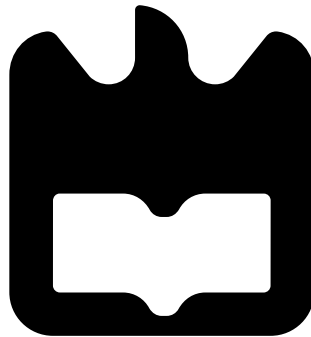




**Ricardo Jorge
Sousa Bandeira**

Passadeiras de peões de nova geração





**Ricardo Jorge
Sousa Bandeira**

Passadeiras de peões de nova geração

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Eletrónica e Telecomunicações, realizada sob a orientação científica do Professor Doutor José Alberto Fonseca, Professor Associado do Departamento de Eletrónica, Telecomunicações e Informática da Universidade de Aveiro

o júri / the jury

presidente / president

Professor Doutor Alexandre Manuel Moutela Nunes da Mota
Professor associado da Universidade de Aveiro

vogais / examiners committee

Professor Doutor José Alberto Gouveia Fonseca
Professor associado da Universidade de Aveiro (orientador)

Professor Doutor Carlos Jorge Enes Capitão de Abreu
Professor adjunto da Escola Superior de Tecnologia e Gestão do Instituto
Politécnico de Viana do Castelo

**agradecimentos /
acknowledgements**

Aproveito a oportunidade para agradecer a um conjunto de pessoas incríveis que contribuíram direta ou indiretamente para este projeto.

Ao meu orientador, o professor José Fonseca, por ter acreditado no meu potencial e me ter dado a oportunidade de trabalhar com ele. Em muito lhe devo o meu crescimento enquanto profissional e só tenho a agradecer pela disponibilidade e compromisso.

A todos os colaboradores da Microio, com destaque para o departamento de Engenharia, por me acolherem na equipa e estarem sempre disponíveis para ajudar.

Aos meus pais, irmão e afilhada, por todo o apoio e confiança que depositaram em mim, não só agora, mas ao longo de todos estes anos.

Aos meus amigos mais chegados, por semana após semana me ajudarem a recarregar baterias, ouvirem os meus desabafos e acompanharem o meu progresso.

Por fim, aos meus colegas de curso, com quem compartilhei os desafios destes últimos 5 anos.

Um bem-haja a todos!

palavras-chave

comunicações, passadeira de peões, radar, redes neuronais, sensores, segurança rodoviária, sistemas ciber-físicos

resumo

Nos últimos anos temos assistido a uma grande aposta na segurança rodoviária por parte de muitos países ao redor do mundo, incluindo Portugal. Ainda assim, segundo a Autoridade Nacional de Segurança Rodoviária, são atropeladas cerca de catorze pessoas por dia [1].

Tendo em vista a redução destes valores, e como resposta aos novos perigos que advém dos peões se deslocarem a olhar para os telemóveis, ou dos veículos elétricos se deslocarem silenciosamente, propomos um novo tipo de passadeiras designado de Sinalização Transversal Ativa de Passadeiras (STAP).

STAP é uma solução para o alerta de condutores e peões, mediante a identificação de situações de perigo nas proximidades das passadeiras, por meio da deteção em tempo real de pessoas e veículos. Utiliza a projeção de iluminação sobre a estrada para delimitar o acesso e captar a atenção dos intervenientes com perfis luminosos específicos, de acordo com a presença do peão, a velocidade das viaturas e a visibilidade.

No decorrer projeto foi dado ênfase à deteção de pessoas através de uma rede neuronal, à aquisição de velocidades com recurso a um radar e ao controlo das luminárias de acordo com a iluminação do local. Neste momento a sinalização luminosa é feita com tiras de LED RGB endereçáveis aplicadas nos postes, mas considerando a projeção de luz na estrada num futuro próximo.

Foram realizados diversos testes para comprovar a eficácia e as limitações do sistema, tais como contagem e classificação de veículos, exatidão na deteção de peões, otimização dos tempos de autorização e proibição de passagem e controlo de luminosidade, tendo em vista a criação de um protótipo em tamanho real.

keywords

communications, pedestrian crossing, radar, neural networks, sensors, road safety, cyber-physical systems

abstract

In the last years we have witnessed a significant investment on road safety from many countries around the world, including Portugal. Still, according to the National Road Safety Authority, about fourteen people are run over in Portugal every day [1].

With the goal to reduce these values and in response to the new dangers that come from pedestrians looking at mobile phones while walking, or from silently moving electric vehicles, we propose a novel type of crosswalk called Active Transversal Signaling for Pedestrian Crosswalks - STAP (Sinalização Transversal Ativa de Passadeiras).

STAP is a solution for alerting drivers and pedestrians by identifying dangerous situations near the crosswalks through real-time detection of vehicles and people. It uses light stripes to delimit access and capture the attention with specific light profiles, according to pedestrian presence, vehicle speed and visibility.

In the course of the project, emphasis was placed on the detection of people through a neural network, in the acquisition of vehicles speeds using a radar and the control of luminaires according to the lighting conditions of the place. At this moment light signaling is performed using addressable RGB LED strips mounted in the posts, while in the future we consider projection of light on the road.

Several tests have been performed to prove the effectiveness and limitations of the system, such as vehicle counting and classification, pedestrian detection accuracy, optimization of authorization and prohibition times and lighting control, aiming to create a prototype in real size.

Conteúdo

Conteúdo	i
Lista de Figuras	iii
Lista de Tabelas	vii
Acrónimos	ix
1 Enquadramento	1
1.1 Motivações	1
1.2 Objetivos	3
1.3 Estrutura do documento	3
1.4 Apoios	4
2 Estado da arte	5
2.1 Soluções comerciais	5
2.1.1 Soluções de iluminação	6
2.1.2 Soluções com deteção de peões	8
2.1.2.1 Deteção por infravermelhos	8
2.1.2.2 Deteção por sensores de pressão	9
2.1.2.3 Deteção por câmara	10
2.1.3 Soluções com deteção de peões e veículos	11
2.1.4 Análise comparativa sumária	13
2.2 Soluções académicas	14
2.2.1 Evaluation of human behaviour at pedestrian crossings	14
2.2.2 Passive pedestrian detection at unsignalized crossings	14
2.2.3 Innovative solutions for improving safety at pedestrian crossings	15
2.2.4 Informational report on lighting desing for midblock crosswalks	15
2.2.5 The basis of luminance and illuminance recommendations	16
2.3 Técnicas para deteção de peões	16
2.3.1 Piezoelétrico	17
2.3.2 Infravermelhos	17
2.3.3 Ultrassons	18
2.3.4 Micro-ondas / Doppler radar	18
2.3.5 Câmaras	19
2.3.6 Resumo das características	19

3	Especificações do sistema	21
3.1	Descrição inicial	21
3.2	Requisitos do projeto	23
3.3	Funcionamento	23
3.3.1	Zonas de deteção	24
3.3.2	Modos de operação	29
3.4	Vantagens	33
4	Implementação	35
4.1	Arquitetura do protótipo	35
4.1.1	Estrutura	35
4.1.2	Componentes	38
4.1.3	Fluxograma funcional	43
4.1.4	Eletrónica adicional	47
4.1.5	Instalação	51
4.2	Gestão de tráfego	57
4.2.1	Configuração do radar	57
4.2.2	Protocolos de comunicação	61
5	Rede neuronal	65
5.1	Conceitos-chave	65
5.1.1	Rede neural convolucional	65
5.1.2	Precisão média	67
5.1.3	Interseção sobre união	68
5.2	You Only Look Once	68
5.2.1	YOLO v1 – Unified, Real-Time Object Detection	68
5.2.2	YOLO v2 – Better, Faster, Stronger	69
5.2.3	YOLO v3 – An Incremental Improvement	70
5.3	Deteção em tempo real	72
5.3.1	Implementação em Windows	72
5.3.2	Implementação em Raspberry Pi	74
5.3.3	Otimização da rede neuronal	75
6	Validação e resultados	79
6.1	Testes realizados	79
6.1.1	Deteção de peões	79
6.1.2	Análise de tráfego	85
6.1.3	Sistemas de iluminação	94
6.2	Análise de resultados	95
6.2.1	Integração dos subsistemas	98
7	Considerações finais	101
7.1	Conclusões	101
7.2	Trabalho futuro	102
	Bibliografia	105

Lista de Figuras

1.1	Atropelamentos com e sem fuga do condutor, nas estradas portuguesas	1
1.2	Atropelamentos nas passadeiras ou a menos de 50m	2
2.1	As 10 principais causas de morte no mundo [4]	6
2.2	Passadeira iluminada em Eerbeek (Stuijvenburchstraat) [6]	7
2.3	Sinalização luminosa no pavimento em Bodegraven [7]	7
2.4	Sinalização luminosa no pavimento em Tel Aviv [8]	8
2.5	Fotografia com esquema simplista da ativação do processo	9
2.6	Esquema de aplicação do IRWL [9]	9
2.7	Passadeira inteligente em Cambrils [11]	10
2.8	Fotografia do sistema aplicado em Guimarães [12]	10
2.9	Passadeira inteligente em Matosinhos [13]	11
2.10	Passadeira inteligente em Badajoz [15]	12
2.11	Aviso via smartphone na passadeira de Ilsan [16]	12
2.12	Passadeira “Starling CV” [17]	13
2.13	Principais causas de acidentes junto a passadeiras na Hungria [18]	14
2.14	Princípio de iluminação assimétrica [20]	15
2.15	Princípio piezoelétrico [23]	17
2.16	Exemplo de sensor Passive Infrared (PIR) [24]	17
2.17	Funcionamento de um ultrassons [25]	18
2.18	Exemplo de sensor de micro-ondas [26]	19
2.19	Exemplo de câmara (Flexidome IP Startlight 7000 VR) [27]	19
3.1	Esquema de aplicação STAP para 1 faixa de rodagem	22
3.2	Esquema de aplicação STAP para 2 faixas de rodagem	22
3.3	Diferença entre <i>lux</i> , <i>lúmen</i> e <i>candela</i> [28]	24
3.4	Zonas de deteção STAP para 1 e 2 faixas de rodagem	25
3.5	Esquema de passadeira do tipo “zebra” [30]	26
3.6	Distância de reação em função do tempo, para diferentes velocidades	27
3.7	Distância de travagem em função do atrito, para diferentes velocidades	28
3.8	Modo de poupança de energia STAP	30
3.9	Representação da deteção de um peão durante a noite	31
3.10	Representação da deteção um veículo dentro do limite de velocidade	31
3.11	Representação da deteção um veículo em excesso de velocidade	32
3.12	Representação da deteção um excesso de velocidade seguido de um peão	32
4.1	Primeira versão da base de suporte do protótipo	36

4.2	Primeira versão da estrutura do protótipo (separada)	36
4.3	Estrutura definitiva do protótipo	37
4.4	Diagrama de blocos de um poste	38
4.5	Módulo câmara para Raspberry Pi v1.3	39
4.6	Radar Stalker Stationary Speed Sensor II	41
4.7	Sensor de luminosidade TSL2591	41
4.8	Tira de LEDs RGB endereçável WS2812B 5050	42
4.9	Luminária LED “Roma”	43
4.10	Fluxograma funcional da tarefa T1	44
4.11	Fluxograma funcional da tarefa T2 (esquerda) e T3 (direita)	45
4.12	Fluxograma funcional do fluxo principal	46
4.13	Placa de alimentação e amplificação de sinais	47
4.14	Conversor de nível lógico bidirecional [48]	48
4.15	Esquema do amplificador com base em BJTs	48
4.16	Modo de operação do driver [49]	49
4.17	Simulação do circuito da figura 4.15	49
4.18	Esquema do circuito de alimentação	50
4.19	Simulação do circuito da figura 4.18	50
4.20	Passadeira escolhida para instalação do protótipo	51
4.21	Conexão do módulo de câmara (esquerda) e pinout (direita) [53]	52
4.22	Esquema com as dimensões da luminária	53
4.23	Esquema com as dimensões do radar [54]	53
4.24	Modos de instalação da luminária: a) em braço; b) <i>post-top</i>	54
4.25	Adaptador e alimentação do radar	54
4.26	Visão lateral do poste, voltado para a via (detecção de velocidade)	55
4.27	Visão lateral do poste, voltado para a passadeira (detecção de peões)	56
4.28	Esquema representativo da zona de detecção do radar [54]	57
4.29	Diagrama com a geometria da instalação do radar [54]	57
4.30	Esquema de instalação alternativa do radar [54]	60
4.31	Leitura de dados do radar em tempo real [54]	61
5.1	Camada convolutiva [57]	66
5.2	Metodologias de <i>pooling</i> [57]	66
5.3	<i>Precision</i> e <i>recall</i> [59]	67
5.4	Intersection over Union [60]	68
5.5	Arquitetura da primeira versão do YOLO [62]	69
5.6	Arquitetura da terceira versão do YOLO [65]	71
5.7	Velocidade/precisão de diferentes métodos - COCO AP (média e 0,5 IoU) [64]	72
5.8	Exemplo nº1 - YOLOv3 (esquerda) vs YOLOv3-tiny (direita)	73
5.9	Exemplo nº2 - YOLOv3 (esquerda) vs YOLOv3-tiny (direita)	73
5.10	Exemplos de funcionamento do YOLOv3-tiny em <i>Raspberry Pi</i>	74
6.1	Exemplo de anotação correta de objetos da classe <i>person</i>	80
6.2	Exemplo de anotação parcialmente errada de objetos da classe <i>person</i>	81
6.3	Curva <i>precision/recall</i> : YOLOv3-tiny (esquerda) e versão melhorada (direita)	82
6.4	mAP para a classe <i>person</i> : YOLOv3-tiny (esquerda) e versão melhorada (direita)	82
6.5	Número de objetos a detetar da classe <i>person</i>	82

