os 2025 quadros aqui anotados correspondem a apenas 2.13% dos 95,000 quadros de *MobiFace*. Ademais, o conjunto é permeado de faces com baixa resolução, nesses

casos sendo identificadas pelo anotador pelo contexto e informações dos arredores ao invés de identificação visual clara, espera-se que essas faces sejam difíceis de serem identificadas.

5.3 AVALIAÇÃO COMPARATIVA DE DETECTORES DE FACES

Este primeiro experimento visa comparar diferentes modelos de detecção de faces, com a finalidade de escolher o mais apropriado para o sistema de seleção de faces. Analisa-se a performance dos detectores: DSFD (Li et al., 2019), RetinaFace (Deng et al., 2020), SRN (Chi et al., 2019), YOLO5Face (Qi et al., 2021) e TinaFace (Zhu et al., 2020).

A métrica escolhida é a $AP_{.50}$ (*Average Precision at IoU = .50*), comumente utilizada pelos *datasets* COCO e PascalVOC para avaliação de modelos de detecção de objetos. Esta métrica representa a área debaixo da curva de precisão (grau de exatidão do modelo ao predizer a localização do objeto) e *recall* (capacidade do modelo de encontrar todos os objetos na cena), com um limiar de IoU em 50%. Isto é, para que uma detecção seja considerada um acerto, o retângulo envolvente proposto pelo modelo deve possuir um valor de IoU ≥ 0.5 com o retângulo envolvente anotado no *groundtruth.*

Para essa tarefa, o conjunto de dados foi convertido a anotações apenas de retângulos envolventes, tal qual um conjunto para avaliação de detectores de objetos. A Tabela 5.5 demonstra os resultados obtidos de todos os modelos testados.

Tabela 5.5: Resultados dos modelos para detecção de faces testados, avaliados pela métrica de *Average Precision* com limiar de IoU 0.5 e quadros por segundo (FPS).

Modelo	$AP_{.50}$	FPS
SRN	**0.799**	0.913
YOLO5Face(s)	0.792	14.539
YOLO5Face(n)	0.787	**15.325**
RetinaFace	0.758	5.334
Faceboxes	0.716	14.881
DSFD	0.714	0.953
TinaFace	0.692	2.968

Percebe-se que o melhor resultado foi atingido pelo modelo SRN (Chi et al., 2019) com uma $AP_{.50}$ de 0.799, entretanto, sua velocidade perde significativamente se comparada com a velocidade do modelo Faceboxes (Zhang et al., 2017) originalmente utilizado no sistema LTFT (Barquero et al., 2021).

Por outro lado, as duas variações do modelo YOLO5Face oferecem resultados de $AP_{.50}$ de aproximadamente 8 pontos percentuais acima de Faceboxes, enquanto oferecem velocidades de inferência similares, ou até maior no caso do modelo YOLO5Face(n).

O último modelo entre os que apresentam resultados melhores que o *baseline* é o RetinaFace, com uma $AP_{.50}$ igual a 0.758, 4.2 pontos percentuais acima de Faceboxes, no entanto, sua velocidade é 2.79x menor que Faceboxes, de forma que sua escolha não está justificada para processamento de vídeos.

Apresentados os resultados acima, o modelo escolhido para substituir Faceboxes como detector do sistema LTFT foi YOLO5Face(s), devido a sua métrica de detecção superior e velocidade similar.

5.4 EXPERIMENTAÇÃO DE PARÂMETROS

Nesta seção são realizados uma série de experimentos que visam variar os parâmetros do sistema LTFT, tanto em sua forma *baseline*, quanto nas formas propostas neste trabalho. As métricas IDF1 (Ristani et al., 2016), MOTA e MOTP (Bernardin e Stiefelhagen, 2008), assim como métricas adicionais, tais quais, FP, FN, MT e IDS foram apresentadas, anteriormente, na Seção 5.1. Todas as métricas foram geradas utilizando um limiar de IoU = 0.5.

Primeiramente, estabelece-se o *baseline* LTFT com os parâmetros propostos por Barquero et al. (2021), sendo eles: $\lambda_{IOU} = 0.25$, $\lambda_{FBTR} = 0.5$, $\epsilon = 0.8$ e $C = 6$. Para o módulo de reconexão, os valores de confiança de detecção, ângulos da cabeça e nitidez de imagem originais são, respectivamente, 0.95, ±25° e 0.9 para faces de registro, e 0.8, ±60° e 0.75 para faces verificáveis. Para mais informações sobre os parâmetros utilizados e seus significados, o leitor é dirigido ao Apêndice A.

Na Subseção 5.4.1 busca-se encontrar o melhor valor para T_{max}, visto que esse valor não é especificado. Em seguida, na Subseção 5.4.2 estuda-se os efeitos dos limiares de nitidez na reconexão de *tracklets*.

5.4.1 Avaliando a Dependência do Rastreador

No sistema LTFT, inicia-se uma instância do rastreador KCF (Henriques et al., 2015) quando uma face detectada não pode ser associada a nenhuma outra face no quadro seguinte. O rastreador fica então responsável por predizer a posição desta face até que uma nova detecção possa ser associada a ela, ou até que se passem T_{max} quadros.

Para estudar a influência de T_{max} no sistema LTFT e escolher o valor apropriado para os testes subsequentes (o *baseline* não especifica um valor para T_{max}), são realizados uma série de experimentos que variam seu valor uniformemente. Todos os outros parâmetros são fixados nos valores propostos no *baseline*, e T_{max} é variado de 0 a 30 em intervalos uniformes de 5. Os resultados estão dispostos na Tabela 5.6.

Tabela 5.6: Experimentos com a variação do parâmetro T_{max} do sistema LTFT utilizado como *baseline*. ↑ indica que maiores resultados são melhores e ↓ que menores resultados são melhores. Os melhores resultados em cada coluna estão destacados em negrito.

T_{max}	IDF1↑	MOTA↑	MOTP↑	MT↑	FP↓	FN↓	IDS↓
30	28.7%	-51.3%	68.5%	**485**	118523	**36277**	2321
25	30.3%	-34.9%	68.5%	**485**	101436	36382	2337
20	32.4%	-18.2%	68.5%	484	83891	36530	**2318**
15	34.5%	-1.1%	68.6%	480	65928	36745	2339
10	36.7%	16.1%	68.6%	470	47624	37084	2414
5	**38.1%**	33.9%	68.6%	452	28220	37746	2634
0	37.7%	**50.1%**	**68.9%**	357	**6311**	41386	4166

É possível perceber que T_{max} influencia a capacidade do sistema de evitar a fragmentação da trajetória das faces em troca de erros de posicionamento quando o rastreador perde o alvo. Para $T_{max} = 30$, nota-se que a capacidade do sistema de manter a trajetória das faces detectadas é melhor, representada pelo maior valor de MT e segundo menor número de IDS. No entanto, este experimento também apresenta os piores valores de MOTA e FP.

Por outro lado, quando $T_{max} = 0$, os valores de MOTP, MOTA e FP são os melhores, enquanto MT, IDS e FN são os piores entre todos os experimentos. Os resultados desses

dois experimentos, $T_{max} = 30$ e $T_{max} = 0$, apontam para uma relação de troca inversamente proporcional onde é necessário balancear a capacidade do sistema de rastrear as trajetórias por mais tempo, sem sacrificar a acurácia das detecções.

