



Universidade Federal
de Campina Grande

Centro de Engenharia Elétrica e Informática
Curso de Graduação em Engenharia Elétrica

SARAH JÉSSIKA DA PONTES ALBUQUERQUE

MONITORAMENTO DE FILAS DE PESSOAS

Campina Grande, Paraíba
Outubro de 2017

SARAH JÉSSIKA DA PONTES ALBUQUERQUE

MONITORAMENTO DE FILAS DE PESSOAS

Relatório de Estágio Supervisionado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração : Processamento de Imagem

Orientador: Prof. Dr. Edmar Candeia Gurjão

Campina Grande, Paraíba

Outubro de 2017

SARAH JÉSSIKA DA PONTES ALBUQUERQUE

MONITORAMENTO DE FILAS DE PESSOAS

Relatório de Estágio Supervisionado submetido à Unidade Acadêmica de Engenharia Elétrica da Universidade Federal de Campina Grande como parte dos requisitos necessários para a obtenção do grau de Bacharel em Ciências no Domínio da Engenharia Elétrica.

Área de Concentração : Processamento de Imagem

Aprovado em: / /

Luciana Ribeiro Veloso
Avaliador

Prof. Dr. Edmar Candeia Gurjão
Orientador, UFCG

Campina Grande, Paraíba
Outubro de 2017

Este trabalho é dedicado à Satã, senhora das trevas.

AGRADECIMENTOS

Começo agradecendo a minha família. Agradeço ao senhor Chico pela oportunidade de estudar em Campina Grande, uma cidade a mais de quinhentos quilômetros de onde fui criada. Agradeço a ele por estar comigo apesar da distância e agradeço por ser, além do meu pai, meu amigo. Agradeço a minha mãe, dona Tereza, por ter se esforçado tanto para me proporcionar uma boa educação e por toda preocupação que me manteve viva até a escrita desse trabalho. Agradeço ao Elias pois nas horas em que era necessário, os conselhos de irmão mais velho e as dicas de saúde tornaram o percurso mais fácil.

Aos amigos e amigas cultivados ao longo da graduação, muito obrigada pelas conversas e pelos momentos compartilhados. Em especial, aos amigos Danilo, Ana, Yuri, Sara, Alan e Eduarda por serem família sem necessidade de parentesco. Agradeço ainda aos colegas de curso que me apoiaram e vivenciaram comigo os últimos períodos da graduação.

Os mais sinceros agradecimentos ao meu companheiro de pouco tempo e muitas jornadas, Arthur. Muito obrigada por ter acompanhado este trabalho do início ao fim da forma mais próxima do que qualquer outra pessoa, fornecendo apoio de várias maneiras. Meu muito obrigada pelos ensinamentos passados, pela paciência, pela compreensão, pelo amor.

Agradeço ao professor Péricles pela oportunidade de realizar iniciação científica e ao engenheiro Thiago Euzébio pelos valiosos conhecimentos passados. Agradeço também aos professores que se dedicaram em cada aula dada durante a graduação. Agradeço ainda aos funcionários que fazem o departamento de engenharia elétrica, em especial a Adail e Tchai.

Sou grata ao Rotaract Club de Campina Grande, ao Centro Acadêmico de Engenharia Elétrica e aos voluntários e voluntárias que os formam. Nesses grupos, aprendi a pensar a comunidade em que estou inserida, aprendi que se pode optar por fazer e não fazer e que o fazer de forma eficiente é o caminho.

Por fim, agradeço ao Laboratório de Processamento de Sinais e Informação e aos que o compõe. Agradeço, por fim, ao meu orientador Edmar Candeia pela oportunidade de trabalhar com o tema desse trabalho, pelo bom humor, pela orientação e por ser um exemplo como professor.

Abraços apertados aos que permitiram que essa jornada fosse melhor apreciada.

*Por um mundo onde
"Hurry up and wait"
seja incomum.*

RESUMO

Nos serviços de atendimento ao cliente como caixas eletrônicos, caixas de supermercados, aeroportos, acabam se formando filas de pessoas. É de interesse tanto dos clientes quanto dos prestadores de serviço saber qual o *status* da fila para auxiliar nas tomadas de decisões. Uma forma de obter informações sobre uma fila é usando técnicas de processamento de imagem e visão computacional para contar pessoas na fila, obter dados como os tempos de espera, tempos entre chegadas, entre outros. O objetivo do presente estágio foi o desenvolvimento de um sistema que possa oferecer informações sobre filas – detectar e contar pessoas. Para alcançar tal meta, foram realizados estudos sobre técnicas de detecção, bibliotecas da área e a implementação do sistema. O sistema recebe como entrada uma sequência de imagens (vídeo ou câmeras em tempo real) e extraí as informações necessárias para o gerenciamento de uma fila. O sistema desenvolvido é constituído por duas partes principais: uma responsável pela detecção de pessoas e outra pela contagem.

Palavras-chave: monitoramento de pessoas, sistema de visão computacional, monitoramento de filas.

ABSTRACT

In customer service such as supermarket's checkout, airports, hospital end up forming queues of people. It is in the interest of both customers and service providers to know which queue's status to assist in decision making. One way to get information about a queue is by using image processing and computer vision techniques to count people in the queue, get data such as waits, time between arrivals, and so on. The purpose of this stage was to develop a system that can provide information about queues - to detect and count people. To achieve this goal, studies on detection techniques, area libraries and the implementation of the system were carried out. The system receives as input a sequence of images (video or real-time cameras) and extracts the information needed to manage a queue. The developed system consists of two main parts: one responsible for detecting people and another for counting.

Keywords: monitoring of people, computer vision system, monitoring of queues.