6.6	Objetos detetados: YOLOv3-tiny (esquerda) e versão melhorada (direita) . .	83
6.7	Curva <i>precision/recall</i> : YOLOv3-tiny (esquerda) e versão melhorada (direita)	83
6.8	mAP para a classe <i>person</i> : YOLOv3-tiny (esquerda) e versão melhorada (direita)	84
6.9	Objetos detetados: YOLOv3-tiny (esquerda) e versão melhorada (direita) . .	84
6.10	Exemplo de funcionamento da versão melhorada da rede neuronal	85
6.11	Descrição do <i>survey</i>	86
6.12	Teste ao radar - Contagem de veículos	87
6.13	Teste ao radar - Velocidades mais frequentes por tipo de viatura	87
6.14	Teste ao radar - Cumprimento dos limites de velocidade	88
6.15	Teste ao radar - Excessos de velocidade detetados	88
6.16	Teste ao radar - Marcação no tempo de todas as velocidades	89
6.17	Teste ao radar - Número de deteções a cada meia hora	89
6.18	Resumo dos resultados do teste ao radar	90
6.19	Montagem em ângulo para tarefas de classificação [82]	92
6.20	Interface para controlo das tiras de LED RGB	95
6.21	Vista de topo durante o teste de luminosidade - modo de poupança de energia	98
6.22	Deteção de um excesso de velocidade	99
6.23	Deteção de um peão que pretende atravessar a via	99
6.24	Travessia em segurança com o sistema STAP	100

Lista de Tabelas

1.1	Número de vítimas de atropelamento em Portugal	2
2.1	Breve comparação das passadeiras analisadas no estado da arte	13
2.2	Breve comparação dos métodos de deteção de peões	20
3.1	Pré-requisitos do sistema	23
3.2	Valores médios de iluminância (<i>lux</i>) [29]	25
3.3	Distância média necessária para travar um veículo	29
3.4	Resumo de funcionamento STAP	30
4.1	Alternativas de radar para deteção de velocidade	39
4.2	Comparação complementar entre os radares <i>Capsys</i> e <i>Stalker</i>	40
4.3	Alternativas de luminárias para iluminação de presença	43
4.4	Divisão do código em tarefas	44
4.5	Tempos de passagem consoante os peões detetados	47
4.6	<i>Pinout</i> da <i>Raspberry Pi</i> e respetivas conexões ao sistema	52
4.7	Portas COM disponibilizadas pelo radar	54
4.8	Valores aproximados da zona de deteção do radar	58
4.9	Resumo das configurações básicas do radar	59
4.10	Resumo de algumas das configurações avançadas do radar	61
4.11	Protocolos de <i>streaming</i> suportados pelo radar	62
5.1	Comparação da precisão de diferentes detetores	72
5.2	Características da plataforma para teste da rede neuronal	73
5.3	Velocidade média de processamento da rede neuronal	74
5.4	Características da <i>Raspberry Pi 3B+</i>	74
5.5	Comparação de diferentes <i>dataset</i> de pedestres [72]	76
6.1	Resumo do desempenho da rede neuronal no âmbito da deteção de pessoas	84
6.2	Resultados do 1º teste de contagem de veículos	91
6.3	Resultados do 2º teste de contagem de veículos	92
6.4	Valores mínimos da intensidade de reflexão do sinal antes do treino	93
6.5	Valores mínimos da intensidade de reflexão do sinal após os treinos	93
6.6	Número de veículos observados e respetiva classificação	94
6.7	Valores de luminância vertical no centro da passadeira	95

Acrónimos

ADC	Analog-to-Digital Converter
ANN	Artificial Neural Networks
ANSR	Autoridade Nacional da Segurança Rodoviária
AmiEs	Ambient Intelligence and Embedded Systems
AP	Average Precision
API	Application Programming Interfaces
BJT	Bipolar Junction Transistor
CNN	Convolutional Neural Network
COCO	Common Objects in Context
CPU	Central Processing Unit
CUDA	Compute Unified Device Architecture
DETI	Departamento de Eletrónica, Telecomunicações e Informática
DFIS	Departamento de Física
EUA	Estados Unidos da América
FHWA	Federal Highway Administration
FPS	Frames Por Segundo
GPIO	General-Purpose Input/Output
GPS	Global Positioning System
GPU	Graphics Processing Unit
GUI	Graphical User Interface
I2C	Inter-Integrated Circuit
IEEE	Institute of Electrical and Electronics Engineers
IoU	Intersection over Union

IRWL	In-Roadway Warning Light
IT	Instituto de Telecomunicações
JSON	JavaScript Object Notation
KICT	Korea Institute of Civil Engineering and Building Technology
LED	Light Emitting Diode
mAP	Mean Average Precision
MOSFET	Metal Oxide Semiconductor Field Effect Transistor
NaN	Not a Number
NIST	National Institute of Standards and Technology
NRZ	Non-Return to Zero
OMS	Organização Mundial de Saúde
PCB	Printed Circuit Board
PIR	Passive Infrared
PRP	Prevenção Rodoviária Portuguesa
PVC	Policloreto de Vinilo
PWM	Pulse-Width Modulation
RAM	Random Access Memory
R&D	Research and Development
RGB	Red, Green, Blue
SMD	Surface Mount Device
SSD	Single Shot MultiBox Detector
STAP	Sinalização Transversal Ativa de Passadeiras
SUV	Sport Utility Vehicles
VNC	Virtual Network Computing
VOC	Visual Object Classes
VPN	Virtual Private Network
VTTI	Virginia Tech Transportation Institute
YOLO	You Only Look Once

Capítulo 1

Enquadramento

Este capítulo introdutório pretende contextualizar a presente dissertação.

Ao longo de quatro subcapítulos apresentam-se as principais razões que levaram à escolha deste projeto, os objetivos que se pretendem alcançar, a estrutura do documento desenvolvido e, por fim, os apoios que permitiram concretizar o projeto.

1.1 Motivações

Nos dias de hoje, as passeadeiras continuam a ser dos locais mais perigosos da via pública, registando-se um aumento de feridos e vítimas mortais. Zonas pouco iluminadas durante a penumbra ou noite e roupas escuras são fatores que contribuem para agravar o problema.

Os dados do relatório anual de sinistralidade rodoviária de 2018 da Autoridade Nacional da Segurança Rodoviária (ANSR) sustentam estas afirmações e estão representados nos gráficos das figuras 1.1 e 1.2. Constata-se um ligeiro aumento do número de atropelamentos de pedestres perto de passagens sinalizadas, relativamente a 2017, com destaque para as ocorrências dentro das localidades [1].

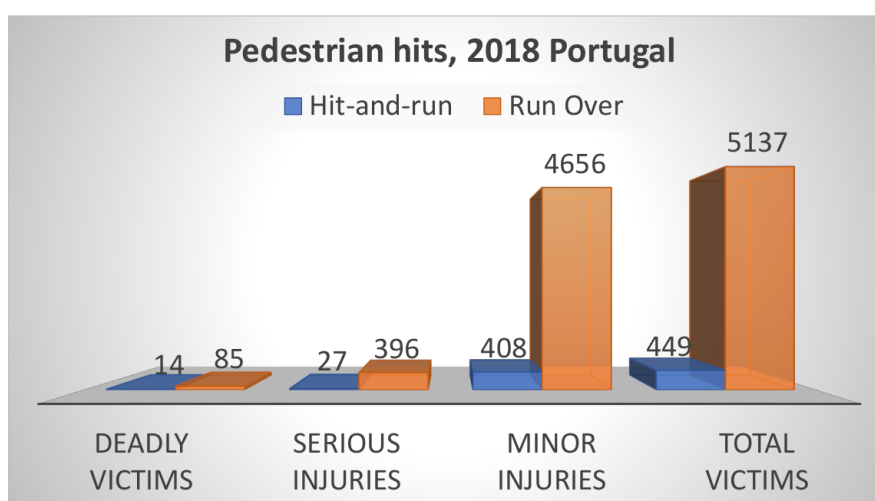


Figura 1.1: Atropelamentos com e sem fuga do condutor, nas estradas portuguesas

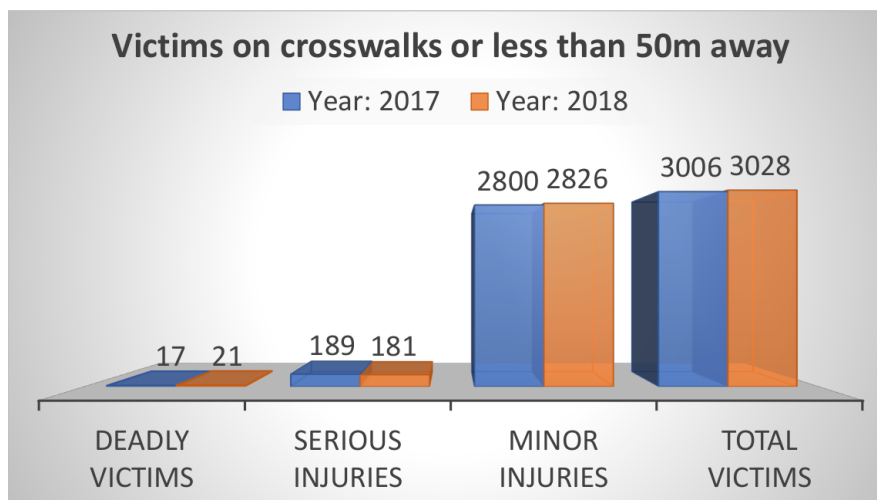


Figura 1.2: Atropelamentos nas passadeiras ou a menos de 50m

Podemos ainda cruzar os dados dos gráficos anteriores e através da tabela 1.1 concluir sobre o local das ocorrências. Cerca de 54% do total de vítimas ocorreram em acidentes junto das passagens para peões, correspondendo a 21% das mortes por atropelamento, 43% dos feridos graves e 56% dos feridos ligeiros. É, portanto, do interesse comum reduzir a taxa de acidentes rodoviários nestes locais.

Portugal, 2018	Mortes	Feridos graves	Feridos ligeiros	Vítimas no total
Nas passadeiras	21	181	2826	3028
No total	99	423	5064	5586

Tabela 1.1: Número de vítimas de atropelamento em Portugal

Outra situação particularmente crítica é o facto dos transeuntes se deslocarem frequentemente a olhar para o telemóvel, sem que consigam observar as viaturas que se aproximam. Um estudo da Prevenção Rodoviária Portuguesa (PRP) realizado em janeiro de 2017, revela que mais de 15% dos 5223 peões observados em Lisboa atravessaram as passadeiras distraídos ao telemóvel [2].

Surgem também cada vez mais veículos elétricos em circulação que, por se deslocarem silenciosamente, podem passar despercebidos aos peões menos atentos.

Estes exemplos demonstram que, apesar de existirem diversos estudos direcionados aos comportamentos de risco do automobilista, os mesmos podem partir do viandante e representam perigos desnecessários que as tecnologias de deteção podem ajudar a combater, quando aliadas à projeção de informação luminosa.

Sabendo disto, a empresa *F.L.Gaspar* que atua em áreas como a sinalização, segurança rodoviária e gestão de tráfego, propôs o desenvolvimento de uma solução de passadeiras inteligentes que integrasse toda a tecnologia necessária num conjunto de postes comunicantes e de fácil instalação, sem necessidade de intervenções externas na via. Estes devem incluir

sistemas de deteção dos intervenientes, controlo de iluminação e projeção de sinalização sobre a estrada para facilitar a chamada de atenção a peões e automobilistas.

Neste sentido foi proposta uma parceria entre a Universidade de Aveiro e a empresa *Microio*, de modo a avaliar possíveis soluções para o desenvolvimento desta passadeira de nova geração.

1.2 Objetivos

Este sistema, que por uma questão de comodidade será designado por STAP, é uma solução para o alerta de condutores e peões, mediante a identificação de potenciais situações de perigo nas proximidades de uma passadeira. Sendo um projeto criado de raiz com vista a uma futura implementação, é necessário seguir um conjunto de etapas bem definidas:

- Conhecer os sistemas similares.
- Estabelecer os requisitos que o sistema deve cumprir.
- Estudar tecnologias úteis existentes no mercado.
- Reunir uma lista do material essencial e adquiri-lo.
- Definir a arquitetura do sistema de controlo.
- Implementar.
- Executar testes parciais dos diferentes módulos que compõem a solução.
- Integrar os subsistemas e conceber o protótipo.
- Testar globalmente o protótipo e efetuar uma análise crítica.

A parte ótica que consiste na projeção da sinalização na estrada, por ser responsabilidade do Departamento de Física, não fará parte dos objetivos do projeto.