Por esse motivo, admite-se que $T_{max} = 5$ seja o valor que equilibra melhor os resultados. A Figura 5.9 demonstra em gráficos o comportamento das métricas em relação aos valores testados de T_{max}. Constata-se uma melhora substancial nos valores de MT e IDS quando $T_{max} = 5$, enquanto que para valores subsequentes (onde $T_{max} = 10, 15, 20, 25$ e 30) os ganhos inferiores em MT e IDS não justificam a deterioração de MOTA e FP.

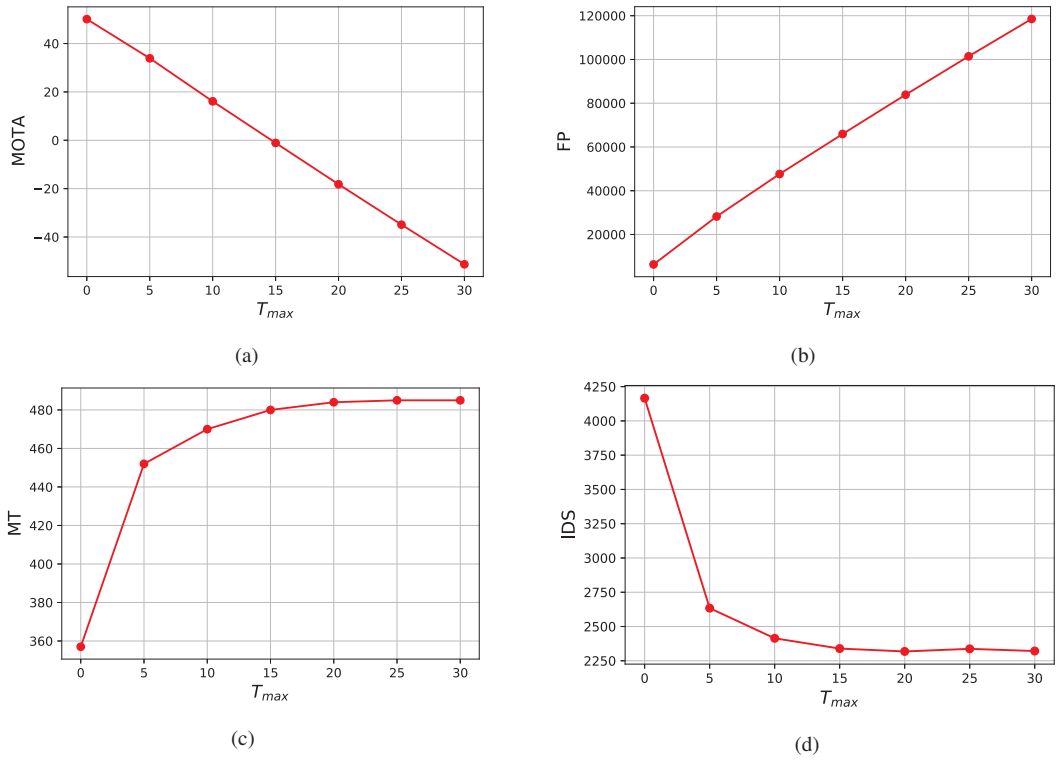


Figura 5.9: Gráficos das métricas resultantes de acordo com a variação do valor T_{max}, é possível perceber que MOTA e FP tem uma relação linear com T_{max}, enquanto MT e IDS apresentam ganho substancial no intervalo de 0 a 5. (a) Gráfico de MOTA por T_{max}; (b) Gráfico de FP por T_{max}; (c) Gráfico de MT por T_{max}; (d) Gráfico de IDS por T_{max}.

5.4.2 Avaliando os Limiares de Nitidez

Nesta subseção avalia-se como os limiares de nitidez afetam o módulo de Reconexão de Faces Baseado em *Tracklets* (FBTR) do sistema LTFT. O módulo FBTR avalia a qualidade de cada face detectada e as separa em uma entre três categorias, são elas: face descartada, face verificável e face de registro. Em seguida, o módulo FBTR utiliza um modelo de rede neural (Deng et al., 2019) para extrair as características das faces verificáveis e de registro e reconectar *tracklets* que pertençam a mesma identidade.

As faces são separadas em cada categoria de acordo com medidas de qualidade, como pose, confiança de detecção e nitidez da face. Diante disso, é imprescindível que a escolha dos limiares para cada categoria seja abrangente o bastante para a inclusão de várias faces por *tracklet*, sem ser irrestrita de forma a incluir faces com qualidade visual comprometida.

Os limiares de nitidez propostos no *baseline* são 0.9 para faces de registro (n_E) e 0.75 para faces verificáveis (n_V). No entanto, como apresentado na Tabela 4.2, os valores obtidos

podem ser consideravelmente menores. Para examinar a influência dos limiares n_E e n_V, a Tabela 5.7 exibe as métricas obtidas com a redução dos limiares de nitidez. Ademais, a Figura 5.10 elenca a quantia de faces de registro e faces verificáveis catalogadas pelo sistema para cada combinação dos valores de n_E e n_V.

Tabela 5.7: Experimentos com a variação dos parâmetros n_E e n_V do sistema LTFT utilizado como *baseline*. ↑ indica que maiores resultados são melhores e ↓ que menores resultados são melhores. Os melhores resultados em cada coluna estão destacados em negrito e os valores utilizados no *baseline* estão sublinhados.

n_E	n_V	IDF1↑	MOTA↑	MOTP↑	MT↑	FP↓	FN↓	IDS↓
<u>0.9</u>	<u>0.75</u>	38.1%	**33.9%**	**68.6%**	**452**	28220	**37746**	2634
0.65	0.5	38.1%	**33.9%**	**68.6%**	**452**	28220	**37746**	2634
0.35	0.2	39.0%	33.6%	**68.6%**	451	28200	38107	2626
0.25	0.1	40.4%	33.6%	**68.6%**	451	28196	38155	2613
0.1	0.05	42.3%	33.1%	**68.6%**	445	28099	38786	2573
0.08	0.04	43.8%	33.0%	**68.6%**	444	28052	38960	2538
0.06	0.04	**44.5%**	32.5%	**68.6%**	435	27998	39566	2503
0.03	0.015	**44.5%**	31.5%	68.5%	420	**27686**	41048	**2421**

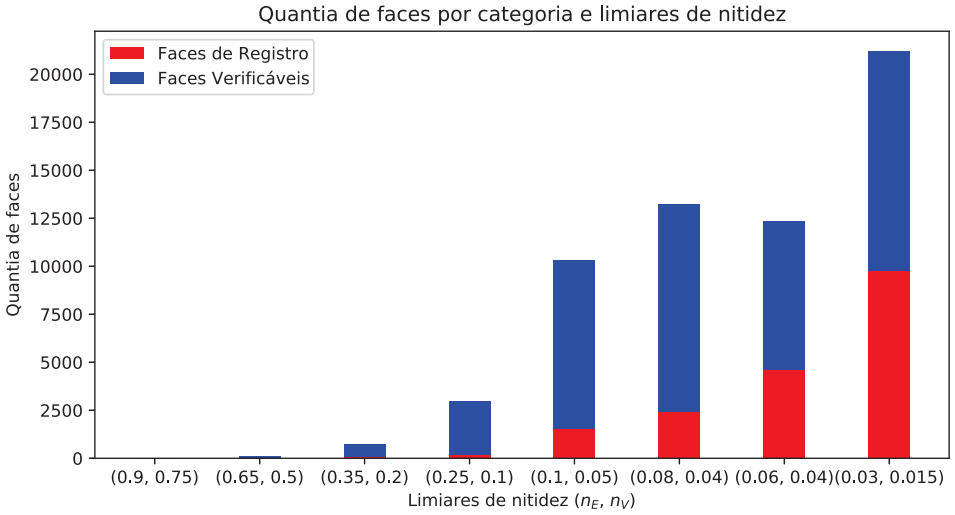


Figura 5.10: Gráfico relacionando o número de faces de registro e verificáveis catalogadas pelo sistema LTFT para cada combinação dos valores de n_E e n_V.