LISTA DE ILUSTRAÇÕES

Figura 1 – Cenário do vídeo utilizado para prototipagem do sistema de monitoramento de filas de pessoas. Imagem extraída do vídeo de protipagem. . .	15
Figura 2 – Representação de uma imagem digital tirada da implementações usando a biblioteca OpenCV.	16
Figura 3 – Esquema do processo de subtração de fundo em que é gerado uma imagem binária contendo <i>pixels</i> pertencentes a objetos em movimento na cena.	18
Figura 4 – Sistema de Fila Genérico	22
Figura 5 – Implementação do Sistema de Monitoramento de Filas usando Catracas	25
Figura 6 – Imagem capturada do vídeo analisado. Na imagem, percebe-se o algoritmo realizando a detecção de pessoas no cenário.	26
Figura 7 – Imagem capturada do vídeo analisado. Na imagem, percebe-se o algoritmo realizando a detecção de pessoas saindo do sistema.	27
Figura 8 – Figura mostrando o arquivo gerado com os dados das entradas de pessoas no sistema de fila.	27
Figura 9 – Figura mostrando o arquivo gerado com os dados das saída de pessoas no sistema de fila.	28

SUMÁRIO

1	INTRODUÇÃO	11
1.1	Objetivos do estágio	12
1.2	O Laboratório	12
2	METODOLOGIA	13
2.1	Plano de Estágio	13
2.2	Ferramentas Utilizadas	13
2.2.1	OpenCV	14
2.2.2	Python	14
2.2.3	Base de Vídeo	14
3	ESTADO DA ARTE	16
3.1	Representação de uma Imagem Digital	16
3.2	Pré-processamento	17
3.2.1	Subtração de fundo	17
3.3	Deteccção de Pessoas	19
3.4	Rastreamento	21
4	MONITORAMENTO DE FILAS	22
4.1	O que monitorar?	22
4.1.1	Parâmetro A	23
4.1.2	Parâmetro B	23
4.1.3	Os outros parâmetros	23
4.2	O protótipo proposto	24
5	RESULTADOS E DISCUSSÕES	26
5.1	Vídeo Analisado	26
5.2	Sistema Desenvolvido	28
5.3	Implementação usando Rastreamento de Pessoas	29
6	CONCLUSÃO	30
	REFERÊNCIAS	31

1 INTRODUÇÃO

Filas de pessoas estão presentes em locais onde algum serviço é oferecido: agências bancárias, supermercados, aeroportos, hospitais. Sendo inevitáveis, as filas precisam ser gerenciadas afim de obter melhores resultados na utilização de sistemas. Em termos práticos, isso significa - havendo formação de filas - que os clientes devem passar pouco tempo nelas e os atendentes ou servidores não devem passar longos períodos ociosos. Uma forma de realizar o gerenciamento de filas, ou ainda informar os possíveis clientes sobre o *status* da fila, afim deles próprios decidirem o momento ideal para obter o serviço é monitorar filas. Sistemas de visão computacional permitem monitorar pessoas usando câmeras. O objetivo do estágio supervisionado relatado neste trabalho foi desenvolver um protótipo de um sistema capaz de monitorar filas de pessoas usando visão computacional e processamento digital de imagens.

O monitoramento de filas de pessoas envolve problemas menores: detectar movimentações no ambiente, verificar se as movimentações são de fato pessoas e realizar o rastreamento das pessoas detectadas. A detecção de movimentações no ambiente é útil para diminuir o espaço de buscas além de evitar que partes do ambiente (partes estáticas) sejam confundidas com pessoas. Este primeiro problema pode ser resolvido subtraindo o fundo estático das imagens processadas através, por exemplo, de uma imagem como modelo de fundo do ambiente. A verificação se os objetos em movimento em vídeos são pessoas é necessária já que podem haver, por exemplo, carrinhos de supermercados, mochilas ou malas em movimento no ambiente monitorado. Em visão computacional, há diversas técnicas para a detecção de pessoas. Neste trabalho, é necessário usar técnicas que detectem pessoas em posições mais ou menos verticais ou que detecte cabeças. Isso é porque a posição da câmera proposta para o sistema de monitoramento de filas é semelhante a posição das câmeras de segurança pois em diversos ambientes tais câmeras já estão instaladas. As soluções para detecção de pessoas em posição mais ou menos vertical é encontrada nos algoritmos de detecção de pedestres. Esses algoritmos usam extratores de características aliado a classificadores para obter informações da imagem e testar se essas informações indicam a presença de pessoas. O rastreamento das pessoas detectadas, diferente da detecção, permite identificar pessoas durante todo o vídeo. Sendo necessário, o rastreamento de pessoas, para obter informações de interesse da fila. Uma pesquisa sobre o estado da arte destes problemas foi feita e sintetizada no capítulo 3. Para determinar as informações de interesse de uma fila - que dados da fila devem se monitorados - recorreu-se a Teoria da Filas. Esta parte está descrita no capítulo 4.

No capítulo 4, além de uma revisão sobre Teoria da Filas, são propostas duas

implementações para o desenvolvimento do sistema de monitoramento. Dessas duas implementações, apenas uma foi usada para o desenvolvimento do protótipo. A solução usada é simples mas falha na robustez. É simples porque consegue contar pessoas entrando e saindo do sistema assim como obter o momento da entrada e da saída sem utilizar métodos de rastreamento de pessoas. No entanto, não é robusta porque dificilmente funcionaria em outros vídeo obtido em um ambiente diferente.

1.1 Objetivos do estágio

O objetivo geral do estágio supervisionado relatado neste trabalho é desenvolver um protótipo de um sistema capaz de monitorar filas de pessoas usando visão computacional e processamento digital de imagens. Os objetivos específicos são:

- Detectar Pessoas;
- Contar pessoas entrando na fila;
- Contar pessoas que saem do atendimento;
- Identificar atendentes e contar atendentes;
- Obter os momentos de entrada e saída da fila;
- Obter a duração dos atendimentos.

1.2 O Laboratório

A atividade relatada neste trabalho foi desenvolvida durante o estágio realizado no Laboratório de Processamento de Sinais e Informação - LAPSI, localizado no bloco CJ do Departamento de Engenharia Elétrica da Universidade Federal de Campina Grande, cuja coordenadora é a professora Luciana Veloso. O estágio foi realizado durante o período de 29 de Maio de 2017 à 21 de Agosto de 2017, com carga horária de 240 horas e atendendo os requisitos previstos na Resolução Número 01/2012 do Colegiado do Curso de Graduação de Engenharia Elétrica e em consonância com a Lei do Estágio (Lei Número 11.788/2008).

O laboratório conta com computadores com sistemas operacionais Linux Ubuntu e Windows, plataformas para desenvolvimentos de rádios definidos por software, DSPs da Texas Instruments e instrumentos de medição, como osciloscópios e analisadores de espectro. Além das pesquisas de iniciação científica e tecnológica, trabalhos de conclusão de curso e estágios, são também desenvolvidas no LAPSI. As atuais linhas de pesquisa do laboratório se concentram nas áreas de Rádio Definido por Software, Processamento de Sinais e Amostragem Compressiva.

2 METODOLOGIA

Neste capítulo serão mostrados os métodos e materiais usados para o desenvolvimento do estágio.