Espera-se que o resultado final desta dissertação seja um versão funcional, ainda que provisória, do protótipo e que este documento reflita o processo de criação e os fundamentos de engenharia aplicados.

1.3 Estrutura do documento

No cerne desta dissertação estão sete capítulos. Cada capítulo divide-se em sub-capítulos e estes por vezes em sub-seções, de acordo com a dependência e relevância, facilitando assim a compreensão do documento.

Capítulo 1

Contextualiza o projeto e salienta as razões que levaram à sua escolha, definindo os seus objetivos. Apresenta também a organização do documento e os principais apoios.

Capítulo 2

Espelha o panorama geral de soluções similares, separadas nas vertentes comerciais e académicas. Este estado da arte integra também uma análise às técnicas de deteção de peões mais utilizadas.

Capítulo 3

Explica de um modo geral a solução proposta. Aborda as suas características, modos de funcionamento e outras especificações.

Capítulo 4

Expõe em detalhe a conceção e implementação de um protótipo do ponto de vista técnico.

Capítulo 5

Remete à rede neuronal, ao seu estudo e implementação detalhada no âmbito da deteção de peões.

Capítulo 6

Descreve os testes parciais e globais realizados ao sistema e analisa os resultados de forma crítica, de modo a validar o funcionamento do projeto.

Capítulo 7

Evidencia os objetivos atingidos e algumas propostas para melhoria do trabalho realizado.

1.4 Apoios

Só foi possível desenvolver um protótipo para um projeto desta envergadura graças ao apoio de algumas entidades:

- Em primeiro lugar graças à empresa *Microio*, por ter permitido o desenvolvimento em contexto empresarial e colocado os seus recursos à disposição (material, *firmware* e até mesmo o tempo dispensado pelos seus colaboradores).
- Segue-se, obviamente, a Universidade de Aveiro, o Departamento de Eletrónica, Telecomunicações e Informática (DETI) e o meu orientador, por permitirem a parceria referida anteriormente e pela aquisição do radar a título definitivo.
- Por fim, graças à empresa *Lux Concept*, que teve a amabilidade de produzir e ceder temporariamente duas luminárias Light Emitting Diode (LED).

Uma palavra ainda para a organização da conferência internacional de Ambient Intelligence and Embedded Systems (AmiEs), na qual tive a honra de participar a 19 de setembro de 2019 [3]. Ter a oportunidade de apresentar o trabalho e trocar ideias com especialistas de todo o mundo, foi sem dúvida uma experiência enriquecedora que permitiu melhorar e validar este projeto.

Capítulo 2

Estado da arte

Para dar início ao projeto e prever a sua viabilidade, surge a necessidade de estudar o nível de maturidade das soluções já existentes. Pretende-se no final deste capítulo conseguir definir as tecnologias que melhor se adaptam ao sistema e evidenciar fatores diferenciadores.

O estado da arte é composto por três subcapítulos que retratam o conceito de passadeira inteligente de diferentes perspetivas, dando particular destaque às metodologias de deteção utilizadas.

Vamos começar por fazer um estudo sobre soluções comerciais semelhantes e conhecidas ao redor do mundo, tendo como diretrizes de pesquisa os seguintes conceitos-chave: deteção de peões, controlo de velocidade, projeção de sinalização, ajuste de luminosidade, comunicações sem fios e otimização dos tempos de passagem. Esta pesquisa deve culminar numa análise comparativa sumária de características e funcionalidades.

Segue-se o enumerar de alguns trabalhos académicos pertinentes no contexto da dissertação e que serão utilizados ao longo da mesma como base teórica para justificar decisões mais complexas.

Por último, e por se considerar crucial no panorama da solução, retratam-se diferentes técnicas de deteção automática de peões.

2.1 Soluções comerciais

Como é do conhecimento geral, nos dias de hoje existem inúmeras passadeiras de peões que se auto-denominam de inteligentes. Isto surge naturalmente como resultado dos avanços tecnológicos que vivemos e do crescente interesse por parte de alguns governos e entidades, em particular nos países desenvolvidos, de encontrar soluções para reduzir os acidentes rodoviários.

De acordo com a Organização Mundial de Saúde (OMS) [4], os acidentes de trânsito foram responsáveis por 1.4 milhões de mortes em 2016, sendo a oitava maior causa de morte no mundo, como se observa na figura 2.1, à frente de doenças como a tuberculose e o VIH¹/SIDA². Mas o verdadeiro impacto dos acidentes de trânsito na sociedade vai muito para além destes números.

Falamos também de grandes gastos financeiros que advém, por exemplo, de custos hospitalares, indemnizações, perdas de produção e gastos com materiais, bem como de implicações

¹Vírus da Imunodeficiência Humana

²Síndrome da Imunodeficiência Adquirida

Top 10 global causes of deaths, 2016

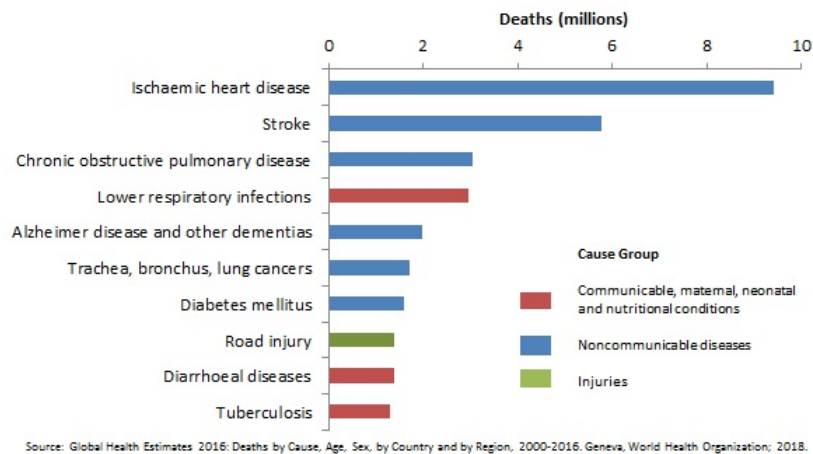


Figura 2.1: As 10 principais causas de morte no mundo [4]

psicológicas que resultam de situações de incapacidade, traumas ou outros distúrbios, não apenas por parte das vítimas, mas também das suas famílias e pessoas mais próximas, que precisam de aprender a lidar com estas adversidades [5].

Por todo este peso social e económico é possível compreender a aposta em soluções preventivas como sejam as passadeiras inteligentes.

2.1.1 Soluções de iluminação

Embora por vezes se considerem passadeiras inteligentes, já que utilizam a tecnologia de uma forma a dar resposta a um problema comum, algumas soluções baseiam-se apenas na iluminação do local.

A correta iluminação de uma passadeira permite diminuir o risco de atropelamento, por aumentar a visibilidade dos intervenientes, em particular durante a noite. No entanto, ao optar por uma solução estática, podemos não estar a transmitir nenhuma sensação de perigo aos utilizadores, o que pode levar a uma habituação e tornar o sistema pouco eficiente.

Seguem-se três exemplos que se destacaram no âmbito da iluminação:

Eerbeek, Holanda

A empresa *Lighted Zebra Crossing B.V.* desenvolveu na Holanda uma solução simples que passa por substituir a tinta utilizada para marcar a passadeira no alcatrão por luzes LED, como se verifica na figura 2.2. Estas luzes são ativadas assim que a restante iluminação pública for acesa, tratando-se de uma passagem elevada normal durante o dia, com visibilidade reforçada durante a noite.

Bodegraven, Holanda

Ainda na Holanda, encontramos uma alternativa mais próxima da que se pretende alcançar com este projeto em termos de sinalização luminosa.



Figura 2.2: Passadeira iluminada em Eerbeek (Stuijvenburchstraat) [6]

Desenvolvido pela *HIG Traffic Systems*, o projeto *+Lichtlijn* que em português significa “+linha de luz”, é aplicável a passadeiras semaforizadas e utiliza uma linha de LEDs embutida no pavimento, que reproduz o comportamento do semáforo que lhe é adjacente [7].

O objetivo é captar a atenção dos peões que atravessam a olhar para o telemóvel, um fenómeno recente e cada vez mais recorrente, que deu origem ao termo “smombie”, um anglicismo que resulta da junção das palavras *smartphone* e *zombie*. Estes pedestres caracterizam-se pelo seu andar lento e falta de atenção ao ambiente que os rodeia. São considerados um perigo na via pública devido à redução significativa do seu campo de visão, provocada pelo uso do telemóvel.



Figura 2.3: Sinalização luminosa no pavimento em Bodegraven [7]

Tel Aviv, Israel

Tal como na solução anterior, temos um semáforo especial desenhado para combater os chamados “smombies” - ver figura 2.4.

Espera-se que a luz ao nível do chão seja visível por um maior número de utilizadores. Ainda assim, há relatos de quem tenha atravessado a zona em questão ao telemóvel, sem reparar no sinal luminoso [8].

Sistemas semelhantes podem ser encontrados em Singapura, na Austrália ou na Alemanha.



Figura 2.4: Sinalização luminosa no pavimento em Tel Aviv [8]

2.1.2 Soluções com detecção de peões

Anteriormente constatamos a importância de uma boa iluminação e que o uso de projeção da sinalização é uma solução válida para o combate aos “smombies”, no entanto, para que o conceito de passadeira inteligente seja verdadeiramente aplicado precisamos de ir mais longe com o nosso projeto.

Olhamos agora para soluções um pouco mais completas, que juntam a detecção de peões à gestão da iluminação/sinalização.

Para facilitar a exposição, as soluções encontram-se subdivididas de acordo com três tipos de detecção: por infravermelhos, por sensor de pressão e por câmara de vídeo. Nenhum dos exemplos que se seguem é semaforizado.

2.1.2.1 Detecção por infravermelhos

Cascais, Portugal

Esta é uma passadeira elevada que se encontra junto à saída do parque de estacionamento subterrâneo do *shopping* de Cascais.

É composta por iluminação vertical de ambos os lados (verde/vermelha), um projetor de topo que emite o sinal de STOP no chão e um sinal de passagem para peões com os cantos luminosos. Todos estes componentes são acionados pelo peão através de duas células de infravermelhos, como representado no esquema da figura 2.5.

O comportamento do sistema foi analisado numa visita conjunta ao local pelas empresas *Microio* e *F.L.Gaspar*. Observaram-se algumas limitações e falhas no sistema, tais como:

- perante um mudança de estado, a proximidade excessiva da célula à passadeira não permite tempo de reação suficiente aos condutores.
- a detecção em linha numa região relativamente reduzida, leva a que por vezes a célula não seja ativada.
- a interrupção do feixe leva sempre à ativação do sistema durante 7s, e se acontecer apenas quando o peão está a sair da passadeira, gera-se um falso positivo, ficando o sistema ativo sem ninguém a atravessar.

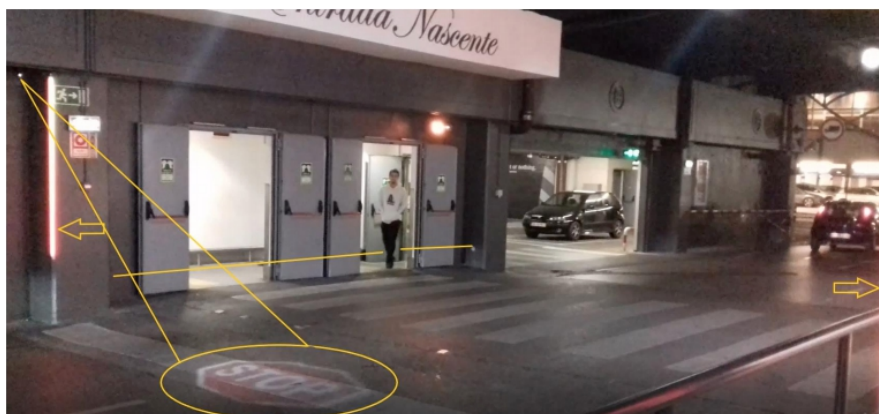


Figura 2.5: Fotografia com esquema simplista da ativação do processo

Estados Unidos da América

Segue-se um sistema bastante consolidado e largamente implementado, em particular nos Estados Unidos da América (EUA).

Desenvolvido pela *Lightguard Systems*, dá-se pelo nome de *Smart Crosswalk In-Roadway Warning Light (IRWL)*. É composto por módulos LED colocados na estrada em grupos de 5 ou 7 de cada lado da passadeira, orientados de forma a criar uma zona de convergência direcionada ao condutor, que deve conseguir observar o sistema a uma distância segura.

É possível ativar a passadeira tanto por sensores infravermelhos bidirecionais como por botão de pressão, estando disponíveis controladores básicos ou avançados, tendo em conta a necessidade (ou não) de armazenar dados, programar horários específicos e energia de *backup* no caso de falha de energia [9].



Figura 2.6: Esquema de aplicação do IRWL [9]

2.1.2.2 Detecção por sensores de pressão

Cambrils, Espanha

Desenvolvidas inicialmente pela *Interlight* e melhoradas pela *Stepvial*, as passadeiras inteligentes semelhantes à da figura 2.7 estão presentes em diversos locais do mundo, incluindo Portugal, México e Espanha.