Conforme diminui-se os limiares n_E e n_V o sistema LTFT cataloga mais faces nas categorias verificáveis e de registro, o que permite que o módulo FBTR realize mais reconexões de *tracklets* pertencentes a mesma identidade. Esse aspecto é denotado pela elevação do valor de IDF1 e diminuição do valor de IDS, que indicam a capacidade do sistema de recuperar trajetórias com os identificadores corretos associados para cada identidade. No entanto, os valores de MOTA e FN também pioram conforme n_E e n_V são decrescidos.

Na Figura 5.11 é mais fácil perceber a relação de troca entre IDF1 e MOTA, assim como o ponto de retornos decrescentes, onde a partir de $n_E = 0.06$ e $n_V = 0.04$, IDF1 deixa de aumentar enquanto MOTA continua a decrescer. Em suma, esses experimentos demonstram que é possível obter 6.4 pontos percentuais a mais em IDF1 em troca de apenas 1.4 pontos percentuais em MOTA. Por esse motivo, argumenta-se que o sistema se comporta melhor com limiares de nitidez menores para esta medida em específico.

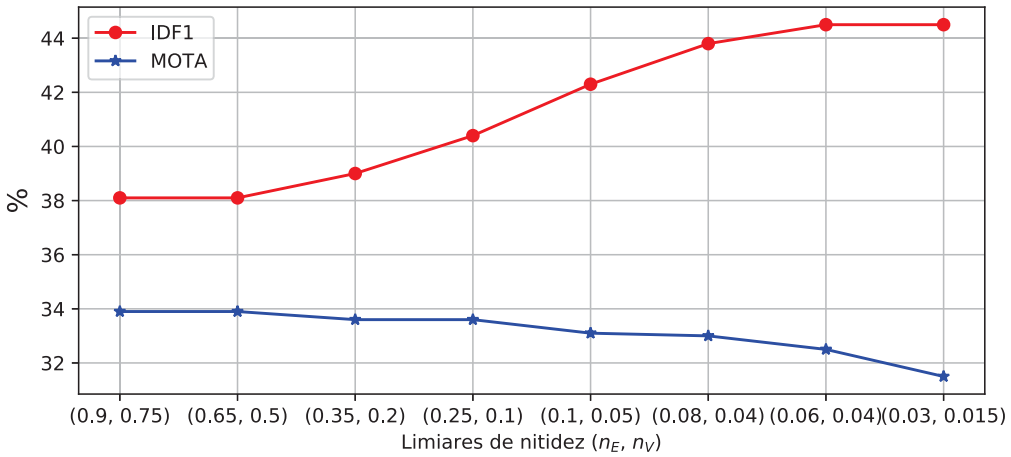


Figura 5.11: Gráfico que demonstra o comportamento das métricas IDF1 e MOTA de acordo com a variação dos parâmetros n_E e n_V.

5.5 ESTUDO COMPARATIVO

Nesta seção disserta-se sobre os experimentos realizados e quantifica-se o desempenho de cada um para a tarefa de rastreamento e seleção de faces. O sistema LTFT é comparado em seu estado original, como proposto por Barquero et. al (2021), com os resultados obtidos das alterações propostas. Os experimentos são descritos a seguir.

- LTFT: o sistema LTFT em seu estado *baseline*, utilizando o detector *Faceboxes* (Zhang et al., 2017) e parâmetros originais, com $T_{max} = 5$;
- LTFT+limiares: o mesmo que o LTFT *baseline*, com o detector *Faceboxes* e $T_{max} = 5$, porém com $n_E = 0.06$ e $n_V = 0.04$, como discutido na Subseção 5.4.2;
- LTFT+Yolo5s: neste experimento avalia-se a proposta de substituição do detector *Faceboxes* pelo modelo YOLO5Face (Qi et al., 2021);
- LTFT+Yolo5s+AQF: neste ultimo experimento, além da utilização do modelo YOLO5Face como detector, substitui-se o filtro Laplaciano modificado (Nikitin et al., 2014) pelo método de Nasrollahi e Moeslund (2011) para medida de nitidez, ademais, adiciona-se a medida da resolução do recorte de face para o Aferimento de Qualidade de Face (AQF).

Os parâmetros dos experimentos LTFT e LTFT+limiares foram explicados detalhadamente na Seção anterior 5.4. Quanto aos experimentos LTFT+Yolo5s e LTFT+Yolo5s+AQF, seus parâmetros foram encontrados por meio de exploração aleatória dentro de intervalos predefinidos, cada um executado por 250 iterações. Os parâmetros de cada experimento estão listados no Apêndice A.

A Tabela 5.8 demonstra os resultados de cada experimento avaliados em todo o *dataset*, isto é, a união do *dataset* desenvolvido por Barquero et al. (2021) e do *dataset* proposto neste trabalho.

Tabela 5.8: Resultados dos experimentos realizados em todo o *dataset*. ↑ indica que maiores resultados são melhores e ↓ que menores resultados são melhores. Os melhores resultados em cada coluna estão destacados em negrito.

Experimento	IDF1↑	MOTA↑	MOTP↑	MT↑	FP↓	FN↓	IDS↓
LTFT	38.1%	**33.9%**	68.6%	452	28220	**37746**	2634
LTFT+limiares	44.5%	32.5%	68.6%	435	27998	39566	2503
LTFT+Yolo5s	44.8%	33.2%	**78.7%**	483	27649	39650	2051
LTFT+Yolo5s+AQF	**47.1%**	33.5%	**78.7%**	**492**	**27647**	39356	**2038**

O uso do modelo YOLO5Face (Qi et al., 2021) para detecção de faces eleva a medida de precisão MOTP em 10.1% (de 68.6% para 78.7%), além de gradativamente melhorar os números de FP, IDS, MT e IDF1, visto que os 2 experimentos que utilizam YOLO5Face apresentam os melhores valores para essas 4 métricas.

O experimento LTFT+Yolo5s+AQF corrobora a eficácia das alterações das medidas de qualidade de face por meio dos melhores valores de IDF1, MOTP, MT, FP e IDS, e segundos melhores valores nas métricas MOTA e FN. Especificamente, os valores de IDF1, MT e IDS indicam que esse modelo é o que melhor atribui os identificadores para cada identidade além de conseguir rastrear consistentemente cada face, isto é, o modelo rastreia por mais tempo e com menos trocas de identificador.

No que tange a métrica de acurácia, quando comparados ao *baseline*, LTFT+limiares diminui seu valor de MOTA em 1.4% para aumentar IDF1 em 6.4%, enquanto que LTFT+Yolo5s+AQF atinge um acréscimo de 9% em IDF1 em troca de apenas 0.4% em MOTA. Acredita-se que essa variação negativa da métrica MOTA se deve ao número de FN que, apesar de ser o segundo menor em LTFT+Yolo5s+AQF, ainda é maior que o valor apresentado pelo *baseline*.

A Figura 5.12 demonstra exemplos de faces detectadas e separadas por qualidade pelo experimento LTFT+Yolo5s+AQF. A primeira e segunda linhas representam faces de registro e verificáveis, respectivamente. Espera-se que essas faces apresentem nitidez, resolução e pose apropriadas para o reconhecimento e reconexão de *tracklets*. A última linha exemplifica faces descartáveis, cuja qualidade é designada insuficiente e sua inclusão pode inserir ruídos na reconexão.