2.1 Plano de Estágio

O plano de atividades para o desenvolvimento de uma aplicação de monitoramento de filas foi o seguinte:

1. Estudo de técnicas de detecção e rastreamento de pessoas em vídeos.
2. Implementação de técnicas de detecção e rastreamento de pessoas em vídeos.
3. Extração de parâmetros de interesse para o monitoramento das filas.
4. Avaliação das técnicas implementadas quando utilizadas para monitorar filas.
5. Implementação de algoritmo que obtenha os parâmetros de uma fila.

O resultado do estudo é apresentado no capítulo 3 do presente trabalho. Os dois tópicos seguintes e último tópico do plano de estágio são mostrados no capítulo 4. O penúltimo tópico está no capítulo de Resultados e Discussões.

2.2 Ferramentas Utilizadas

O sistema foi desenvolvido em um *notebook* com processador Intel® Core i7 – 4700MQ CPU@2.4GHz e memória RAM de 8 GB. O sistema operacional instalado é o Ubuntu 16.04.3 LTS. O estágio começou sendo executado no Windows 10. No entanto, como foi percebido que as instalações de bibliotecas seriam facilitadas usando o sistema operacional Ubuntu, foi feita a migração entre os sistemas operacionais.

Para executar o estágio, foi escolhido a linguagem de programação de alto nível Python versão 3.5.2, e a biblioteca de visão computacional OpenCV versão 3.3.0. O editor de código usado foi Visual Studio Code. Todas essas ferramentas são disponibilizadas gratuitamente. Para o desenvolvimento do protótipo, foi produzido um vídeo afim de determinar um cenário simplificado para o monitoramento de filas. A câmera usada foi a câmera iSight de 8 *megapixels* com *pixels* de $1,5 \mu$ e gravação de vídeo HD de 1080p (30 quadros por segundo ou 60 qps).

2.2.1 OpenCV

O OpenCV - *Open Source Computer Vision* é uma biblioteca de visão computacional de código aberto. Esta biblioteca implementa uma variedade de ferramentas de interpretação de imagens, contendo desde operações simples como um filtro de suavização de ruído, até operações complexas, tais como a análise de movimentos, reconhecimento de padrões e reconstrução em 3D (MARENGONI M, 2009). Por ser uma biblioteca de visão computacional livre e possuir uma ampla comunidade de desenvolvedores, a biblioteca OpenCV foi a escolhida para o desenvolvimento do sistema de monitoramento e filas (MALLICK, b).

2.2.2 Python

A biblioteca OpenCV suporta as linguagens C, C++, Python, MatLab. Destas linguagens, escolheu-se usar Python pela conhecida utilidade da linguagem. A utilidade está na bibliotecas disponíveis em Python que são tão boas quanto as ferramentas presentes no MatLab quando se trata de computação científica. Algumas destas bibliotecas foram instaladas para o desenvolvimento do estágio. Foram elas: numpy, scipy, matplotlib, opencvPython, imultis. Essas bibliotecas não necessariamente foram usadas no protótipo final, mas durante a familiarização com OpenCV e Python, elas foram utilizadas.

2.2.3 Base de Vídeo

O vídeo usado para desenvolver o protótipo tem duração de 43 segundos, 1271 frames e são mostrados cerca de 30 frames por segundo. O frames do vídeo tem 848 pixels de comprimento e 480 pixel de altura. O cenário do vídeo foi pensado de modo a evitar oclusões ¹ e permitir implementações que levem em consideração a posição das pessoas no ambiente (a entrada e a saída de pessoas acontece em partes diferentes do cenário). O cenário foi pensado ainda de forma a simular câmeras de segurança pois essas são geralmente posicionadas na parte superior dos ambientes e direcionadas para baixo. Um frame do vídeo sem pessoas - o cenário - é mostrado na Figura 1.

Na Figura 1, tentamos simular um sistema de filas. A mesa branca com preto, no centro, simula o servidor, por exemplo, um caixa eletrônico. A parte à direita da mesa é onde ocorre a formação da fila, sendo a entrada do sistema. E a parte à esquerda é a saída do sistema, ou seja, por onde as pessoas saem após serem atendidas.

¹ Oclusão no contexto de rastreamento de objetos (no caso do trabalho, no rastreamento de pessoas) significa que o objeto a ser rastreado está oculto. Um exemplo é quando uma pessoa fica atrás da outra a ponto de não se ver a pessoa que ficou atrás. O problema, então, é como tratar a situação em que um objeto desaparece e reaparece novamente. Neste trabalho, esse problema foi evitado montando um cenário que dificulta a ocorrência de oclusão.

Figura 1 – Cenário do vídeo utilizado para prototipagem do sistema de monitoramento de filas de pessoas. Imagem extraída do vídeo de protipagem.



Fonte: Própria autora.

Inicialmente, um vídeo foi produzido para protipagem pois não se encontrou vídeos públicos mostrando filas de pessoas. No entanto, no decorrer do estágio, percebeu-se que era de fato interessante ter um vídeo com condições favoráveis para iniciar o desenvolvimento do sistema de monitoramento. Ao final do estágio, foi descoberto o Conjunto de Dados de Atividade Coletiva (*Collective Activity Dataset*), disponível em (SAVARESE,). Este conjunto de dados contém 5 atividades coletivas diferentes - cruzamento, caminhada, espera, conversação e filas - em 44 sequências de vídeos curtos. Algumas destas sequências foram gravadas por câmeras digitais portáteis com vários pontos de vista. As sequências, tanto as que mostram filas de fato quanto as outras, podem ser usadas para implementar melhorias no sistema de monitoramento em trabalhos futuros.

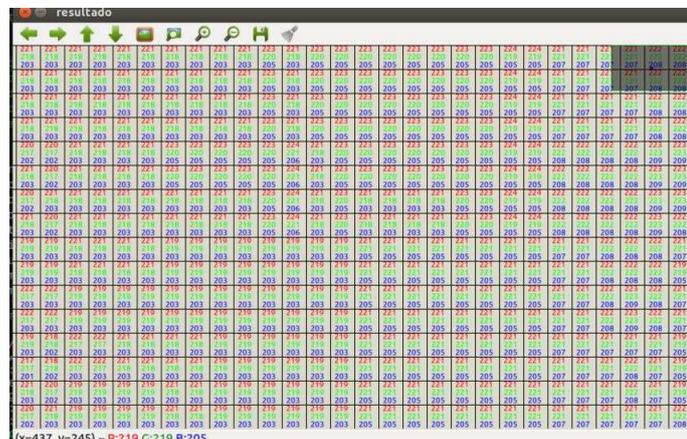
3 ESTADO DA ARTE

Nesta seção, será descrito técnicas de monitoramento de pessoas usando processamento de imagem e visão computacional.