O sistema é frequentemente instalado pela *Llumtraffic* (pelo menos em Espanha) e a título de exemplo temos as passadeiras de Soho, Málaga e Tarragona, Cambrils.

Em Málaga foram utilizados sensores volumétricos para detetar a passagem e as obras custaram 11.841,43€. Em Cambrils foram utilizados sensores de pressão, com um custo aproximado de 10.000,00€ [10]. Ambas utilizam tecnologia LED para iluminar as laterais da passadeira a partir do solo e placas de sinalização luminosas.

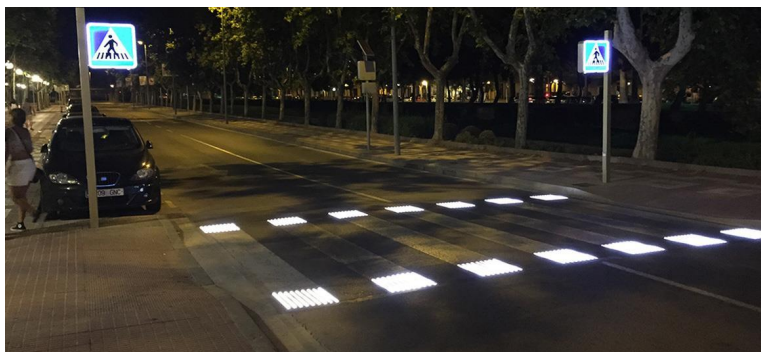


Figura 2.7: Passadeira inteligente em Cambrils [11]

2.1.2.3 Detecção por câmara

Guimarães, Portugal

Instalada na Avenida D. João IV pela *Communicate Way* em parceria com a *Bosch*, a *Siemens* e a *Exporlux*. Tal como na passadeira de Cascais foi realizada uma visita conjunta ao local pela *Microio* e a *F.L. Gaspar*.

O sistema é composto por dois postes de aproximadamente 3m, com luzes alaranjadas intermitentes viradas para o condutor e para cima. Um poste utiliza uma câmara modelo *Flexidome IP Startlight 6000/7000 VR* (figura 2.19) responsável pela deteção de peões e o outro detém uma luminária para iluminação noturna da passadeira.

Foram verificados vários falsos positivos que descredibilizam o sistema, bem como zonas de fraca deteção. Apresenta o registo de patente nacional número 110379.



Figura 2.8: Fotografia do sistema aplicado em Guimarães [12]

Matosinhos, Portugal

A *Habidom* é uma empresa que exerce a sua atividade na área da sinalização rodoviária, com passadeiras inteligentes instaladas em cidades como Matosinhos, Loulé e Viseu.

Em Matosinhos o sistema utiliza uma câmara com sensor multi-ponto de alta precisão para detetar os peões e acende uma luz intensa com o objetivo de iluminar a passadeira apenas no modo noturno, ficando ligada durante 5 minutos desde o momento da deteção, como se observa na figura 2.9.

Em Loulé o sistema é ativado por sensores infravermelhos, surgindo luzes de cor alaranjada no topo do sinal de trânsito ao detetar o peão. Em Viseu temos novamente a presença destes sinais luminosos (Sinais H7 sincronizados com marcadores SR45) [13].



Figura 2.9: Passadeira inteligente em Matosinhos [13]

2.1.3 Soluções com deteção de peões e veículos

De momento, as soluções mais evoluídas de todas são aquelas que conseguem aliar eficazmente a deteção de peões com a deteção de velocidade, sem esquecer as questões relacionadas com a iluminação.

A deteção da velocidade de aproximação dos veículos é uma mais valia já que permite não só atuar no sentido de sensibilizar o condutor, na tentativa de controlar riscos desnecessários, mas também porque torna possível avisar os peões do perigo de atravessar em determinado momento, caso o condutor ignore os avisos para redução da velocidade.

Badajoz, Espanha

Situada na Avenida Villanueva, no topo da rua Díaz Brito, em Badajoz.

É uma iniciativa no âmbito do projeto “Smart City” que pretende melhorar a cidade do ponto de vista da segurança rodoviária e mobilidade sustentável. Um sistema semelhante já tinha sido antes instalado em Burgos, próximo da “plaza del Rey”.

Utiliza um sistema de câmaras que detetam a presença de peões e veículos através de sensores, com um custo a rondar os 26.000,00€ [14].

Se um carro se aproximar a mais de 30 km/h, uma luz vermelha acende para o peão, alertando-o. Durante o dia surge um sinal de cor âmbar a avisar os condutores caso haja uma travessia e de noite além dos sinais já referidos é aceso um foco de luzes LED.



Figura 2.10: Passadeira inteligente em Badajoz [15]

Ilsan, Coreia do Sul

Desenhado por engenheiros do Korea Institute of Civil Engineering and Building Technology (KICT), este sistema combina uma aplicação, um projetor e luzes intermitentes.

Quando o radar e a câmara térmica detetam um veículo e um pedestre em simultâneo é projetado um aviso no chão a dizer que uma viatura se aproxima, as luzes intermitentes acendem e é despoletada um notificação através da *app*, alertando os peões - figura 2.11 [16].

O sistema tem um custo de instalação aproximado de 12.000,00€ por passadeira.

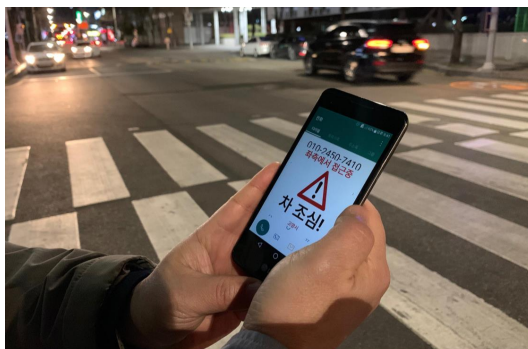


Figura 2.11: Aviso via smartphone na passadeira de Ilsan [16]

Londres, Inglaterra

Este projeto piloto foi implementado em Londres pela *Umbrellium*, sendo uma das passadeiras mais mediáticas pelo efeito visual que provoca - figura 2.12.

Funciona tendo por base uma rede neuronal e um sistema de visão por computador, que monitorizam e classificam os objetos em tempo real, prevendo as suas trajetórias [17].

A matriz de LEDs no pavimento permite apresentar limitadores luminosos para veículos e ciclistas, bem como revelar uma passadeira ao detetar peões que pretendam atravessar. Caso estes atravessem antes de o sistema revelar a passadeira, surgem setas vermelhas na tentativa de prever a trajetória e alertar os condutores.

A complexidade do sistema parece torna-lo pouco intuitivo e de difícil implementação, no entanto, segundo a empresa, está a ser desenvolvido um novo exemplar.



Figura 2.12: Passadeira “Starling CV” [17]

2.1.4 Análise comparativa sumária

Perante todos os exemplos explanados, surge a necessidade de sumariar as principais características de cada solução e enquadrar o projeto STAP, indicando semelhanças e fatores diferenciadores. A tabela 2.1 representa esta comparação de forma muito breve.

Devemos ressaltar que o objetivo não passa por avaliar as soluções, já que para isso seria necessária uma análise estatística intensiva de inúmeros parâmetros, impraticável no âmbito desta dissertação.

Localização	Deteção Peões	Deteção Veículos	Alertas Peões	Alertas Veículos	Simplicidade de instalação ³
Eerbeek	×	×	×	×	×
Tel Aviv	×	×	×	×	×
Bodegraven	×	×	×	×	×
Cascais	✓	×	×	✓	✓
EUA	✓	×	×	✓	×
Cambrils	✓	×	×	✓	×
Guimarães	✓	×	×	✓	✓
Matosinhos	✓	×	×	✓	✓
Badajoz	✓	✓	✓	✓	×
Ilsan	✓	✓	✓	✓	×
Londres	✓	✓	✓	✓	×
Aveiro STAP	✓	✓	✓	✓	✓

Tabela 2.1: Breve comparação das passadeiras analisadas no estado da arte

Das 11 soluções analisadas, apenas 3 (Badajoz, Ilsan e Londres) são capazes de detetar velocidades dos veículos e emitir alertas para os peões, precavendo esta situação.

Por outro lado, são também 3 (Cascais, Guimarães e Matosinhos), as soluções que não

³Não serem necessárias intervenções significativas no pavimento da via pública

precisam de grandes intervenções na via pública, permitindo uma instalação fácil em grande parte das passadeiras para peões.

O STAP pretende juntar as melhores características aqui descritas, aliando a deteção de peões e de velocidade a um sistema de instalação rápido e prático.

2.2 Soluções académicas

Uma visão mais académica do problema pode ser interessante para aumentar conhecimentos no âmbito da segurança rodoviária, do comportamento e tempos de reação do ser humano, bem como descobrir novos protótipos e tecnologias.

Através do estudo de alguns trabalhos académicos nesta área, espera-se que seja possível fortalecer as bases deste projeto.

Resumem-se em seguida um conjunto de informações relevantes que podem ser encontradas em cada um dos artigos estudados.

2.2.1 Evaluation of human behaviour at pedestrian crossings [18]

Consiste numa análise às causas dos acidentes junto a passadeiras, levando a cabo uma comparação do comportamento dos condutores e dos peões, antes e após a implementação de medidas de segurança adicionais na estrada.

Expõe dados estatísticos sobre as causas dos acidentes na Hungria entre 2004 e 2013, com destaque para o gráfico da figura 2.13.

Avalia também os tempos de espera dos pedestres e o tempo de reação dos condutores.

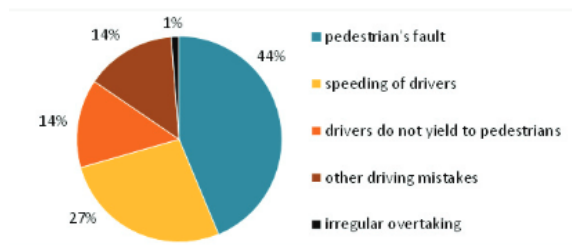


Figura 2.13: Principais causas de acidentes junto a passadeiras na Hungria [18]

Ilacões:

Permitiu concluir que passadeiras com luzes âmbar a piscar, ilhas de refúgio e semáforos, levam a comportamentos mais cuidadosos por parte dos condutores.

Confirmou as suspeitas de que a maioria dos acidentes são culpa dos pedestres e do excesso de velocidade dos veículos, justificando novamente a necessidade da deteção de velocidade e alertas direcionados aos peões.

2.2.2 Passive pedestrian detection at unsignalized crossings [19]

Explica a importância dos condutores saberem quando é que um peão pretende atravessar, dando conta das razões que levam os botões de pressão a serem inviáveis. Em alternativa, descrevem-se 5 soluções passivas para deteção de peões: PIR, ultrassons, doppler radar, imagem de vídeo e piezoelétrico.

Foram testados num projeto em Portland 3 das soluções apresentadas (PIR, ultrassons e radar) e avaliada a sua performance, necessidade de manutenção e custo.

Ilacões:

Decidiu-se não utilizar qualquer botão de pressão no protótipo e foram levadas em consideração todas as alternativas apresentadas como potenciais soluções.

2.2.3 Innovative solutions for improving safety at pedestrian crossings [20]

Aborda diferentes *designs* para passadeiras com e sem sinalização, descrevendo medidas adicionais que podem ser implementadas para redução da velocidade.

Complementa o documento anterior [19], na medida em que impõe uma visão mais atual das técnicas de deteção de peões, focando-se particularmente em soluções de vídeo e processamento de imagem.

Introduz o conceito de iluminação assimétrica de uma passadeira para melhoria da visibilidade - figura 2.14.

Por fim, surgem ao longo do texto exemplos da realidade Polaca, dando conta de alguma informação estatística e sistemas implementados.

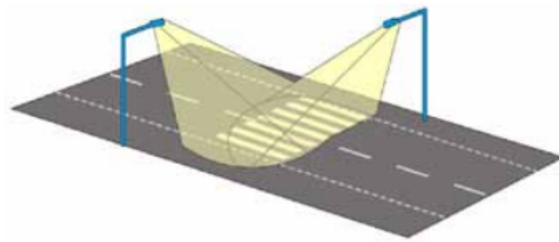


Figura 2.14: Princípio de iluminação assimétrica [20]

Ilacões:

A análise mais recente às metodologias de deteção fizeram com que o processamento de imagem e as soluções baseadas em câmara ganhassem força em relação aos restantes métodos, apesar da sua complexidade.

Compreendeu-se a necessidade de iluminar não só a passadeira, mas também a zona onde os pedestres aguardam para atravessar. Estes locais devem permanecer mais iluminados do que o *background*.

2.2.4 Informational report on lighting desing for midblock crosswalks [21]

Apresenta informações sobre parâmetros de iluminação para passadeiras, baseado em experiências estáticas e dinâmicas realizadas no Virginia Tech Transportation Institute (VTTI). As variáveis nestes testes foram o tipo de lâmpada, a luminância vertical, a cor das roupas, a posição do pedestre e o brilho.

- **Experiência estática:** Tempo necessário para o observador ver o pedestre.
- **Experiência dinâmica:** Distância a partir da qual o observador vê o pedestre.

Definem-se os conceitos de visibilidade, contraste, luminância vertical, entre outros e as respectivas equações teóricas.