Figura 5.12: Exemplos de faces detectadas e separadas por qualidade pelo experimento LTFT+Yolo5s+AQF. A primeira, segunda e terceira linha representam faces de registro, verificáveis e descartadas, respectivamente. Para fins de apresentação todas as detecções foram redimensionadas para tamanhos iguais.

As Tabelas 5.9 e 5.10 apresentam os resultados de cada experimento quando avaliados separadamente no *dataset* proposto neste trabalho e no *dataset* desenvolvido por Barquero et al. (2021), respectivamente.

Tabela 5.9: Resultados dos experimentos realizados no conjunto de 3 vídeos retirados do *dataset* MOT17 e anotados manualmente neste trabalho. ↑ indica que maiores resultados são melhores e ↓ que menores resultados são melhores. Os melhores resultados em cada coluna estão destacados em negrito.

Experimento	IDF1↑	MOTA↑	MOTP↑	MT↑	FP↓	FN↓	IDS↓
LTFT	**13.0%**	3.3%	64.9%	7	1577	20169	**87**
LTFT+limiares	**13.0%**	3.3%	64.9%	7	1577	20169	**87**
LTFT+Yolo5s	12.7%	**9.9%**	**76.3%**	**11**	**556**	**19677**	111
LTFT+Yolo5s+AQF	12.7%	**9.9%**	**76.3%**	**11**	**556**	**19677**	111

Dentre os três videos anotados no *dataset* proposto, dois são particularmente desafiadores devido às baixas resoluções das faces e má iluminação. Neste caso, os experimentos equipados com o YOLO5Face apresentam melhor acurácia e precisão multiobjeto, como demonstrado pelas métricas MOTA, MOTP, FP e FN. Isso pode indicar que o sistema se comporta melhor em sequências de vídeo não previamente vistas quando é implementado com YOLO5Face.

A Figura 5.13 é um quadro da sequência MOT17-04 do *dataset* proposto. Esta sequência é desafiadora devido à distância das faces da câmera e a variação de luz ao longo da cena. Nesta figura estão destacados os retângulos envolventes do *groundtruth*, do *baseline* e do experimento LTFT+Yolo5s+AQF.

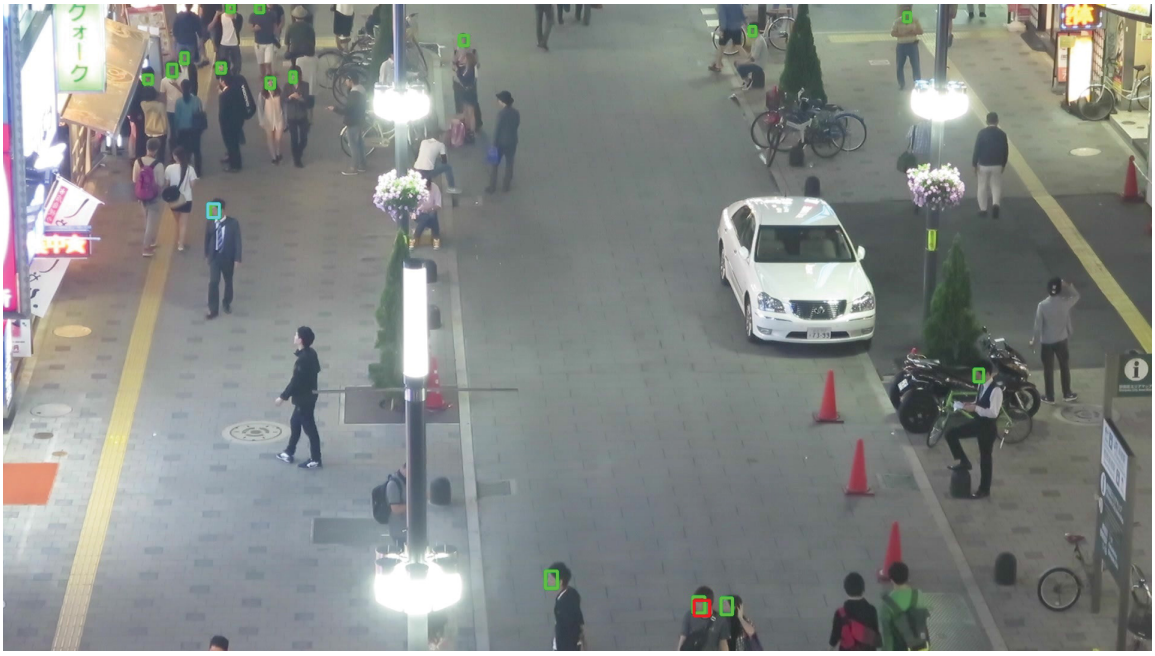


Figura 5.13: Quadro da sequência MOT17-04, anotado como parte do *dataset* suplementar proposto. Os retângulos em verde, vermelho e azul claro são referentes, respectivamente, ao *groundtruth*, *baseline* e experimento LTFT+Yolo5s+AQF.

Tabela 5.10: Resultados dos experimentos realizados no conjunto de vídeos proposto por Barquero et al. (2021). ↑ indica que maiores resultados são melhores e ↓ que menores resultados são melhores. Os melhores resultados em cada coluna estão destacados em negrito.

Experimento	IDF1↑	MOTA↑	MOTP↑	MT↑	FP↓	FN↓	IDS↓
LTFT	42.0%	**42.5%**	68.8%	445	26643	**17577**	2547
LTFT+limiares	49.4%	40.7%	68.7%	428	**26421**	19397	2416
LTFT+Yolo5s	49.7%	39.7%	**78.8%**	472	27093	19973	1940
LTFT+Yolo5s+AQF	**52.3%**	40.1%	**78.8%**	**481**	27091	19679	**1927**

A Tabela 5.10 expressa o desempenho dos experimentos no *dataset* desenvolvido por Barquero et al. (2021). Nela é possível perceber que os experimentos que utilizam *Faceboxes* (Zhang et al., 2017) como detector apresentam as melhores métricas de acurácia, como MOTA, FP e FN. Esse resultado é esperado, visto que *Faceboxes* também foi utilizado no processo de anotação deste mesmo *dataset*.

No entanto, pode-se argumentar que a diferença de 2.4% de MOTA é ofuscada pelo ganho de 10% em MOTP e 10.3% em IDF1. Ou seja, LTFT+Yolo5s+AQF atribui retângulos envolventes mais precisos, o que robustece o rastreamento quando uma face é perdida e sua última posição é utilizada para iniciar o rastreador KCF (Henriques et al., 2015). Além disso, as medidas de qualidade propostas também aparentam fornecer detecções mais apropriadas para a reconexão dos *tracklets*. Ambas afirmações são favorecidas pelas métricas IDS e MT, onde no experimento LTFT+Yolo5s+AQF ocorrem 620 trocas de identificadores (IDS) a menos e 36 faces a mais são rastreadas por pelo menos 80% de seu trajeto total (MT).

Em suma, é possível melhorar a consistência de identificação e rastreamento do sistema LTFT por meio do detector YOLO5Face (Qi et al., 2021), da substituição da medida de nitidez pela proposta por Nasrollahi e Moeslund (2011) e pela adição da resolução como métrica de qualidade adicional.

As Figuras 5.14, 5.15 e 5.16 demonstram para fins comparativos, respectivamente, anotações do *groundtruth*, resultados do *baseline* e resultados do experimento LTFT+Yolo5s+AQF em quatro quadros da sequência de vídeo Terminal1.