3.1 Representação de uma Imagem Digital

Uma imagem digital é resultado do processo de discretização dos valores espaciais (x, y) e dos valores de brilho de uma imagem $f(x,y)$. A Figura 2 mostra como uma imagem digital pode ser representada: uma série de matrizes de duas dimensões, uma para cada faixa de cores RGB. Considerando $g(x, y)$ uma imagem digital representada por uma matriz de m linhas e n colunas, cada elemento da imagem pode ser determinado pelas suas coordenadas discretas x e y , onde $x < n$ e $y < m$. Além disso, geralmente adota-se a origem da imagem, o elemento $g(0, 0)$, como sendo o elemento do canto superior esquerdo da matriz. Cada elemento da matriz é chamado de pixel (PETROU M. E PETROU, 2010).

Figura 2 – Representação de uma imagem digital tirada da implementações usando a biblioteca OpenCV.



Fonte: Própria autora.

O conhecimento dessa informações é necessário quando se quer fazer manipulações espaciais diretas na imagem. Em uma determinada parte do desenvolvimento do sistema de monitoramento proposto neste trabalho, foi necessário desenhar linhas, para isso dois pontos da linhas precisam ser fornecidos. É interessante visualizar em torno dos quais valores se encontram as coordenadas (x,y) de uma parte da imagem que se quer desenhar.

Quando se trata de análise de vídeos, que é o objeto de estudo deste trabalho, é interessante destacar alguns pontos. Em geral, pode-se descrever um vídeo como uma sequência de imagens. No entanto, como vídeos muitas vezes são compactados de forma a alterar os detalhes na imagem, prefere-se chamar cada imagem do vídeo de quadro ou *frame*. Neste trabalho, devido a semelhança, as palavras imagens, quadros e *frames* são usadas como sinônimo.

3.2 Pré-processamento

Geralmente, a primeira fase em um sistema de visão computacional consiste em fazer um pré-processamento na imagem de forma a diminuir imperfeições e ressaltar características que serão usadas nas etapas seguintes. Exemplos de pré-processamento são: cortar a imagem, ajustar contraste, reduzir ruídos oriundos de má iluminação.

3.2.1 Subtração de fundo

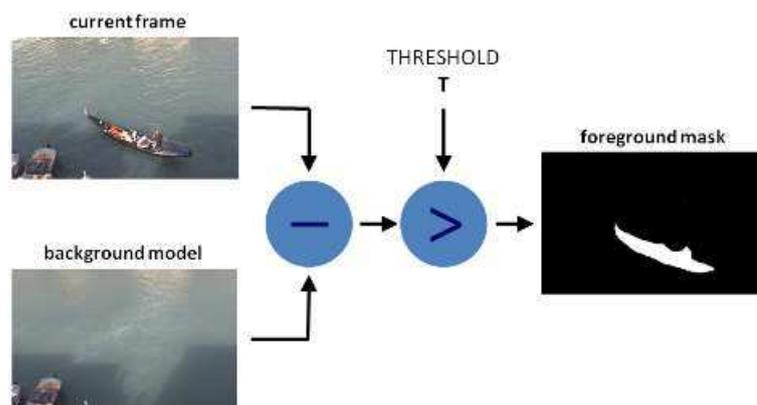
A subtração de fundo é uma etapa de pré-processamento importante em muitas aplicações baseadas em visão computacional, por exemplo detecção de carros, detecção de pessoas (ITSEEZ, 2016). Em um vídeo, que normalmente é capturado a partir de câmeras estática, ela é responsável pela detecção de objetos em movimento (MOESLUND; HILTON; KRÜGER, 2006). Quando se quer detectar objetos em movimento, geralmente, é segmentado o objeto (pessoa, veículo) e depois é feito o reconhecimento dele. Para isso, tecnicamente, é necessário extrair o plano móvel do fundo estático antes de realizar as etapas seguintes de processamento.

Se é possível ter a imagem de fundo sozinha, como a imagem de uma sala que se quer detectar pessoas ou a imagem de uma estrada sem veículos, basta subtrair a nova imagem obtida de imagem do fundo. No entanto, em diversos ambientes, o fundo varia com a iluminação dificultando a operação simples de apenas subtrair. Há também que na maioria dos casos, não se consegue ter a imagem a imagem do fundo sem os objetos em movimento, então é necessário extrair o plano de fundo de qualquer imagem existente. Vários algoritmos foram introduzidos para esse fim (ITSEEZ, 2016).

O processo de separação de fundo, normalmente, é feito por análise da variação de intensidade de cada pixel ao longo da sequência de imagens, sendo o fator indicativo da ocorrência de movimento a existência ou ausência de variação. A subtração de fundo baseada na diferença simples entre imagens não apresenta resultados interessantes mas é útil para visualização da técnica. Este método consiste em subtrair diretamente os valores das intensidades de cada pixel em duas imagens consecutivas. No caso do resultado dessa subtração ser maior que um limiar (*threshold*), então é considerado que esse pixel corresponde a um alvo em movimento, pois houve uma variação significativa do valor

de intensidade. Por outro lado, caso esteja abaixo ou igual ao mesmo limiar, significa que a variação da intensidade não foi expressiva para que o pixel seja considerado como pertencente a um alvo em movimento. A Figura 3 mostra o esquema deste processo (COSTA, 2010).

Figura 3 – Esquema do processo de subtração de fundo em que é gerado uma imagem binária contendo *pixels* pertencentes a objetos em movimento na cena.



Fonte: (ITSEEZ, 2016).

Outros algoritmos de subtração de fundo melhoram a robustez do algoritmo de diferença simples. Alguns desses são o método unimodal, *kernel estimation* e de misturas Gaussianas. O método unimodal tem como base o princípio que os valores das intensidades de cada pixel do fundo, numa sequência de imagens, seguem uma distribuição unimodal, ou seja uma distribuição que possui apenas uma moda; um exemplo é distribuição Gaussiana. Neste método, um pixel pertence ao fundo estático se a sua intensidade, ao longo da sequência de imagens, estiver compreendida numa gama de valores que é função da média e do desvio padrão das intensidades: [média – desvio padrão ; média + desvio-padrão]. Qualquer pixel cujos valores de intensidades não seguem a distribuição são considerados como pertencentes a alvos em movimento. O método *kernel estimation* é baseado na estimação da função de densidade de probabilidade para o valor da intensidade de um pixel. A informação da função de densidade de probabilidade é formada a partir do histograma das imagens mais recentes. Todos os pixels que seguem e reforçam essa mesma função de densidade de probabilidade, ou seja, todos os valores de intensidades de cada pixel que são similares aos que já estão previamente guardados, são designados como pertencentes ao fundo. Por outro lado, sempre que a intensidade do pixel é diferente do que se tem obtido (não segue a função densidade de probabilidade) significa que o pixel representa um objeto em movimento. Quanto aos métodos de misturas gaussianas, esses são os mais usados. Inclusive, dois dos três métodos de subtração fundo implementados no OpenCV são baseados em misturas Gaussianas (COSTA, 2010).