Por último, abordam-se condições necessárias para um *design* adequado, de acordo com o tipo de luminária, a altura a que é aplicada e a localização da passadeira.

Ilacões:

Permite referenciar conceitos e fórmulas que podem ser utilizadas ao longo da dissertação para otimizar o *layout* da instalação, levando a cabo testes de luminosidade que garantam os valores recomendados, como por exemplo, um mínimo de 20lux verticais a 1,5m de altura da estrada.

Prevê-se a necessidade de uma componente horizontal na distribuição da intensidade da luz, visto que se pretende iluminar um objeto vertical (pedestres).

Comprovou-se que as luminárias devem ser instaladas na diagonal, nos cantos da passadeira, entre o acesso para peões e o lado de aproximação dos veículos.

2.2.5 The basis of luminance and illuminance recommendations [22]

Dá a conhecer uma breve história dos *standards* para iluminação pública e em que é que estes se baseiam. Não sendo um artigo específico para passadeiras, tem a vantagem de complementar bastante o documento anterior [21] com uma análise mais profunda.

Defende a necessidade de novos *standards* para que os utilizadores se sintam seguros e que se minimize o risco de colisões.

Ilacões:

Trata-se de um artigo que, não sendo específico para passadeiras, permite complementar as informações e conhecimentos necessários para este projeto.

2.3 Técnicas para deteção de peões

Como temos visto até aqui, o conceito de passadeira inteligente está fortemente relacionado com a capacidade de detetar eficazmente a ocorrência de eventos. Estes eventos podem ser a presença de um peão nas proximidades, o excesso de velocidade de um veículo, o anoitecer, entre outros. Há no entanto um evento se destaca dos demais: a travessia de um peão.

Se um sistema não conseguir detetar corretamente as travessias, o produto não funciona. Por outro lado, se existirem falsos positivos, isto levará inevitavelmente à perda de credibilidade e o sistema será ignorado.

A deteção da velocidade também é importante porque permite evitar que os peões se lancem à passadeira sem qualquer cuidado. No entanto, as normas da estrada já dizem que o peão deve aguardar e olhar para ambos os lados antes de atravessar, enquanto que o condutor está mais focado nas múltiplas ações que tem de realizar e não tanto na passadeira.

Deste modo podemos afirmar que a deteção de peões assume uma importância maior na viabilidade da solução, razão pela qual se decidiu aprofundar as alternativas mencionadas em [19] e [20].

Já foi referido que, apesar da hipótese mais simples passar pelos tradicionais botões de pressão, o utilizador tem por hábito ignora-los, surgindo a necessidade de aplicar um método automático e passivo.

2.3.1 Piezoelétrico

Sabendo de antemão que uma das especificações do projeto é a simplicidade de instalação e que este método implica obras no pavimento, podemos desde já excluir esta opção. Ainda assim será abordado por ser muito utilizado em projetos semelhantes.

Estes sensores são pouco ou nada influenciados pelas condições atmosféricas e ambientais.

O seu funcionamento baseia-se no aparecimento de carga elétrica, resultado de uma força mecânica aplicada em certos materiais cristalinos, como representado na figura 2.15. A este fenómeno dá-se o nome de efeito piezoelétrico direto, sendo possível o efeito inverso, por se tratar de um processo reversível.

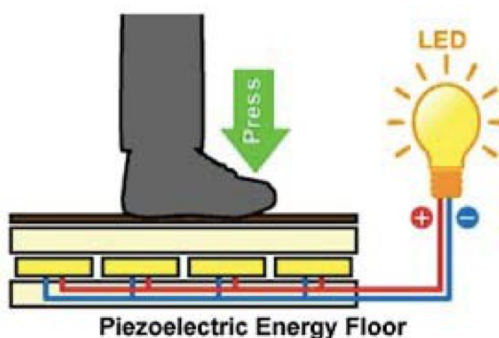


Figura 2.15: Princípio piezoelétrico [23]

Por norma a instalação destes sensores em passadeiras inteligentes é complexa, visto que a localização é crucial. Se o peão não caminhar diretamente pela zona sensível o sistema falhará.

2.3.2 Infravermelhos

Estes dispositivos detetam radiação infravermelha emitida pelos objetos no seu campo de visão. Um exemplo é mostrado na figura 2.16.



Figura 2.16: Exemplo de sensor PIR [24]

A grande vantagem para além do custo reduzido, é que podem ser montados de qualquer ângulo sem que haja perda do sinal. As chuvas também não costumam afetar estes sensores.

O seu desempenho faz com que seja uma solução a ter em conta, mas no entanto se a área de deteção for vasta o número de sensores necessários vai aumentar significativamente, para garantir que não surjam zonas mortas.

Por funcionarem com contrastes de temperatura, se um peão se locomover lentamente e diretamente na sua direção, pode não ser detetado. O maior problema está no entanto associado à manutenção destes equipamentos. Quando se trata de uso no exterior, a luz solar leva a um desgaste muito grande das lentes, precisando de ser substituídas regularmente.

2.3.3 Ultrassons

Os ultrassons possuem a capacidade de emitir ondas com frequências acima dos 20 kHz e que, portanto, os humanos não conseguem ouvir. É medido o tempo que o sinal demora desde o momento em que foi enviado até que retorna (delay) e com esse valor é possível calcular a distância, tendo em conta a velocidade de propagação do som - figura 2.17.

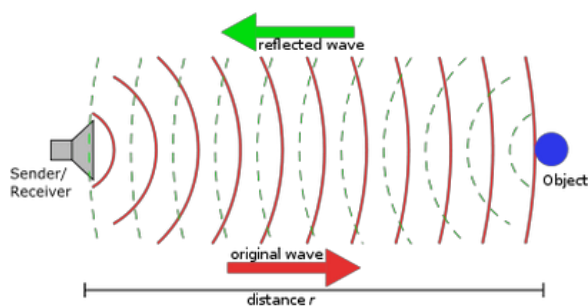


Figura 2.17: Funcionamento de um ultrassons [25]

Por norma são capazes de detetar até uma distância máxima de 10 metros.

As maiores desvantagens para a aplicação pretendida são as seguintes: a zona de deteção é afetada por alterações de temperatura e humidade; roupas feitas de algodão e fibras naturais podem escapar facilmente à deteção.

Nos testes realizados [19] observa-se que as exigências quanto à posição e ângulo dos sensores, a possibilidade de objetos bloquearem a deteção e as flutuações no comportamento devido a fatores externos, reduzem significativamente as garantias de uma deteção correta.

2.3.4 Micro-ondas / Doppler radar

Nestes sensores é enviado um sinal linear modulado em frequência e medidas variações do sinal de retorno, de acordo com o efeito de *Doppler*.

Quando um objeto se aproxima do emissor, as cristas de ondas sucessivas também se aproximam, levando a um aumento da frequência. Da mesma forma, se o objeto se afastar, resulta numa diminuição da frequência observada. Sempre que o recetor regista alguma destas alterações é gerado um alarme.

A principal vantagem destes equipamentos é o seu alcance. De registar também que temperaturas elevadas não afetam o comportamento do sensor.

Como inconveniente, o facto de chuvas muito intensas levarem a falsas deteções e travessias perpendiculares por vezes não serem detetadas.

Nos testes para deteção de peões apresentaram das maiores taxas de não deteção [19].

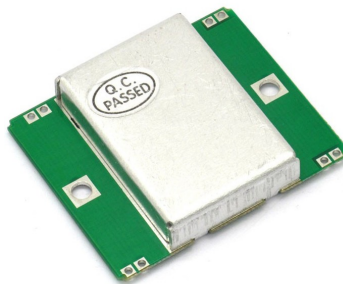


Figura 2.18: Exemplo de sensor de micro-ondas [26]

2.3.5 Câmaras

A tecnologia mais complexa, mas também com maior potencial nos dias de hoje, são as câmaras. Dentro deste tipo de deteção temos por exemplo as câmaras térmicas, que funcionam com base nas diferentes temperaturas do objeto a detetar. Também é possível realizar deteção de objetos apenas por processamento de imagem, por exemplo, recorrendo à classificação de objetos por uma rede neuronal.



Figura 2.19: Exemplo de câmara (Flexidome IP Startlight 7000 VR) [27]

A quantidade de informação que se pode retirar de diferentes cenários é muito superior à dos sensores tradicionais.

No entanto, nem tudo são vantagens. O custo é por norma muito elevado e alguns sistemas são afetados pela iluminação externa e sombras, sem contar com a sujidade das lentes.

2.3.6 Resumo das características

Para especificar o sistema é necessário não só conhecer as tecnologias existentes, mas também compara-las. Só assim é possível optar pela solução mais adequada.

Atendendo à informação recolhida ao longo deste capítulo, sumariou-se na tabela 2.2 um conjunto de características dos métodos de deteção de peões mais utilizados.

Os dados apresentados estão de acordo com a análise levada a cabo no artigo “Passive Pedestrian Detection at Unsignalized Crossings” [19], faltando ressaltar que estas são características genéricas, podendo variar de acordo com o modelo dos sensores, fabricantes e evolução das tecnologias.

⁴Variações de temperatura, humidade, raios solares diretos ou chuvas intensas

Detetor	Alcance	Condições Atmosféricas ⁴	Instalação	Manutenção	Custo
Piezoelétrico	-	Imune	Complexa	Baixa	Elevado
Infravermelhos	Baixo	Afetado	Simples	Elevada	Baixo
Ultrassons	Intermédio	Afetado	Problemática	Baixa	Baixo
Micro-ondas	Elevado	Afetado	Problemática	Baixa	Intermédio
Câmara	Intermédio	Afetado	Simples	Intermédia	Elevado

Tabela 2.2: Breve comparação dos métodos de deteção de peões

Além do piezoelétrico, que já havia sido excluído das alternativas para este projeto, é possível pôr de parte os sensores PIR, principalmente devido à manutenção que requerem (as lentes teriam de ser trocadas no máximo de 5 em 5 anos).

Do mesmo modo, descartam-se as micro-ondas devido a todas as suas limitações, como por exemplo as falsas deteções perante chuvas intensas. Será, no entanto, utilizado um radar para deteção da velocidade de aproximação dos veículos, por possuir um grande alcance e ser uma tecnologia muito desenvolvida para esta aplicação.

Restam os ultrassons e as câmaras como alternativas para deteção de peões. Foram exploradas as duas tecnologias na prática, tendo-se optado por uma tecnologia de deteção de objetos em tempo real chamada You Only Look Once (YOLO). O YOLO recorre a uma rede neuronal para processar imagens que podem ser obtidas por uma câmara de vídeo e a sua implementação será abordada em detalhe no capítulo cinco.

Capítulo 3

Especificações do sistema

Enquadrado o projeto, conhecidas as soluções semelhantes e tendo uma noção das tecnologias mais utilizadas, podemos finalmente dar início ao planeamento do sistema.

O objetivo deste capítulo passa por delinear um plano de ação e estabelecer metas. Isto é especialmente importante no contexto da dissertação, para que se realize trabalho útil sem perder o foco do projeto, cumprindo com os prazos estabelecidos.

Trata-se de uma solução pensada e criada de raiz, não possuindo grande trabalho prévio. As bases de trabalho serão as visitas de campo realizadas às passadeiras de Cascais e Guimarães, já referidas no capítulo anterior, e as conclusões que daí resultaram.

São quatro os tópicos que compõem este capítulo. Inicialmente é dado a conhecer globalmente o sistema. No seguimento são apresentados os requisitos do projeto. O terceiro ponto consiste numa explicação mais detalhada do funcionamento e, por fim, estabelece-se uma lista das vantagens desta solução.

Todas as informações contidas nestas especificações constituem uma visão leiga do projeto, sendo facilmente compreendidas por qualquer pessoa, mesmo sem bases técnicas.

3.1 Descrição inicial

Numa colaboração entre a *Microio*, o Instituto de Telecomunicações (IT), o DETI e o Departamento de Física (DFIS) da Universidade de Aveiro, pretende-se desenvolver uma passadeira de nova geração, que permita delimitar o acesso de peões e veículos com perfis luminosos específicos, aumentando a segurança nestes locais.

Assim surge o sistema STAP que consiste num conjunto de postes comunicantes, com toda a sensorização e eletrónica embutida, facilmente aplicáveis a qualquer passadeira.

Os seus principais objetivos são os seguintes:

- Evitar situações de perigo perto das passadeiras
 - Devido a falta de visibilidade
 - Devido a excessos de velocidade
 - Devido a distrações dos pedestres

- Alertar todos os intervenientes
 - Através de projeção de sinalização na estrada

- Através de sinais sonoros para os invisuais
- Através de um painel informativo com a velocidade instantânea.
- Ser simples e rápido de instalar
 - Utilizando painéis solares
 - Utilizando baterias complementares

A solução está otimizada para uma ou duas faixas de rodagem, podendo estas ser em sentidos opostos ou no mesmo sentido. A instalação de cada poste é realizada nos cantos da passareira, nas bermas da estrada, entre o acesso para os peões e do lado de aproximação dos veículos, como se verifica nas figuras 3.1 e 3.2.