Figura 5.14: Quatro quadros da sequência Terminal1, os retângulos verdes se referem ás anotações de *groundtruth*.

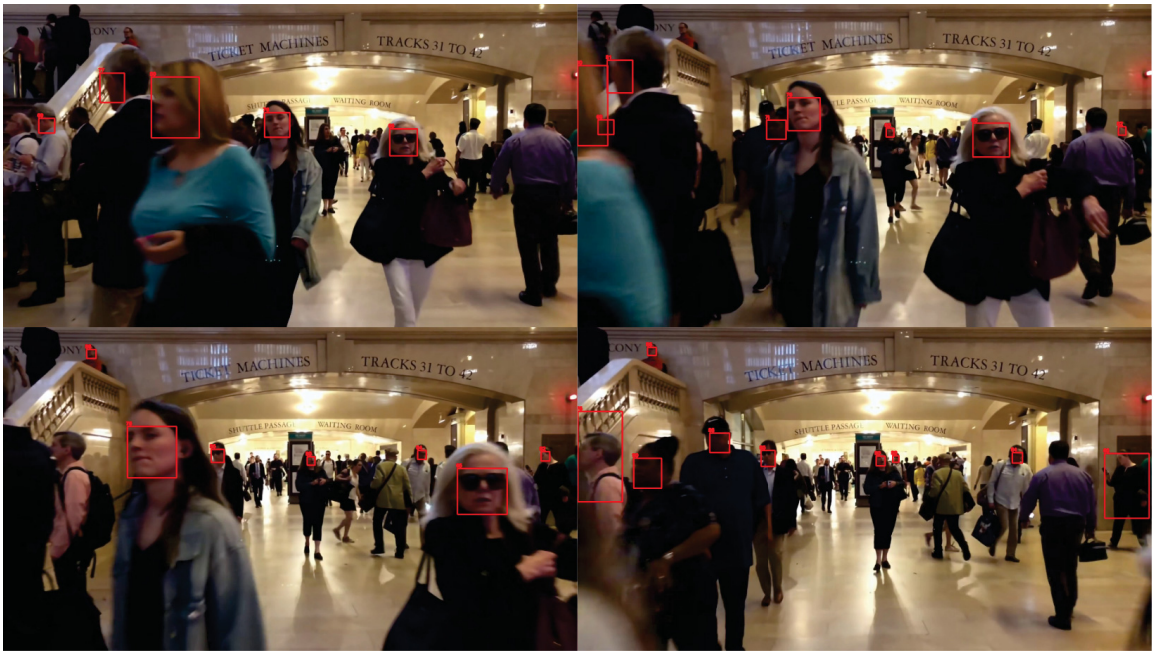


Figura 5.15: Resultados do *baseline* em quatro quadros da sequência Terminal1.

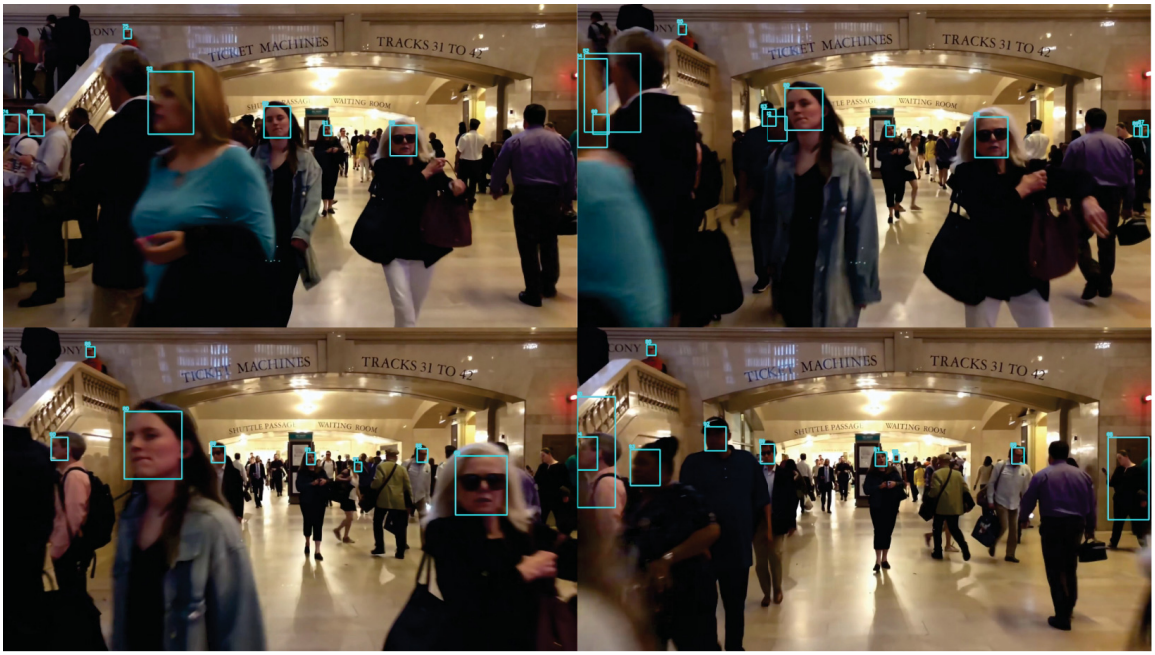


Figura 5.16: Resultados do experimento proposto, LTFT+Yolo5s+AQF, em quatro quadros da sequência Terminal1.

Acredita-se que os resultados de acurácia podem ser aperfeiçoados se o *dataset* for refinado para remover os ruídos discutidos na Seção 4.2. Além disso, nota-se que o desempenho destas abordagens na tarefa de rastreamento de faces (especialmente quando as faces estão distantes, como demonstrado pela sequência MOT17-04) ainda deixa a desejar se comparado aos resultados em outras tarefas de Visão Computacional, como detecção de faces ou rastreamento de pessoas. Acredita-se que, um dos motivos é a pequena quantia de *datasets* apropriadamente anotados para o rastreamento de múltiplas faces, e que anotar sequências com desafios de resolução, iluminação e baixas taxas de quadros são os próximos passos para compreender as

dificuldades de aplicação do rastreamento, seleção e reconhecimento de faces em cenários de vigilância.

Ademais, dados os resultados de rastreamento no *dataset* proposto, onde o melhor resultado de acurácia multiobjeto foi 9.9%, é possível deduzir boas práticas para a implementação do rastreamento de múltiplas faces, como: para evitar a necessidade de implementação de tecnologias suplementares, como super resolução, é imprescindível que as câmeras de interesse estejam posicionadas o mais próximo possível da altura dos olhos das pessoas que transitam na região; e a iluminação deve ser uniforme, visto que os melhores resultados foram obtidos em corredores bem iluminados (como nas sequências Choke1 e Choke2) ou em ruas durante o dia (como nas sequências Bengal e Street).

Não obstante, tópicos como reconhecimento facial e videovigilância são áreas sensíveis quanto a privacidade e liberdade das pessoas monitoradas. A obtenção de informações visuais e processamento destes dados devem seguir boas práticas e princípios éticos para garantir a segurança e privacidade de todos os envolvidos.

6 CONCLUSÃO

Sistemas de videovigilância utilizam câmeras de vídeo para assegurar a segurança de pessoas e propriedades por meio de gravações em locais de interesse. Devido a grande quantia de dados gerados por esses sistemas, existe um incentivo em automatizar a extração e interpretação dos dados visuais obtidos por essas câmeras. Rastreamento e reconhecimento de faces são temas relacionados dentro da área de Visão Computacional que podem ser aplicados à videovigilância para a criação de um sistema de vigilância inteligente.