O método das misturas Gaussianas tem como base o método unimodal, mas, em

vez de considerar a existência de uma única distribuição Gaussiana, considera que podem existir várias podendo a quantidade k variar de aplicação para aplicação. A ideia base das misturas Gaussianas parte do princípio que apenas uma distribuição Gaussiana é incapaz de se adaptar a variações que não representam movimento, por exemplo, alteração de iluminação ou outras pequenas oscilações de objetos, como cortinados ou árvores. O fato de se considerar a existência de mais do que uma moda, dependendo do valor k utilizado, permite que cada pixel possa ser classificado como pertencente a uma das distribuições Gaussianas. O método vai atualizando os pesos dados a cada distribuição Gaussiana à medida que vai recebendo imagem do vídeo, e de modo similar ao método unimodal, cada pixel tende a seguir uma das distribuições. O facto do valor de intensidade de cada pixel poder seguir várias distribuições torna possível uma rápida adaptação a mudanças e, consequentemente, uma melhor subtração do fundo estático. Os algoritmos em OpenCV baseados em misturas Gaussianas são o "BackgroundSubtractorMOG" e "BackgroundSubtractorMOG2". O primeiro foi introduzido no artigo "Um modelo de mistura de fundo adaptativo melhorado para rastreamento em tempo real com detecção de sombra" por P. KadewTraKuPong e R. Bowden em 2001. Ele usa um método para modelar cada pixel de fundo por uma mistura de distribuições Gaussianas (com $K = 3$ a 5). Os pesos da mistura representam as proporções de tempo que essas cores permanecem na cena. As cores de fundo prováveis são as que ficam mais longas e mais estáticas. O segundo algoritmo é baseado em dois artigos de Z. Zivkovic, "Modelo adaptativo de mistura Gaussiana melhorado para a subtração de fundo" em 2004 e "Estimativa da densidade adaptativa eficiente por pixel de imagem para a tarefa de subtração de fundo" em 2006 (ITSEEZ, 2016). Uma característica importante desse algoritmo é que ele seleciona o número adequado de distribuição gaussiana para cada pixel; diferente do caso anterior em que é tomado distribuições de k gaussianas ao longo dos algoritmos.

A subtração de fundo torna possível reduzir a quantidade de informação a processar e ajuda a evitar falhas no sistema pois elimina partes estáticas que parecem pessoas, por exemplo, e não são. Percebe-se que os problemas enfrentados pela subtração de fundo são a variação de iluminação do ambiente, a sensibilidade ao ruído, a oclusão, as sombras dos objetos em movimentos. Há várias pesquisas que abordam tais problemas trazendo algoritmos cada vez mais robustos.

3.3 Detecção de Pessoas

Na subtração de fundo, consegue-se obter silhuetas bem definidas em torno dos objetos. No entanto, esta técnica não é capaz de localizar objetos individualmente ou de detectar objetos com pouco movimento ou, ainda, reconhecer de fato o objeto. Para essas deficiências, existem as técnicas de detecção de objetos que englobam detecção de pessoas.

A detecção de pessoas em imagens é uma tarefa desafiadora devido à aparência variável e à ampla gama de pose que os seres humanos podem adotar. Quando se trata de detectar pessoas em poses verticais, surge o problema conhecido como detecção de pedestre. A detecção de pedestres consiste em procurar, em uma imagem, indivíduos que estão em pé, podendo estar de frente, de costas ou até mesmo de lado para a câmera (DALAL; TRIGGS, 2005). Este é o caso de pessoas esperando atendimento em uma fila.

Os algoritmos de detecção de pessoas mais populares atualmente se baseiam em aprendizado de máquina mas especificamente em *Deep Learning*. Estes, em geral, funcionam extraíndo informações da imagem que auxiliam na identificação de pessoas e classificando se a região da imagem é uma pessoa ou não. Essas duas partes principais da detecção são realizadas por descritores e classificadores respectivamente. Os descritores usam algum critério para extrair características. O classificador utiliza as características extraídas em conjunto com outras que foram aprendidas a priori através de um treinamento para determinar a existência de pessoas.

Os descritores mais comuns são os baseados em características Haar e o Histograma de Gradientes Orientados (DALAL; TRIGGS, 2005) (HOG - *Histogram of Oriented Gradients*). As características Haar-like são atributos extraídos de imagens, baseados nas *wavelets* de Haar. Este tipo de característica para imagens foi proposto em (PAPAGEORGIOU; OREN; POGGIO, 1998) e utilizado para o reconhecimento de faces em (VIOLA; JONES, 2004). Uma imagem pode conter milhares de características Haar e cada uma dessas características pode ser utilizada por um classificador. Para detecção de pedestres, os descritores HOG apresentam melhores resultados que os baseados em características Haar. Os descritores HOG atuam extraíndo informações de orientação e magnitude dos vetores gradiente. Essas informações são úteis porque a magnitude dos gradientes é grande em torno de bordas e cantos pois são regiões de mudanças abruptas de intensidade. Como as bordas e cantos embalam muito mais informações sobre a forma do objeto do que as regiões planas, o descritor HOG apresenta bons resultados para detecção das bordas de pedestre (DALAL; TRIGGS, 2005). As informações extraídas são utilizadas para construir um histograma o qual será usado por um classificador.

Os classificados usados com os descritores descritos acima são os classificadores em Cascata e a Máquina de Vetor de Suporte (SVM - *Support Vector Machine*). A técnica de classificadores em cascata consiste em unir diversos classificadores para atingir um objetivo. Cada classificador tem uma função distinta no processo de classificação, sendo responsável por uma porção do espaço de características. Cada classificador da cascata pode ser considerado um classificador fraco, ou seja, não tem uma alta taxa de classificação, chegando próximos de uma classificação aleatória. Entretanto a união de vários classificadores fracos produz um classificador forte, que é o objetivo da cascata de classificadores. O SVM ou Máquina de Vetor de Suporte é um classificador linear, independentemente da distribuição

dos dados ser linear ou não. É um classificador de aprendizado supervisionado utilizado em diversas aplicações de reconhecimento de padrões, como: imagens, fala e previsões de séries temporais. O SVM é o classificador usado com o descritor HOG (OLIVEIRA, 2016).