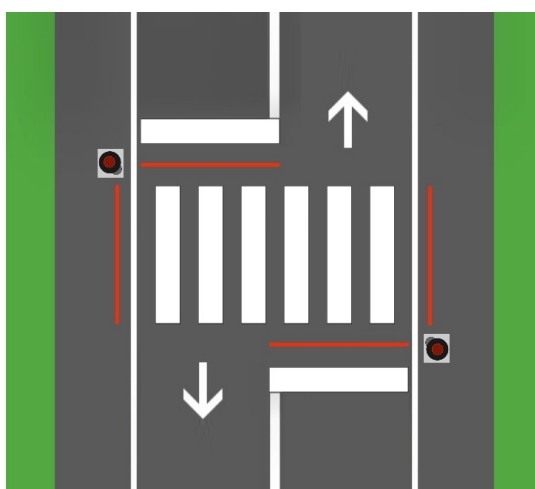


Figura 3.1: Esquema de aplicação STAP para 1 faixa de rodagem

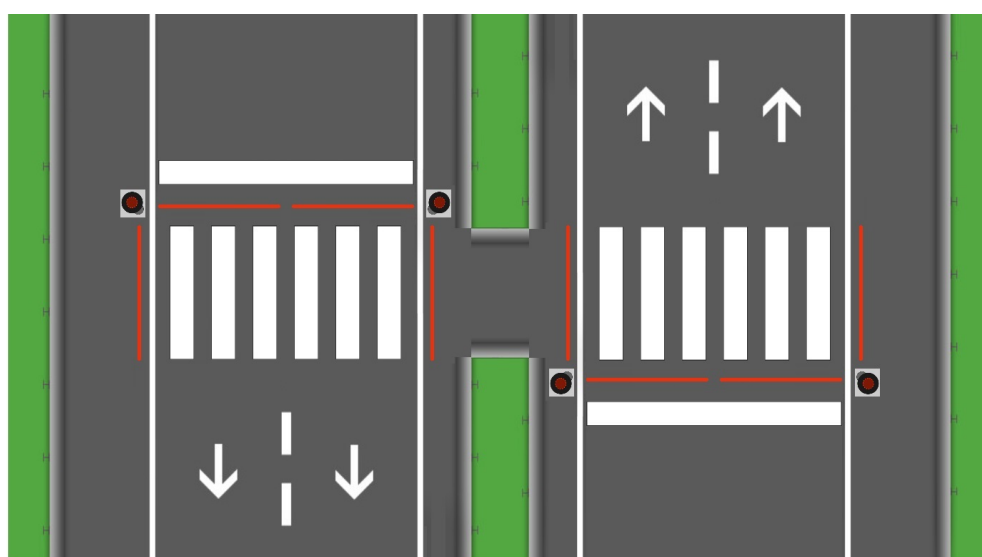


Figura 3.2: Esquema de aplicação STAP para 2 faixas de rodagem

No que à iluminação diz respeito, esta instalação encontra-se conforme o princípio de iluminação assimétrica [20, 21] referido no ponto 2.2.3 deste documento (figura 2.14).

3.2 Requisitos do projeto

Quando se iniciou o projeto, a *Microio*, em parceria com a *F.L.Gaspar*, definiu um conjunto de requisitos gerais que se encontram na tabela 3.1.

Grandeza	Requerimento	Dados específicos
Deteção	- Detetar veículos em movimento - Detetar peões	Capacidade de detetar os diferentes tipos de veículos em trânsito na via e travessias
Velocidade	- Medir velocidade instantânea de veículos - Informar os condutores da sua velocidade	Capacidade de informar o condutor, se este estiver acima do limite de velocidade legal
Sinalização	- No modo diurno ou noturno, iluminar os limites da passadeira	Capacidade de sinalizar veículos que não abrandem e a presença de peões
Iluminação	- No modo noturno, iluminar diretamente a passadeira	Capacidade de iluminar a zona de passagem de peões com um foco retangular

Tabela 3.1: Pré-requisitos do sistema

Para que todos os requisitos sejam cumpridos, a solução deve contemplar a lista de características que agora se define:

- Ser aplicável a passadeiras com ou sem semáforos;
- Detetar a velocidade de aproximação dos veículos através de um radar;
- Detetar a aproximação de peões através de uma câmara;
- Projetar feixes luminosos de diferentes cores que delimitem a passadeira;
- Configurar os perfis dos feixes luminosos mediante os eventos detetados;
- Incorporar um sensor de luminosidade ambiente;
- Possuir um foco de luz branca, configurável em intensidade, mediante a luminosidade;
- Permitir comunicação sem-fios entre os postes;
- Permitir o envio de dados para plataformas de informação e gestão urbana.

3.3 Funcionamento

Considerando os objetivos que se pretende alcançar e os requisitos impostos pelas partes interessadas, foram tomadas as decisões mais adequadas e definido o funcionamento do sistema.

Dividiu-se a explicação em duas partes lógicas: o despoletar de eventos, que diz respeito às tarefas de deteção; e as respostas a esses eventos, que correspondem ao modo de funcionamento e preferências de comportamento do sistema.

3.3.1 Zonas de deteção

O sistema pode alterar o seu comportamento perante um destes acontecimentos:

- Diminuição da luminosidade ambiente;
- Presença de um peão que pretenda atravessar;
- Excesso de velocidade por parte de um veículo em aproximação;

Relativamente à luminosidade, será utilizado um pequeno sensor, instalado no topo de cada poste para calcular a intensidade luminosa, cuja unidade é o *lux* (*lx*).

O *lux* indica o fluxo luminoso que uma determinada fonte de luz incide sobre uma superfície, a uma certa distância da mesma, correspondendo a 1 *lúmen* incidido perpendicularmente numa área de 1 metro quadrado (lm/m^2) [28]. O esquema da figura 3.3 ajuda a compreender melhor o conceito.

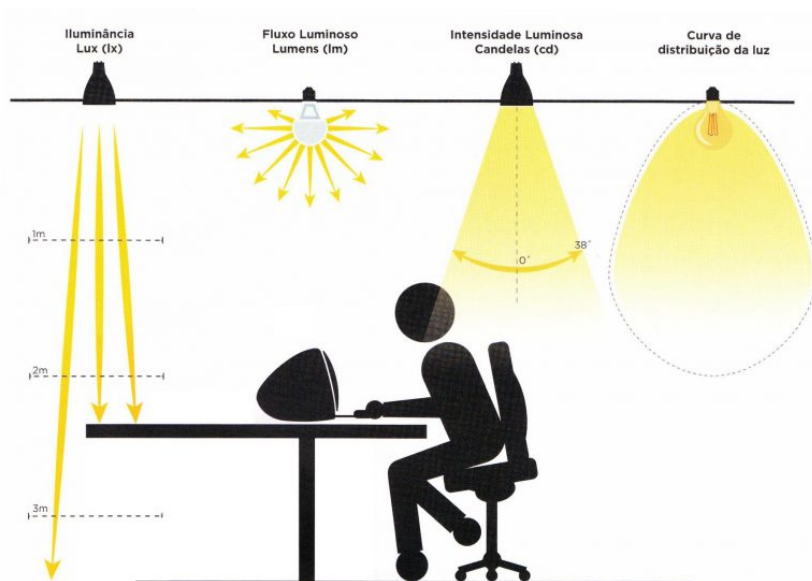


Figura 3.3: Diferença entre *lux*, *lúmen* e *candela* [28]

Sabendo a intensidade da iluminação ambiente, podemos definir um valor mínimo a partir do qual as luminárias acendem, para compensar o anoitecer ou situações de fraca visibilidade.

A tabela 3.2 permite comparar um conjunto de valores de intensidade luminosa conhecidos.

Perante estes dados, decidiu-se utilizar como mínimo necessário para ativação das luminárias 400 lx. Se os valores medidos pelo sensor forem inferiores, os focos de luz LED no topo do sistema ligam de forma progressiva para garantir uma boa visibilidade, ou seja, quanto menor o valor lido, maior a intensidade luminosa fornecida pelos LEDs.

Em situações normais, a luminária permanecerá acesa desde o pôr até ao nascer do sol, bem como em dias muito nublados.

Iluminância (lx)	Fonte de luz
0,0001 - 0,0004	Noite de céu nublado - luz das estrelas
0,002	Noite de céu claro - brilho do céu
0,05 - 0,30	Noite clara e de lua cheia
3,0 - 3,4	Limite mais escuro do crepúsculo civil ¹
100	Dia muito nublado e escuro
400	Nascer ou pôr do sol
1.000	Dia ligeiramente nublado
10.000 - 25.000	Luz em dia claro, sem luz direta do sol
32.000 - 130.000	Luz solar direta

Tabela 3.2: Valores médios de iluminância (lux) [29]

Falando agora sobre a detecção de peões e de velocidade, estas requerem um posicionamento específico para que o sistema funcione corretamente. Uma representação das áreas de detecção aconselhadas é exibida nos esquemas da figura 3.4, onde a região mais estreita é a detecção de peões, direcionada às entradas da passadeira e a região mais larga é a detecção de velocidade, direcionada para a via e no sentido de aproximação dos veículos.

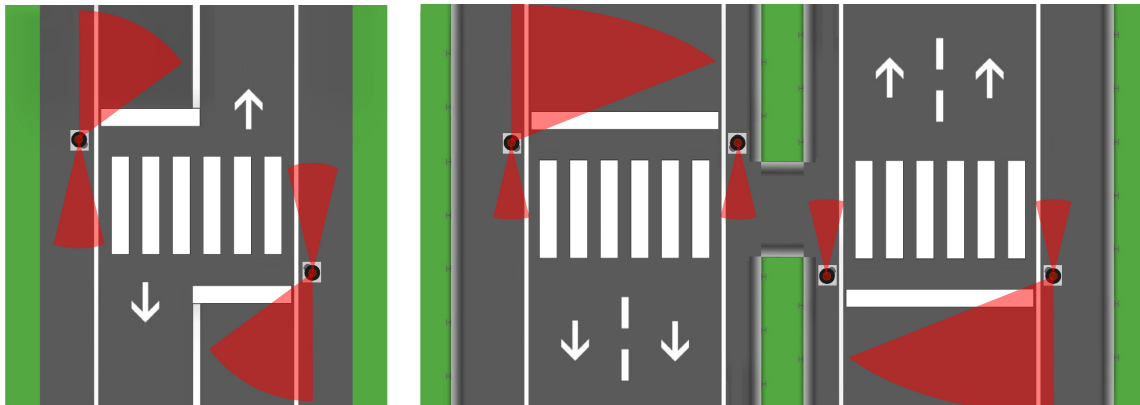


Figura 3.4: Zonas de detecção STAP para 1 e 2 faixas de rodagem

Como foi referido anteriormente, a detecção de peões será feita em tempo real através de uma pequena câmara. As imagens obtidas serão processadas por uma rede neuronal, treinada especificamente para detetar apenas pessoas.

Para definirmos concretamente a zona de detecção dos peões, precisamos de mais informações à cerca das passadeiras do tipo “zebra”. Estas passadeiras devem ser utilizadas apenas quando se puder garantir velocidades inferiores a 50 km/h e suas principais características físicas são as seguintes [30]:

- A largura normal da passagem é de 4m;

¹Horário que divide a noite do dia, onde o sol se encontra a menos de 6° da linha do horizonte, contribuindo com iluminação indireta por difusão e reflexão

- Em casos particulares, a largura pode baixar para valores entre 2,5 e 3m²;
- A linha de cedência de passagem deve estar distanciada entre 1,5 e 2m da passadeira;
- O sinal vertical de passagem de peões deve ser colocado junto à linha de cedência;
- Se o comprimento for superior a 10m deve ser considerado um refúgio central.

A figura 3.5 permite uma confirmação visual das informações descritas anteriormente.

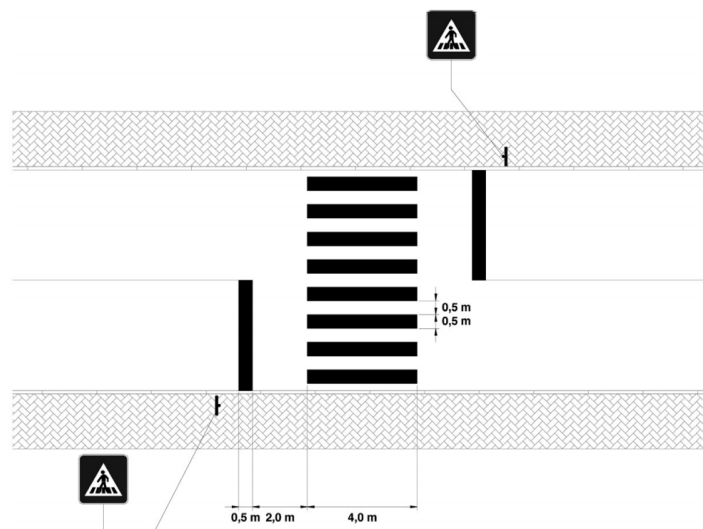


Figura 3.5: Esquema de passadeira do tipo “zebra” [30]

Definiu-se que os postes serão colocados entre a linha de cedência e a passadeira, ficando a 1m da zona de passagem.

O alcance mínimo para a deteção de peões deve ser então de 6m, garantindo que estão contemplados na zona de deteção os 4m da largura normal das passadeiras e 1m adicional para cada um dos lados.