No entanto, a análise de múltiplas faces simultâneas em vídeo ainda apresenta desafios como baixa resolução de faces, troca de identificadores e iluminação e pose irregulares. Apesar de várias abordagens terem sido propostas para rastrear e selecionar faces em vídeo, este tema ainda requer instrumentos elementares como um *dataset* vasto, unificado e publicamente disponível.

No presente trabalho estudou-se a seleção de faces e sua importância. O sistema e *dataset* apresentados por Barquero et al. (2021) foram utilizados como *baseline* e expandidos ao longo deste trabalho. O *dataset* foi enriquecido com três vídeos do conjunto MOT17 (Milan et al., 2016) rotulados com as posições e identificadores das faces. Além disso, novos modelos para detecção de faces foram analisados, de forma que o modelo CNN YOLO5Face (Qi et al., 2021) foi o selecionado. Por fim, a medida de nitidez foi substituída por outra abordagem e a resolução das faces detectadas foi empregada para o aferimento de qualidade.

O sistema modificado resultou em um aumento de 10.1% na métrica de precisão multiobjeto (MOTP) e 9% a mais na métrica de identificação IDF1. Esses resultados aliados ao maior número de *Mostly Tracked* (MT) e menor quantia de ID *Switches* (IDS), indicam que o sistema proposto consegue rastrear faces por mais tempo, sua política de seleção de faces baseada em qualidade auxilia o reconhecimento de faces e, consequentemente, a reconexão de *tracklets* do sistema. Embora, a métrica MOTA foi negativamente afetada por uma diferença de 0.4% quando comparada ao *baseline*.

Para trabalhos futuros recomenda-se refinar e expandir os *datasets* utilizados neste trabalho para que se construa um conjunto de testes unificado (assim como existem *benchmarks* bem estabelecidos para treino e testes em outras áreas da Visão Computacional). O sistema tratado neste trabalho também pode beneficiar-se de melhoras em seu tempo de execução, de forma a tornar cada vez mais factível a aplicação da seleção de faces para videovigilância.

Não obstante, um dos aspectos a ser considerado neste trabalho é a complexidade de sua implementação devido à dependência de múltiplos modelos. Portanto, há a necessidade de unificar todo o processo em uma única rede, capaz de detectar, rastrear e reconhecer faces em vídeo. Uma sugestão é o uso de redes recorrentes com camadas *Long Short Term Memory* (LSTM), devido sua capacidade de analisar o aspecto temporal inerente a gravações de vídeo.

REFERÊNCIAS

Alzubaidi, L., Zhang, J., Humaidi, A. J., Al-Dujaili, A., Duan, Y., Al-Shamma, O., Santamaría, J., Fadhel, M. A., Al-Amidie, M. e Farhan, L. (2021). Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of Big Data*, 8(1):53.

Arachchilage, S. W. e Izquierdo, E. (2020). Adaptive aggregated tracklet linking for multi-face tracking. Em *2020 IEEE International Conference on Image Processing (ICIP)*, páginas 1366–1370.

Ashby, M. P. J. (2017). The value of cctv surveillance cameras as an investigative tool: An empirical analysis. *European Journal on Criminal Policy and Research*, 23(3):441–459.

Bagdanov, A. D., Del Bimbo, A., Dini, F., Lisanti, G. e Masi, I. (2012). Posterity logging of face imagery for video surveillance. *IEEE MultiMedia*, 19(4):48–59.

Barquero, G., Hupont, I. e Fernández Tena, C. (2021). Rank-based verification for long-term face tracking in crowded scenes. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 3(4):495–505.

Barra, P., Barra, S., Bisogni, C., De Marsico, M. e Nappi, M. (2020). Web-shaped model for head pose estimation: An approach for best exemplar selection. *IEEE Transactions on Image Processing*, 29:5457–5468.

Bergmann, P., Meinhardt, T. e Leal-Taixé, L. (2019). Tracking without bells and whistles. Em *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, páginas 941–951.

Bernardin, K. e Stiefelhagen, R. (2008). Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008.

Bewley, A., Ge, Z., Ott, L., Ramos, F. e Upcroft, B. (2016). Simple online and realtime tracking. Em *2016 IEEE International Conference on Image Processing (ICIP)*, páginas 3464–3468.

Boom, B., Beumer, G., Spreeuwers, L. e Veldhuis, R. N. J. (2006). The effect of image resolution on the performance of a face recognition system. Em *2006 9th International Conference on Control, Automation, Robotics and Vision*, páginas 1–6.

Bradski, G. R. (1998). Computer vision face tracking for use in a perceptual user interface.

Brasó, G. e Leal-Taixé, L. (2020). Learning a neural solver for multiple object tracking. Em *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 6246–6256.

Cai, Y. e Gan, H. (2019). An online face clustering algorithm for face monitoring and retrieval in real-time videos. Em *2019 IEEE Intl Conf on Parallel Distributed Processing with Applications, Big Data Cloud Computing, Sustainable Computing Communications, Social Computing Networking (ISPA/BDCloud/SocialCom/SustainCom)*, páginas 825–830.

Camps, O., Cucchiara, R., Bimbo, A. D., Matas, J., Pernici, F. e Sclaroff, S. (2014). LTDT2014: Long-Term Detection and Tracking. CVPR2014, Columbus, Ohio. `http://www.micc.unifi.it/LTDT2014/organization/index.html`. Acessado em 21/03/2021.

Chen, T.-W., Hsu, S.-C. e Chien, S.-Y. (2007). Automatic feature-based face scoring in surveillance systems. Em *Ninth IEEE International Symposium on Multimedia (ISM 2007)*, páginas 139–146.

Chi, C., Zhang, S., Xing, J., Lei, Z., Li, S. e Zou, X. (2019). Selective refinement network for high performance face detection. Em *Association for the Advancement of Artificial Intelligence (AAAI)*.

Clarke, R. V. (1983). Situational crime prevention: Its theoretical basis and practical scope. *Crime and Justice*, 4:225–256.

De Marsico, M., Nappi, M. e Riccio, D. (2014). Es-ru: An entropy based rule to select representative templates in face surveillance. *Multimedia Tools Appl.*, 73(1):109–128.

Del Bimbo, A., Dini, F. e Lisanti, G. (2009). A real time solution for face logging. Em *3rd International Conference on Imaging for Crime Detection and Prevention (ICDP 2009)*, páginas 1–6.

Deng, J., Guo, J., Ververas, E., Kotsia, I. e Zafeiriou, S. (2020). Retinaface: Single-shot multi-level face localisation in the wild. Em *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 5202–5211.

Deng, J., Guo, J., Xue, N. e Zafeiriou, S. (2019). Arcface: Additive angular margin loss for deep face recognition. Em *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 4685–4694.

Felzenszwalb, P. F., Girshick, R. B., McAllester, D. e Ramanan, D. (2010). Object detection with discriminatively trained part-based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645.

Ferryman, J. e Shahrokni, A. (2009). Pets2009: Dataset and challenge. Em *2009 Twelfth IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, páginas 1–6.

Fisher, R. (2003). Caviar: Context aware vision using image-based active recognition. `http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/`. Acessado em 15/02/2021.

Gejguš, P. e Šperka, M. (2003). Face tracking in color video sequences. Em *Proceedings of the 19th Spring Conference on Computer Graphics*, SCCG '03, página 245–249, New York, NY, USA. Association for Computing Machinery.