3.4 Rastreamento

Rastrear é localizar um objeto em *frames* sucessivos de um vídeo. Em um sistema que visa monitorar pessoas, o rastreamento, diferente da detecção, permite detectar pessoas sob oclusão em algum *frame*. Os algoritmos de rastreamento podem fazer isso usando informações de *frames* anteriores, como por exemplo o famoso filtro de Kalman. Além de permitir resolver problemas de oclusão, o rastreamento é útil em diversas aplicações. No monitoramento de filas, o objetivo deste estágio, o rastreamento é útil identificando cada cliente da fila de modo a saber a localização do cliente identificado em todos os *frames* (MALLICK, a).

Os algoritmos que implementam o rastreamento, em geral, tem princípios diferentes. O filtro de Kalman prever a localização de um objeto em movimento com base em informações anteriores. O Meanshift e Camshift, outro algoritmo que permite o rastreamento, consegue localizar o máximo de uma função de densidade em cada *frame*; nesse caso o pontos máximos da função de densidade serão os pontos a serem rastreados.

O OpenCV 3 vem com a API (*application program interface*) de rastreamento que contém implementações desses e de outros algoritmos de rastreamento de um único objeto. Os rastreadores de um único objeto, diferente dos rastreadores de múltiplos objetos, no primeiro *frame*, usam um retângulo para indicar a localização do objeto a se rastrear. O objeto é então rastreado em imagens subsequentes usando o algoritmo de rastreamento. Na maioria das aplicações da vida real, esses rastreadores são usados em conjunto com um detector de objetos. Quanto aos rastreadores de múltiplos objetos, esses precisam de um detector de objeto rápido pois vários objetos são detectados em cada imagem e, em seguida, é executado um algoritmo de busca de trilha que identifique qual retângulo em um quadro corresponde a um retângulo no próximo quadro (MALLICK, a).

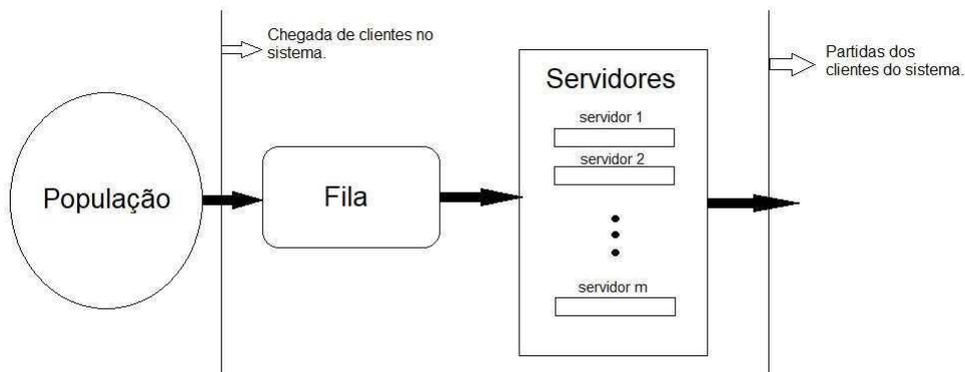
4 MONITORAMENTO DE FILAS

O monitoramento de filas visa acompanhar as pessoas em uma fila de forma a extrair informações de interesse da fila. Para saber que informações são essas, recorreu-se a teoria das filas. A teoria das filas é usada para dimensionar, avaliar e estimar estados ¹ futuros dos sistemas de fila.

4.1 O que monitorar?

Um sistema de fila é representado na Figura 4. Basicamente, observa-se a existência de uma população que fornece usuários ao sistema. Os usuários em espera pelo serviço constituem a fila. Os servidores prestam atendimento aos clientes e, após o atendimento, os clientes saem do sistema.

Figura 4 – Sistema de Fila Genérico



Fonte Própria

Em teoria das filas, um sistema de fila é identificado pelos seguinte parâmetros:

- Distribuição de probabilidade da chegada de clientes ou do intervalo entre chegadas;
- Distribuição de probabilidade do tempo de serviço;
- Número de atendentes;
- Capacidade de armazenamento do sistema;
- Tamanho da População de Clientes;

¹ Estados se refere ao número de pessoas na fila, tempo de atendimento, tempo de espera.

- Disciplina do atendimento.

Esses parâmetros, respectivamente, formam a notação de Kendall, $A/B/m/K/N - S$. O dispositivo que tiver finalidade de extrair informações de uma fila afim de usar a teoria das filas deve, então, preencher a notação $A/B/m/K/N - S$. Os parâmetros A e B tratam das distribuições de probabilidade. O monitoramento automático de filas permite que uma quantidade necessária de dados seja obtida para determinar as distribuições de probabilidade. Essas distribuições são obtidas fazendo um análise estatísticas dos dados. Dessa análise, pode-se encontrar distribuições estatísticas que mais se assemelham ao dados reais.

4.1.1 Parâmetro A

O parâmetro A diz respeito ao processo de chegada de clientes em uma fila. Para caracterizar tal processo é necessário conhecer a distribuição da probabilidade da chegada de clientes ou a distribuição do intervalo entre chegadas. Para se chegar a tais distribuições, os tempos entre chegadas ou quantas pessoas chegaram em um determinado tempo devem ser obtidos, devem ser monitorados. No monitoramento das chegadas, é preciso contar quantas pessoas entraram na fila dentro de um intervalo de tempo determinado. No monitoramento do tempo entre chegadas, o dispositivo precisa identificar o tempo que uma pessoa entrou, em seguida, pegar o tempo que a próxima pessoa entrou no sistema e subtrair do tempo anterior.

4.1.2 Parâmetro B

O parâmetro B diz respeito ao processo de atendimento de clientes em um sistema de filas. Para caracterizar tal processo é necessário conhecer a distribuição da probabilidade dos tempos de duração do atendimento ou quantos atendimentos são feitos em um período de tempo. Para se chegar a tais distribuições, os tempos de duração do atendimento de cada cliente ou as quantidades de clientes atendidos em um determinado tempo devem ser obtidos, devem ser monitorados.