Não é definido um limite máximo para a zona de deteção de peões, já que este valor é normalmente limitado pela câmara e pela capacidade da rede neuronal em detetar objetos muito distantes. No entanto, não é aconselhável uma deteção a grandes distâncias.

A imagem obtida pela câmara não deve ser muito ampla, para que não sejam detetados ciclistas ou outras pessoas que passem na faixa rodoviária e não pretendam atravessar.

Por fim, no que diz respeito à deteção de velocidade, foi previamente definido que a velocidade instantânea seria obtida a uma distância de pelo menos 150m da passadeira. Para percebermos se este valor é adequado, precisamos garantir que a distância de paragem é suficiente.

A distância de paragem (d_p) é composta pela soma das distâncias de reação (d_r) e de travagem (d_t).

Distância de reação [31]:

A distância de reação é a distância percorrida pelo veículo durante tempo de reação (t_r) do condutor e este tempo é o intervalo que decorre entre a perceção de um estímulo externo,

²Velocidades dos veículos inferiores a 50 km/h, fluxo de peões baixo ou restrições físicas inultrapassáveis

até ao instante em que o condutor atua o travão. Está relacionado com as características e condições físicas e psicológicas do mesmo, tais como: idade, presença de álcool no sangue, efeitos de medicamentos ou drogas, fadiga, concentração e estado emocional.

Aproximando o comportamento do veículo durante este intervalo de tempo, a um movimento retilíneo uniforme, obtêm-se a expressão 3.1:

$$d_r = v \times t_r \quad (3.1)$$

O gráfico da figura 3.6 permite verificar o impacto da velocidade e do tempo de reação, na distância percorrida pelo veículo.

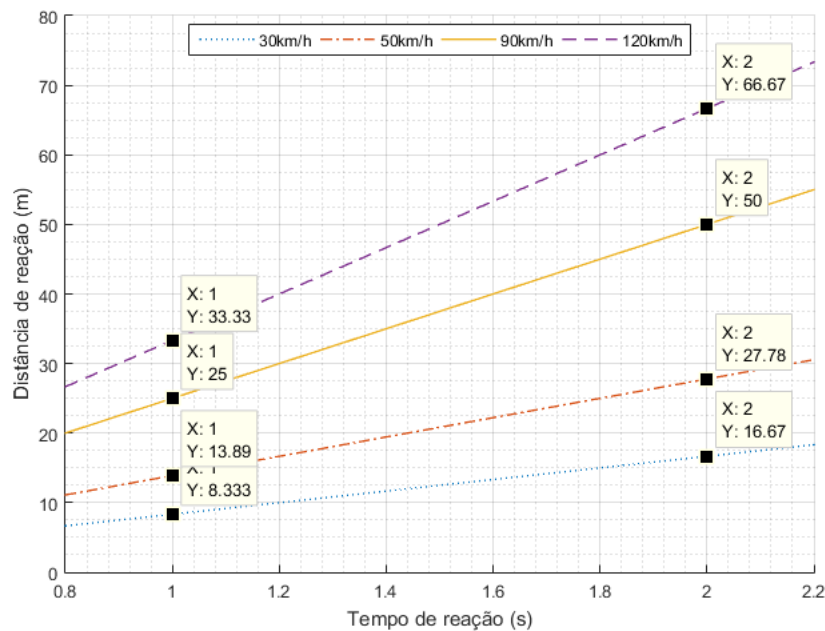


Figura 3.6: Distância de reação em função do tempo, para diferentes velocidades

Considera-se que o tempo de reação normal de um condutor está em torno de 1s, correspondente no gráfico aos pontos (X=1). Por uma questão de segurança e para efeitos de cálculo, será também considerado um pior cenário, onde o tempo de reação é 2s (X=2).

Distância de travagem [32]:

Por sua vez, a distância de travagem é a distância percorrida desde o momento em que o travão é atuado, até que o veículo se encontre imóvel.

Para o cálculo desta distância é necessário considerar a variação da energia cinética (E_c) resultante da energia dissipada pelas forças de atrito (E_d), durante o tempo de travagem (t_t). Deste modo, e aproximando a travagem a um movimento retilíneo uniformemente retardado, temos que:

$$\Delta E_c = E_d \quad (3.2)$$

$$\frac{1}{2} \times m \times v_i^2 - \frac{1}{2} \times m \times v_f^2 = \mu \times m \times g \times d_t$$

$$d_t = \frac{v_i^2 - v_f^2}{2 \times \mu \times g} \quad (3.3)$$

- v_i é a velocidade inicial
- v_f é a velocidade final
- μ é o coeficiente de atrito ($\mu \simeq 0,8$ pavimento seco; $\mu \simeq 0,4$ pavimento molhado)
- g é a aceleração gravítica ($g \simeq 9,81m/s^2$)

Considerando que o veículo pretende parar completamente para ceder passagem a um peão, temos $v_f = 0$ e a equação 3.3 passa a ser dada por:

$$d_t = \frac{v_i^2}{2 \times \mu \times g} \quad (3.4)$$

O gráfico da figura 3.7 permite verificar o impacto da velocidade na distância de travagem, consoante as condições do pavimento, supondo que a estrada não possui inclinação.

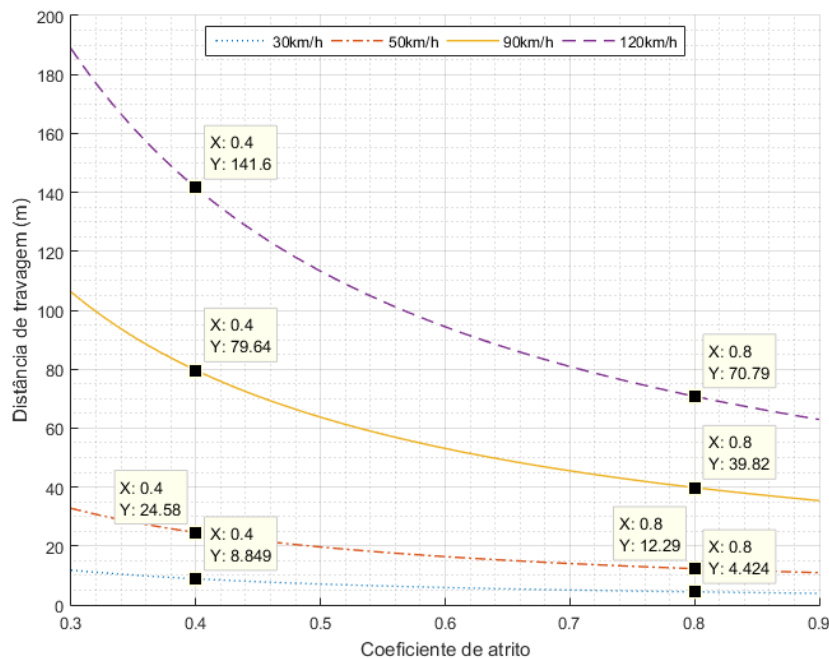


Figura 3.7: Distância de travagem em função do atrito, para diferentes velocidades

Considerou-se que um coeficiente de atrito em torno de 0,8 corresponde a um pavimento seco, enquanto que valores na ordem de 0,4 dizem respeito a um pavimento molhado.

Ressaltar que este coeficiente, por depender das superfícies de contacto, varia bastante com o tipo de viatura em questão, com o estado do pavimento, entre outros fatores. O estado da viatura em si (suspensão, travões, estado dos pneus, distribuição do peso) também afeta significativamente a distância de travagem.

30km/h		Pavimento (μ)		50km/h		Pavimento (μ)	
		seco	molhado			seco	molhado
Reação (t_r)	normal	12,8m	17,2m	Reação (t_r)	normal	26,2m	38,5m
	lenta	21,1m	25,5m		lenta	40,1m	52,4m

90km/h		Pavimento (μ)		120km/h		Pavimento (μ)	
		seco	molhado			seco	molhado
Reação (t_r)	normal	64,8m	104,6m	Reação (t_r)	normal	104,1m	174,9m
	lenta	89,8m	129,6m		lenta	137,5m	208,3m

Tabela 3.3: Distância média necessária para travar um veículo

Distância de paragem:

Podemos finalmente verificar se um alcance de 150m é adequado para o radar. Para isso recorreremos aos dados da tabela 3.3, onde se sumaria a informação dos gráficos anteriores, apresentando uma estimativa da distância percorrida pelos veículos, perante diferentes cenários. Esta distância foi calculada segundo a expressão $d_p = d_r + d_t$.

Conclui-se que acionar um alerta direcionado aos veículos quando estes se encontram a uma distância de 150m é suficiente na maioria dos casos, sendo que, de entre as velocidades analisadas, apenas no caso em que um veículo se desloca num pavimento molhado a 120 km/h não conseguiria travar a tempo.

Como vimos anteriormente, estas passadeiras destinam-se, por norma, a estradas onde a velocidade recomendada é de 50 km/h ou inferior, garantindo uma boa margem de manobra.

Ainda assim, sempre que a estrada onde é feita a instalação o permitir, é possível aumentar o alcance do radar, sendo os 150m um valor mínimo recomendado.

Não nos podemos esquecer que se por alguma razão os alertas luminosos não estiverem na linha de vista dos condutores, seja por existirem obstáculos à visibilidade, por distração do condutor, ou qualquer outra razão, estes podem ser insuficientes, devendo condutores e peões ter sempre o máximo de precaução, em especial junto às passadeiras e cumprir as regras em vigor no código da estrada.

Este é um sistema auxiliar e não obrigacional.

3.3.2 Modos de operação

Sabendo como é que o sistema é acionado, precisamos definir como se vai comportar perante esses estímulos. Dão-se assim a conhecer as opções tomadas durante o planeamento da solução e que explicam o seu funcionamento.

Percebeu-se a necessidade de manter um comportamento simples e intuitivo, até porque, durante a pesquisa de soluções similares foram encontradas alternativas que não singraram por serem de difícil interpretação.

A tabela 3.4 resume o funcionamento geral do sistema. Apesar das entradas da tabela estarem todas correlacionadas, é possível dividi-la em duas partes para compreender melhor o seu significado. Do lado esquerdo existem duas colunas intituladas detetor e atuador que dizem respeito à iluminação ambiente, cujo objetivo é permitir uma boa visibilidade junto à passadeira. Do lado direito, temos outras duas colunas com os mesmos nomes, referentes aos alertas emitidos pelo sistema através da projeção de sinalização luminosa.

Detetor	Atuador	Detetor		Atuador	
<i>Luminosidade</i>	<i>Luminárias LED</i>	<i>Velocidade</i>	<i>Peão</i>	<i>Sinalização veículos</i>	<i>Sinalização peões</i>
Boa visibilidade ($> 400lx$)	Desligado	Segura	Ausente	Desligado	Desligado
		Excessiva	-	Intermitente	Ligado
		-	Presente	Ligado	Desligado
Visibilidade reduzida	Ligado com baixa intensidade ($\leq 50\%$)	Segura	Ausente	Desligado	Desligado
		Excessiva	-	Intermitente	Ligado
	Ligado	-	Presente	Ligado	Desligado

Tabela 3.4: Resumo de funcionamento STAP

Iluminação de presença:

Já foi referido que, ao anoitecer, as luminárias se ativam automaticamente e aumentam progressivamente a sua intensidade, em função da intensidade luminosa medida pelo sensor. Porém, valores ambiente próximos de $0lx$ não vão corresponder à intensidade máxima das luminárias, mas sim a 50%. Desta forma, é possível não só reduzir o consumo energético, como também implementar uma medida de segurança adicional.

A luminária escolhida deverá iluminar razoavelmente a zona envolvente à passadeira, apenas com metade do consumo, permitindo que não haja um grande desperdício de energia em locais remotos, ou em horas de pouco fluxo de tráfego - figura 3.8.

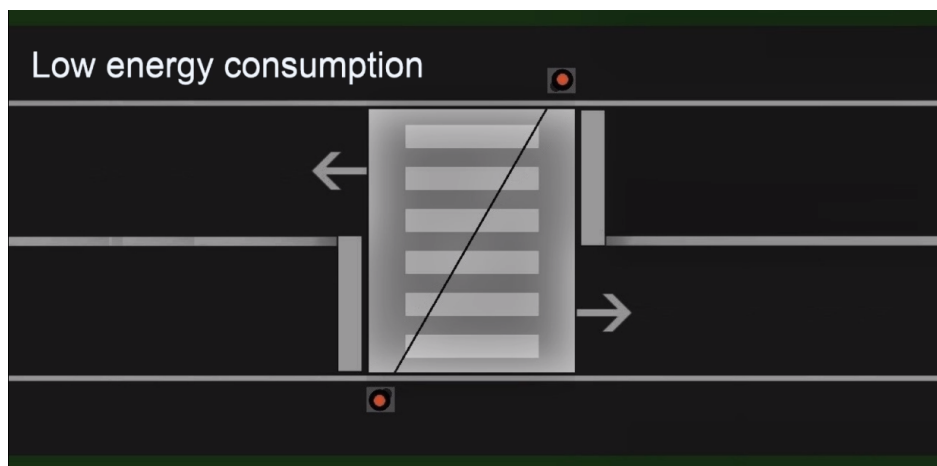


Figura 3.8: Modo de poupança de energia STAP

No entanto, sempre que um peão for detetado durante a noite, a luz passará de imediato para 100% da sua intensidade, captando a atenção de todos os intervenientes e permanecendo assim até que o peão acabe de atravessar - figura 3.9.