Goodfellow, I., Bengio, Y. e Courville, A. (2016). *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Hannane, R., Elboushaki, A. e Afdel, K. (2015). An automatic video surveillance indexing based on facial feature descriptors. Em *2015 5th International Conference on Information Communication Technology and Accessibility (ICTA)*, páginas 1–6.

Henriques, J. F., Caseiro, R., Martins, P. e Batista, J. (2015). High-speed tracking with kernelized correlation filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):583–596.

i-LIDS Team (2006). Imagery library for intelligent detection systems (i-lids); a standard for testing video based detection systems. Em *Proceedings 40th Annual 2006 International Carnahan Conference on Security Technology*, páginas 75–80.

Jie Yang e Waibel, A. (1996). A real-time face tracker. Em *Proceedings Third IEEE Workshop on Applications of Computer Vision. WACV'96*, páginas 142–147.

Kim, H.-I., Lee, S. H. e Ro, Y. M. (2014). Investigating cascaded face quality assessment for practical face recognition system. Em *2014 IEEE International Symposium on Multimedia*, páginas 399–400.

Kristan, M., Pflugfelder, R., Leonardis, A., Matas, J., Porikli, F., Cehovin, L., Nebehay, G., Fernandez, G., Vojir, T., Gatt, A., Khajenezhad, A., Salahledin, A., Soltani-Farani, A., Zarezade, A., Petrosino, A., Milton, A., Bozorgtabar, B., Li, B., Chan, C. S., Heng, C., Ward, D., Kearney, D., Monekosso, D., Karaimer, H. C., Rabiee, H. R., Zhu, J., Gao, J., Xiao, J., Zhang, J., Xing, J., Huang, K., Lebeda, K., Cao, L., Maresca, M. E., Lim, M. K., El Helw, M., Felsberg, M., Remagnino, P., Bowden, R., Goecke, R., Stolkin, R., Lim, S. Y., Maher, S., Poullot, S., Wong, S., Satoh, S., Chen, W., Hu, W., Zhang, X., Li, Y. e Niu, Z. (2013). The visual object tracking vot2013 challenge results. Em *2013 IEEE International Conference on Computer Vision Workshops*, páginas 98–111.

Krizhevsky, A., Sutskever, I. e Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60(6):84–90.

Kuhn, H. W. (1955). The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97.

Leal-Taixé, L., Milan, A., Reid, I., Roth, S. e Schindler, K. (2015). Motchallenge 2015: Towards a benchmark for multi-target tracking.

LI, H.-Y., XIA, P.-P. e XU, H.-L. (2017). The study on the model of effective face retrieval in pedestrian detection. Em *Proceedings of the 3rd Annual International Conference on Electronics, Electrical Engineering and Information Science (EEEIS 2017)*, páginas 306–312. Atlantis Press.

Li, J., Wang, Y., Wang, C., Tai, Y., Qian, J., Yang, J., Wang, C., Li, J. e Huang, F. (2019). Dsfd: Dual shot face detector. Em *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 5055–5064.

Lin, T., Dollár, P., Girshick, R., He, K., Hariharan, B. e Belongie, S. (2017). Feature pyramid networks for object detection. Em *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, páginas 936–944.

Lin, Y., Cheng, S., Shen, J. e Pantic, M. (2019). Mobiface: A novel dataset for mobile face tracking in the wild. Em *2019 14th IEEE International Conference on Automatic Face Gesture Recognition (FG 2019)*, páginas 1–8.

Liu, Q. e Cai, C. (2006). Dual searching window based face tracking. Em *2006 International Symposium on Intelligent Signal Processing and Communications*, páginas 279–282.

Lucas, B. D. e Kanade, T. (1981). An iterative image registration technique with an application to stereo vision. Em *Proceedings of the 7th International Joint Conference on Artificial*

Intelligence - Volume 2, IJCAI'81, página 674–679, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W. e Kim, T.-K. (2021). Multiple object tracking: A literature review. *Artificial Intelligence*, 293:103448.

Marciniak, T., Chmielewska, A., Weychan, R., Parzych, M. e Dabrowski, A. (2015). Influence of low resolution of images on reliability of face detection and recognition. *Multimedia Tools and Applications*, 74(12):4329–4349.

Mathias, M., Benenson, R., Pedersoli, M. e Van Gool, L. (2014). Face detection without bells and whistles. Em *Computer Vision – ECCV 2014. ECCV 2014. Lecture Notes in Computer Science*, volume 8692, páginas 720–735.

Mau, S., Chen, S., Sanderson, C. e Lovell, B. C. (2010). Video face matching using subset selection and clustering of probabilistic multi-region histograms. Em *2010 25th International Conference of Image and Vision Computing New Zealand*, páginas 1–8.

Milan, A., Leal-Taixe, L., Reid, I., Roth, S. e Schindler, K. (2016). Mot16: A benchmark for multi-object tracking.

Momin, B. F. e Jere, Y. (2015). Mining visitors in video surveillance system. Em *2015 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)*, páginas 1–4.

Nasrollahi, K. e Moeslund, T. B. (2008). *Face Quality Assessment System in Video Sequences*, página 10–18. Springer-Verlag, Berlin, Heidelberg.

Nasrollahi, K. e Moeslund, T. B. (2010a). Face log generation for super resolution using local maxima in the quality curve. Em *Proceedings of the International Conference on Computer Vision Theory and Applications - Volume 1: VISAPP, (VISIGRAPP 2010)*, páginas 124–129. INSTICC, SciTePress.

Nasrollahi, K. e Moeslund, T. B. (2010b). Hybrid super resolution using refined face logs. Em *2010 2nd International Conference on Image Processing Theory, Tools and Applications*, páginas 435–440.

Nasrollahi, K. e Moeslund, T. B. (2011). Extracting a good quality frontal face image from a low-resolution video sequence. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10):1353–1362.

Nghiem, A. T., Bremond, F., Thonnat, M. e Valentin, V. (2007). Etiseo, performance evaluation for video surveillance systems. Em *2007 IEEE Conference on Advanced Video and Signal Based Surveillance*, páginas 476–481.

Nikitin, , Konushin, A. e Konushin, V. (2014). Face quality assessment for face verification in video. Em *The 24th International Conference on Computer Graphics and Vision (GraphiCon2014)*, páginas 111–114.

Oh, S., Hoogs, A., Perera, A., Cuntoor, N., Chen, C., Lee, J. T., Mukherjee, S., Aggarwal, J. K., Lee, H., Davis, L., Swears, E., Wang, X., Ji, Q., Reddy, K., Shah, M., Vondrick, C., Pirsiavash, H., Ramanan, D., Yuen, J., Torralba, A., Song, B., Fong, A., Roy-Chowdhury, A. e Desai, M. (2011). A large-scale benchmark dataset for event recognition in surveillance video. Em *CVPR 2011*, páginas 3153–3160.

Park, Y., Dang, L. M., Lee, S., Han, D. e Moon, H. (2021). Multiple object tracking in deep learning approaches: A survey. *Electronics*, 10(19).

Patino, L. e Ferryman, J. (2014). Pets 2014: Dataset and challenge. Em *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, páginas 355–360.

Patino, L., Nawaz, T., Cane, T. e Ferryman, J. (2017). Pets 2017: Dataset and challenge. Em *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, páginas 2126–2132.

Pers, J. e Magee, D. R. (2006). CVBASE '06 - Workshop on Computer Vision Based Analysis in Sport Environments. `http://vision.fe.uni-lj.si/cvbase06/`. Acessado em 21/03/2021.