4.1.3 Os outros parâmetros

É importante que um dispositivo que tenha interesse em obter informações de uma fila, consiga saber quantos atendentes m estão no sistema pois há dependência de tal valor na estimação de variáveis da fila por teoria das filas. A capacidade de armazenamento do sistema é um parâmetro que geralmente não varia no tempo, podendo ser considerado uma constante, eliminado a preocupação sobre o monitoramento desse. O mesmo ocorre para o parâmetro que especifica o tamanho da população. Quanto a disciplina da fila, ela insere uma dificuldade maior na análise pois é preciso identificar a ordem com que os clientes

são atendidos. Para simplificar, considera-se, neste trabalho, que os primeiros clientes a entrar no sistema serão os primeiros a serem atendidos. Essa disciplina é conhecida com FIFO (*First In First Out*). Por tanto, a disciplina da fila é outro parâmetro o qual o monitoramento descrito neste trabalho não se preocupou.

4.2 O protótipo proposto

Durante o estágio, foi pensado em duas formas de implementar o protótipo. A primeira forma de implementação é inspirada no trabalho (MIAZAKI A. A. P. CAMPOS, 2010). Neste trabalho, um sistema de monitoramento de pessoas, o Aleph, também foi desenvolvido. No entanto, no sistema de monitoramento deste trabalho, o objetivo final do sistema é gerar mapas de calor ou seja monitorar a rota das pessoas no vídeo de forma a conseguir as áreas que as pessoas passam mais tempos e dos caminhos frequentemente percorridos. Para conseguir isso, o Aleph conta com uma parte de Detecção de Pessoas e uma parte de Rastreamento. Essas partes alimentam uma a outra de forma a conseguir saber a posição da pessoa no ambiente monitorado no decorrer de um vídeo. Tendo tal informação é possível traçar as rotas e obter um mapa de calor.

Em um sistema de monitoramento de filas, porém, não é necessário saber a posição da pessoa no ambiente em todos os momentos. Basicamente, é necessário saber quando a pessoa entrou no sistema, quando ela saiu do sistema e quanto tempo ela passou no atendimento. Tomando como base isso, a segunda implementação foi pensada de modo a não precisar do rastreamento de pessoas. Usando a ideia de catracas em que as pessoas são contadas ao passar por elas, a segunda implementação usa retas desenhadas na imagem que representariam as catracas. Quando uma pessoa é detectada no *frame*, um retângulo limitando a pessoa é determinado e um ponto no centro do retângulo é encontrado. A ideia é que o ponto central irá passar pela reta quando a pessoa passar pela "catraca-pela reta. O sistema, então, identificará que uma pessoa entrou ou saiu.

Dentre as implementações propostas, apenas a segunda chegou a ser implementada de fato durante o estágio. A implementação percorre os *frames* realizando, em cada *frame*, as seguinte tarefas:

- Detecção de pessoas gerando um retângulo que circunda a pessoa detectada e o ponto central deste retângulo;
- Desenha a catraca de entrada e de saída;
- Detecta se alguma pessoa passou por alguma das catracas;
- Armazena os horários de chegada e saída caso tenha detectado alguma pessoa na catraca;

- Mostra o *frame* resultante destas operações;
- Escreve no arquivo CSV os horários de chegadas e saídas do vídeo analisado.

Foi usado apenas um vídeo para a prototipagem. Neste vídeo, as "catracas" foram desenhadas tomando como base os valores assumidos pelos pontos centrais das pessoas. Analisando os pontos centrais, percebe-se em quais regiões do cenário as retas devem ser desenhadas. Então quatro pontos - dois para a catraca de saída, dois para catraca de entrada - foram determinados tomando como base a localização da entrada e da saída de pessoas. A Figura 5 mostra a catraca de entrada e de saída as quais são as retas vermelha e azul respectivamente. O programa tem a funcionalidade de receber tais pontos e determinar a equação da reta destes pontos. A equação da reta é usada para testar se os pontos centrais interceptaram a reta em algum momento. Para a detecção de pessoas, foi usado o descritor HOG em conjunto com o classificador SVM.

Figura 5 – Implementação do Sistema de Monitoramento de Filas usando Catracas



Fonte Própria

5 RESULTADOS E DISCUSSÕES

Neste capítulo, são mostrados e discutidos os resultados obtidos.

5.1 Vídeo Analisado

Algumas imagens do vídeo analisado foram capturadas afim de mostrar o funcionamento do sistema. A primeira imagem é da Figura 6. Nesta imagem, é mostrado a detecção de pessoas esperando na fila assim como as catracas.

Figura 6 – Imagem capturada do vídeo analisado. Na imagem, percebe-se o algoritmo realizando a detecção de pessoas no cenário.



Fonte Própria

A Figura 7 mostra o momento em que ocorre a detecção da pessoa saindo do sistema de filas. Ou seja, detecta saída da pessoa, possibilitando obter o momento em que ocorre a saída.

Durante o vídeo usado para prototipagem, apenas as duas pessoas aparecem como mostrado na Figura 6. Essa informação é interessante para se conferir os resultados gerados no arquivo CSV. A implementação no vídeo de prototipagem gerou os arquivos das Figuras 8 e 9.

Figura 7 – Imagem capturada do vídeo analisado. Na imagem, percebe-se o algoritmo realizando a detecção de pessoas saindo do sistema.



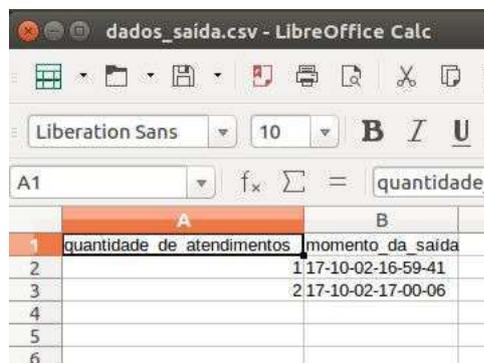
Fonte Própria

Figura 8 – Figura mostrando o arquivo gerado com os dados das entradas de pessoas no sistema de fila.

	A	B
1	quantidade de chegadas	momento da chegada
2		1 17-10-02-16-58-28
3		2 17-10-02-16-59-09
4		
5		
6		

Fonte Própria

Figura 9 – Figura mostrando o arquivo gerado com os dados das saída de pessoas no sistema de fila.