Piza, E., Welsh, B., Farrington, D. e Thomas, A. (2019). Cctv surveillance for crime prevention: A 40-year systematic review with meta-analysis. *Criminology & Public Policy*, 18:135–159.

Qi, D., Tan, W., Yao, Q. e Liu, J. (2021). Yolo5face: Why reinventing a face detector.

Qi, X., Liu, C. e Schuckers, S. (2018). Boosting face in video recognition via cnn based key frame extraction. Em *2018 International Conference on Biometrics (ICB)*, páginas 132–139.

Qiang Liu, Canhui Cai, Ngan, K. N. e Hongliang Li (2007). Camshift based real-time multiple faces match tracking. Em *2007 International Symposium on Intelligent Signal Processing and Communication Systems*, páginas 726–729.

Ren, S., He, K., Girshick, R. e Sun, J. (2017). Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149.

Ristani, E., Solera, F., Zou, R., Cucchiara, R. e Tomasi, C. (2016). Performance measures and a data set for multi-target, multi-camera tracking. Em Hua, G. e Jégou, H., editores, *Computer Vision – ECCV 2016 Workshops*, páginas 17–35, Cham. Springer International Publishing.

Sekachev, B., Manovich, N., Zhiltsov, M., Zhavoronkov, A., Kalinin, D., Hoff, B., TOsmanov, Kruchinin, D., Zankevich, A., DmitriySidnev, Markelov, M., Johannes222, Chenuet, M., a andre, telenachos, Melnikov, A., Kim, J., Ilouz, L., Glazov, N., Priya4607, Tehrani, R., Jeong, S., Skubriev, V., Yonekura, S., vugia truong, zliang7, lizhming e Truong, T. (2020). opencv/cvat: v1.1.0.

Shen, J., Zafeiriou, S., Chrysos, G. G., Kossaifi, J., Tzimiropoulos, G. e Pantic, M. (2015). The first facial landmark tracking in-the-wild challenge: Benchmark and results. Em *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, páginas 1003–1011.

Simonyan, K. e Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556.

Sobottka, K. e Pitas, I. (1996). Extraction of facial regions and features using color and shape information. Em *Pattern Recognition, International Conference on*, volume 3, páginas 421,422,423,424,425, Los Alamitos, CA, USA. IEEE Computer Society.

Vadakkepat, P., Lim, P., De Silva, L. C., Jing, L. e Ling, L. L. (2008). Multimodal approach to human-face detection and tracking. *IEEE Transactions on Industrial Electronics*, 55(3):1385–1393.

Vignesh, S., Priya, K. M. e Channappayya, S. S. (2015). Face image quality assessment for face selection in surveillance video using convolutional neural networks. Em *2015 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, páginas 577–581.

Viola, P. e Jones, M. (2004). Robust real-time face detection. *International Journal of Computer Vision*, 57:137–154.

Wang, J., Kankanhalli, M. S., Yan, W. e Jain, R. (2003). Experiential sampling for video surveillance. Em *First ACM SIGMM International Workshop on Video Surveillance*, IWVS '03, página 77–86, New York, NY, USA. Association for Computing Machinery.

Wong, Y., Chen, S., Mau, S., Sanderson, C. e Lovell, B. C. (2011). Patch-based probabilistic image quality assessment for face selection and improved video-based face recognition. Em *IEEE Biometrics Workshop, Computer Vision and Pattern Recognition (CVPR) Workshops*, páginas 81–88. IEEE.

Yan, J., Lei, Z., Wen, L. e Li, S. Z. (2014). The fastest deformable part model for object detection. Em *2014 IEEE Conference on Computer Vision and Pattern Recognition*, páginas 2497–2504.

Yang, S., Luo, P., Loy, C. C. e Tang, X. (2016). WIDER FACE: A face detection benchmark. Em *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Zhang, S., Chi, C., Lei, Z. e Li, S. Z. (2020). Refineface: Refinement neural network for high performance face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, páginas 1–1.

Zhang, S., Zhu, X., Lei, Z., Shi, H., Wang, X. e Li, S. Z. (2017). Faceboxes: A cpu real-time face detector with high accuracy. Em *2017 IEEE International Joint Conference on Biometrics (IJCB)*, páginas 1–9.

Zheng, J., Ranjan, R., Chen, C.-H., Chen, J.-C., Castillo, C. D. e Chellappa, R. (2020). An automatic system for unconstrained video-based face recognition. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 2(3):194–209.

Zhou, X., Koultun, V. e Krähenbühl, P. (2020). Tracking objects as points. Em *Computer Vision – ECCV 2020. ECCV 2020. Lecture Notes in Computer Science*, volume 12349, páginas 474–490.

Zhu, C., Zheng, Y., Luu, K. e Savvides, M. (2017). Cms-rcnn: Contextual multi-scale region-based cnn for unconstrained face detection. Em *Deep Learning for Biometrics. Advances in Computer Vision and Pattern Recognition (ACVPR)*, páginas 57–79.

Zhu, X., Liu, X., Lei, Z. e Li, S. Z. (2019). Face alignment in full pose range: A 3d total solution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(1):78–92.

Zhu, Y., Cai, H., Zhang, S., Wang, C. e Xiong, Y. (2020). Tinaface: Strong but simple baseline for face detection.

APÊNDICE A – PARÂMETROS DOS EXPERIMENTOS

Este apêndice descreve os parâmetros utilizados nos experimentos realizados no Capítulo 5. Primeiramente, existem parâmetros que são fixados de acordo com o trabalho de Barquero et al. (2021) e são os mesmos em todos os experimentos, são eles: $\lambda_{IOU} = 0.25$, $\lambda_{FBTR} = 0.5$, $\epsilon = 0.8$ e $C = 6$. Além disso os ângulos para categorização da qualidade das faces também mantém-se os mesmos, ±25° para faces de registro e ±60° para faces verificáveis.

Na Seção 5.4 do Capítulo 5 é discutida a dependência do sistema na utilização de um rastreador genérico. Esta dependência é representada pelo parâmetro T_{max}, que controla por quantos quadros uma face é rastreada quando não pôde ser detectada. De acordo com os experimentos realizados na Seção 5.4, decidiu-se por utilizar $T_{max} = 5$ para todos os experimentos subsequentes.

Para o módulo de reconexão, os parâmetros que variam para cada experimento são: confiança de detecção para faces de registro (d_E), confiança de detecção para faces verificáveis (d_V), limiar de nitidez para faces de registro (n_E) e limiar de nitidez para faces verificáveis (n_V). Outro parâmetro é a confiança de detecção mínima para contabilizar uma face, este é representado por λ_{det}. A Tabela A.1 relaciona os parâmetros citados anteriormente a três dentre os quatro experimentos.

Tabela A.1: Parâmetros utilizados para cada experimento.

Experimento	λ_{det}	d_E	d_V	n_E	n_V
LTFT	0.75	0.95	0.8	0.9	0.75
LTFT+limiares	0.75	0.95	0.8	0.06	0.04
LTFT+Yolo5s	0.7	0.83	0.76	0.06	0.04

O experimento LTFT+Yolo5s+AQF fica reservado a Tabela A.2 devido a dois parâmetros adicionais. Neste caso, a qualidade das faces também é analisada com relação a resolução da detecção, onde Res_E é a resolução mínima para faces de registro e Res_V é a resolução mínima para faces verificáveis.

Tabela A.2: parâmetros utilizados no experimento LTFT+Yolo5s+AQF.

Experimento	λ_{det}	d_E	d_V	n_E	n_V	Res_E	Res_V
LTFT+Yolo5s+AQF	0.7	0.84	0.7	82	40	87	28