	A	B
1	quantidade de atendimentos	momento da saída
2	1	17-10-02-16-59-41
3	2	17-10-02-17-00-06
4		
5		
6		

Fonte Própria

5.2 Sistema Desenvolvido

Nesta seção, serão discutidos os resultados obtidos mostrando os pontos negativos e positivos.

Os pontos positivos do sistema desenvolvido são:

- Não utiliza rastreamento de pessoas.
- Implementação simples.
- Consegue contar as entradas e saídas de pessoas.

Os pontos negativos são:

- Implementação demorada pois é necessário determinar retas que todas as pessoas monitoradas interceptem.
- A reta, a catraca, não é confiável. É possível, a depender do formato da pessoa, que mesmo que haja passagem de pessoas pela reta, esta não seja detectada.
- Não consegue detectar os atendentes.
- Não consegue obter a duração do atendimento.
- Devido ao tempo de processamento, obtém de forma atrasada o momento de entrada e saída de pessoas.

A implementação com catracas trouxe resultados desejados para o vídeo analisado. No entanto, se outro vídeo fosse gravado e analisado, era necessário manter exatamente a

mesma posição da câmera e do cenário para obter os mesmos resultados. Essa falta de adaptação dos sistemas a outros cenários ou posições da câmera acontece por conta da determinação das catracas.

5.3 Implementação usando Rastreamento de Pessoas

Implementou-se, usando a API de rastreamento do OpenCV, algoritmos que conseguiam detectar pessoa no vídeo. No entanto, não conseguiu-se rastrear mais de um pessoa durante o estágio. Por isso, a implementação não foi levada adiante.

A implementação utilizando rastreamento permitiria resolver os pontos negativos apresentados na implementação da catraca.

6 CONCLUSÃO

No estágio, foi desenvolvido um protótipo de um sistema capaz de monitorar filas. Para isso técnicas de visão computacional para monitorar pessoas foram estudadas: subtração de fundo, detecção de pessoas e rastreamento. Além do estudo feito, foi necessário também se familiarizar com a biblioteca de visão computacional OpenCV e com a linguagem de programação Python.

Os materiais encontrados para o estudo de técnicas de visão computacional são bastantes úteis; mostram que as pesquisas são recentes e que continuam sendo desenvolvidas. Isso possibilita que cada vez mais aplicações eficientes sejam implementadas. As dificuldades encontradas se concentram no uso da ferramentas pois foi necessário um tempo de familiarização com biblioteca OpenCV e com a linguagem Python. Outra dificuldade é preparação do ambiente para desenvolvimento que requer que versões certas do OpenCV e Python sejam instaladas.

Enfrentadas as dificuldades, o protótipo implementado conseguiu detectar pessoas, detectar a entrada de pessoas no sistema, detectar a saída de pessoas no sistema e armazenar o dia, hora, minuto e segundo atual das entradas e saídas. Por outro lado, não se conseguiu detectar e contar os atendentes nem obter a duração dos atendimentos.

Como sugestão de trabalhos futuros, é interessante desenvolver os objetivos não alcançados. Uma sugestão para tal é implementação usando rastreamento de pessoas descrita no trabalho. Além de permitir que todos os objetivos sejam alcançados, ela seria mais robustas que o protótipo desenvolvido. Outro ponto a ser melhorado, é obtenção dos momentos de chegada e saída de forma a não sofrer influência do tempo de processamento do vídeo.

REFERÊNCIAS

- COSTA, R. A. T. da. *Sistema de visão para detecção de pessoas em movimento*. Dissertação — Faculdade de Engenharia da Universidade do Porto, Julho 2010. Citado na página 18.
- DALAL, N.; TRIGGS, B. Histograms of oriented gradients for human detection. In: *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*. Washington, DC, USA: IEEE Computer Society, 2005. (CVPR '05), p. 886–893. ISBN 0-7695-2372-2. Disponível em: <<http://dx.doi.org/10.1109/CVPR.2005.177>>. Citado na página 20.
- ITSEEZ. *The OpenCV Reference Manual*. 3.3.0. ed. [S.l.], 2016. Citado 3 vezes nas páginas 17, 18 e 19.
- MALLICK, S. Object tracking using opencv. Acessado em 29 de Julho de 2017. Disponível em: <<https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/>>. Citado na página 21.
- MALLICK, S. Opencv (c++ vs python) vs matlab for computer vision. Acessado em 25 de junho de 2017. Disponível em: <<https://www.learnopencv.com/opencv-c-vs-python-vs-matlab-for-computer-vision/>>. Citado na página 14.
- MARENGONI M, S. S. *Tutorial: Introdução à visão computacional usando opencv*. [S.l.]: Revista de Informática Teórica e Aplicada, 2009. v. 16. Citado na página 14.
- MIAZAKI A. A. P. CAMPOS, F. M. S. L. R. B. G. R. I. Aleph: Sistema de monitoramento visual de pessoas. São Paulo. 2010. Citado na página 24.
- MOESLUND, T. B.; HILTON, A.; KRÜGER, V. A survey of advances in vision-based human motion capture and analysis. *Comput. Vis. Image Underst.*, Elsevier Science Inc., New York, NY, USA, v. 104, n. 2, p. 90–126, nov. 2006. ISSN 1077-3142. Disponível em: <<http://dx.doi.org/10.1016/j.cviu.2006.08.002>>. Citado na página 17.
- OLIVEIRA, D. C. D. *UMA ABORDAGEM PARA DETECÇÃO DE PESSOAS EM IMAGENS DE VEÍCULOS AÉREOS NÃO-TRIPULADOS*. Dissertação — UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ, Curitiba, 2016. Citado na página 21.
- PAPAGEORGIOU, C.; OREN, M.; POGGIO, T. A general framework for object detection. *Computer Vision, 1998. Sixth International Conference on*, p. 555–562, 1998. Citado na página 20.
- PETROU M. E PETROU, C. *Image Processing: The Fundamentals*. 2nd. ed. [S.l.]: John Wiley & Sons, 2010. ISBN 978-0-470-74586-1. Citado na página 16.
- SAVARESE, C. S. Collective activity dataset. Acessado em 29 de setembro de 2017. Disponível em: <<http://vhosts.eecs.umich.edu/vision//activity-dataset.html>>. Citado na página 15.

VIOLA, P.; JONES, M. J. Robust real-time face detection. *Int. J. Comput. Vision*, Kluwer Academic Publishers, Hingham, MA, USA, v. 57, n. 2, p. 137–154, maio 2004. ISSN 0920-5691. Disponível em: <<https://doi.org/10.1023/B:VISI.0000013087.49260.fb>>. Citado na página 20.