Posteriormente, a intensidade volta a descer gradualmente até ao ponto onde estava, ou seja, regressa aquilo que podemos designar como o modo de poupança de energia do sistema.

Sinalização luminosa:

No que diz respeito aos alertas emitidos pelo sistema, podem ocorrer três situações distintas, independentemente das condições de visibilidade. Basta-nos então analisar estas

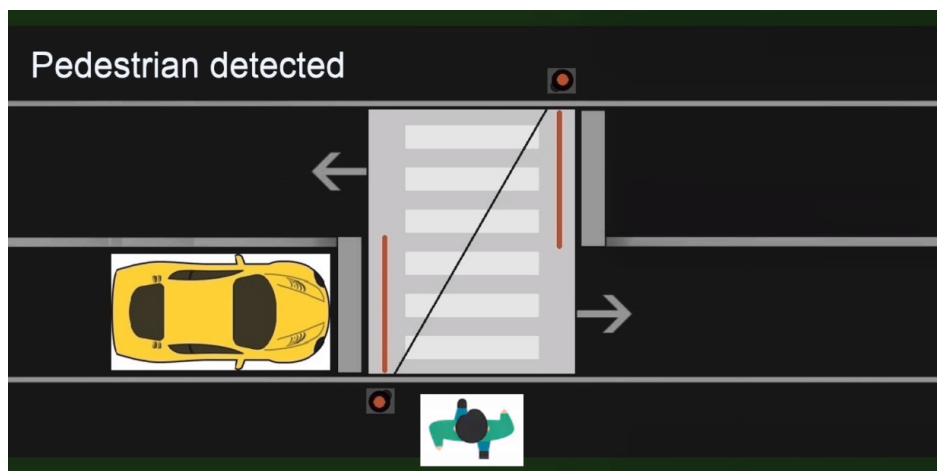


Figura 3.9: Representação da detecção de um peão durante a noite

ocorrências para um dos casos e o raciocínio será igual para o outro.

Considerando um dia com boa visibilidade e uma estrada cujo limite de velocidade é de 50 km/h, se um veículo se aproximar de uma passareira com o sistema STAP, a uma velocidade inferior ao limite estabelecido por lei, nada acontece - figura 3.10.

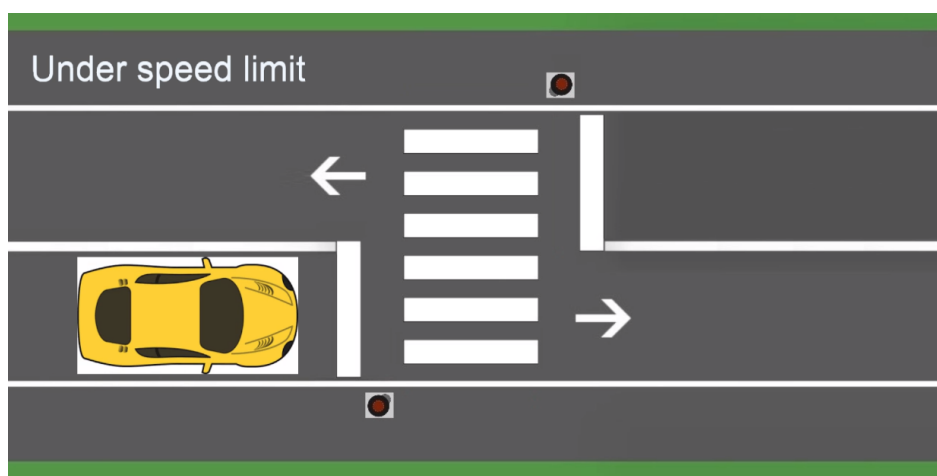


Figura 3.10: Representação da detecção um veículo dentro do limite de velocidade

Se a aproximação foi feita a mais de 50 km/h, estamos perante uma situação de potencial perigo para peões que pretendam atravessar a via. Assim, é acionada uma luz direcionada aos peões, para que não atravessem sem tomar precauções. Ao mesmo tempo, surge uma luz intermitente direcionada ao veículo, para que este tome consciência de que deve reduzir a velocidade - figura 3.11.

Um último caso ocorre quando um peão é detetado. Como sabemos, os peões tem prioridade sobre os veículos nas passareiras, pelo que nestes casos é ativada uma luz direcionada aos condutores, informando-os que devem parar. Uma representação desta situação pode ser revisitada na figura 3.9, já que a sinalização aplicada é a mesma, sendo a única diferença a iluminação de presença.

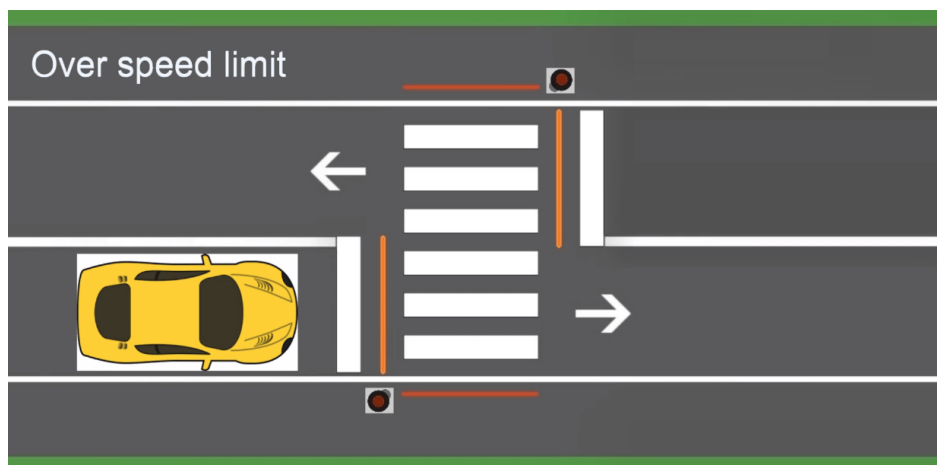


Figura 3.11: Representação da deteção um veículo em excesso de velocidade

É possível que um excesso de velocidade e a deteção de um peão ocorram quase em simultâneo. Este caso não é descrito nesta análise, por se tratar duma situação que vai sempre cair numa das duas últimas referidas:

- Se a deteção do peão acontecer ligeiramente antes do que o excesso de velocidade, mantém-se a prioridade do peão e a ordem de paragem para o veículo.
- Se o excesso de velocidade for detetado ligeiramente antes do que o peão, a luz acionada para que os peões não se lancem à estrada sem as devidas precauções, toma uma importância ainda maior, permitindo que o pedestre aguarde em segurança pela passagem ou abrandamento do veículo, como se verifica na figura 3.12.

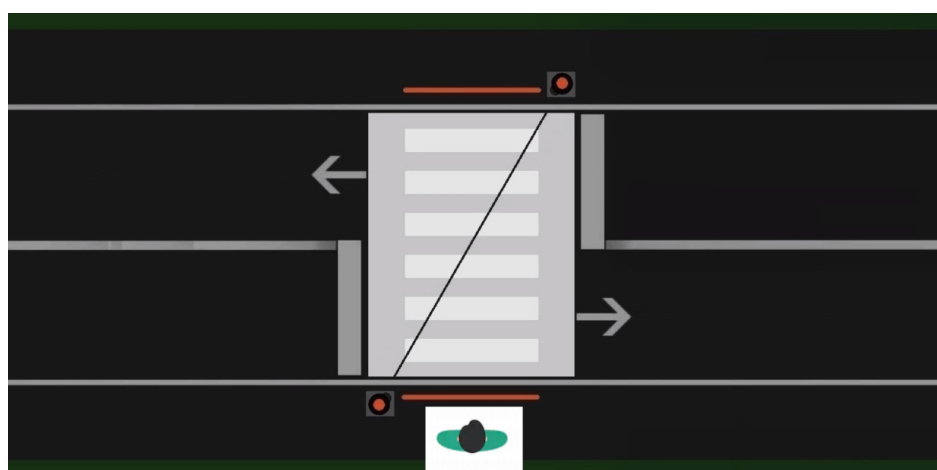


Figura 3.12: Representação da deteção um excesso de velocidade seguido de um peão

Houve o cuidado de não utilizar a cor verde, que no contexto rodoviário é normalmente associada à autorização de passagem, em nenhum dos perfis luminosos. Esta opção surge na medida em que a solução não pretende ser um regulador de tráfego, mas sim um sistema complementar à segurança rodoviária, que apenas emite alertas.

3.4 Vantagens

O sistema STAP apresenta um conjunto de fatores diferenciadores em relação à maioria das passadeiras inteligentes.

Controlo de velocidade: Permite através de um radar, alertar não só veículos mas também os peões.

Combater os “smombies” e veículos elétricos silenciosos: Através da projeção de sinalização LED no chão, é possível colocar os avisos no campo de visão das pessoas que se deslocam distraídas com os seus telemóveis, ou que simplesmente não se conseguem aperceber do aproximar de um veículo.

Não necessita de intervenções significativas na via pública: Os dois postes contêm toda a eletrónica, desde os sensores até aos atuadores, pelo que não é necessário embutir luzes no pavimento.

Troca de informações de trânsito: Em linha com o conceito de *smart cities*, o sistema permite conexão a uma plataforma com uma base de dados, para registar informações e detetar padrões.

Otimização dos tempos de passagem: De acordo com o fluxo de pessoas diário, semanal, ou mensal, é possível prever comportamentos e ajustar os tempos durante os quais determinada sinalização é atuada. Permite também ajustar o modo de poupança de energia, diminuindo ainda mais o consumo.

Aplicável a qualquer tipo de passadeira: É um sistema flexível, aplicável a passadeiras com ou sem semaforização, dentro ou fora das localidades, ideal para escolas e zonas com um grande fluxo de pedestres. Adicionando painéis solares e baterias é possível instalar esta solução em passadeiras temporárias.

Conhecido o projeto e fechadas as especificações, podemos prosseguir com a implementação do sistema na prática. Para isso foi desenvolvido um protótipo em tamanho real, cujo desenvolvimento será analisado em detalhe no próximo capítulo.

Capítulo 4

Implementação

Neste capítulo retratam-se todas as etapas da concepção de um protótipo que visa comprovar a viabilidade do projeto. No contexto da dissertação, este segmento corresponde à parte mais técnica, procurando não só explicar os procedimentos, como também dar algum suporte a futuras instalações.

Começamos então por falar sobre a arquitetura do protótipo, dando a conhecer como é que este foi construído, os materiais que foram utilizados, como foi programado e outros detalhes de implementação.

Segue-se um segundo tópico referente à configuração e utilização do radar para recolha de informações de trânsito.

4.1 Arquitetura do protótipo

O objetivo nesta fase passa por conseguir implementar um dos postes com todos os subsistemas abordados até então. Para que isso fosse possível, seguiram-se os seguintes passos:

1. Pensar numa maneira simples e eficaz de construir a estrutura física;
2. Criar uma lista com todo o material necessário;
3. Definir um fluxograma funcional para auxiliar à programação;
4. Desenvolver a eletrónica necessária;
5. Integrar os componentes e eletrónica na estrutura.

Descreve-se agora cada uma destas etapas.

4.1.1 Estrutura

O primeiro grande desafio que se impôs neste projeto foi a necessidade de criar um protótipo que pudesse ser transportado, já que à partida não existia um local específico para implementar o sistema.

Perante esta limitação, surgiu a ideia de construir um poste de Policloreto de Vinilo (PVC), com uma base móvel que permitisse facilmente mudar a sua posição. Deveria também permitir uma desmontagem parcial para agilizar o transporte.

Rapidamente se percebeu que um sistema móvel não traria grandes vantagens, uma vez que a partir do momento em que o protótipo é colocado a funcionar, não precisamos mais

de o deslocar. Pelo contrário, poderia inclusive trazer algumas desvantagens. Devido ao seu comprimento, peso e outras características físicas, uma base com rodas poderia levar ao desequilíbrio da solução.

Decidiu-se então optar pela construção de uma base sólida que permitisse encaixar e segurar o poste sem comprometer a estabilidade. Uma primeira versão, feita a partir de uma lata metálica, betão e tubo de PVC de 90mm é apresentada na figura 4.1.



Figura 4.1: Primeira versão da base de suporte do protótipo

No que à restante estrutura diz respeito, o PVC revelou-se uma aposta acertada. Este material permitiu, com relativa facilidade, fazer todos os furos necessários para aplicar os sistemas elétricos e eletrónicos, sem comprometer a sua resistência global.

Inicialmente tentou-se dividir o poste em duas partes, o que permitiria trabalhar com apenas uma parte do poste de cada vez, como se vê na figura 4.2.



Figura 4.2: Primeira versão da estrutura do protótipo (separada)

As duas metades eram depois encaixadas com recurso a um *o-ring*¹. Esta abordagem, porém, foi de imediato posta em causa perante a instabilidade criada no topo do sistema, quando se uniam as duas partes.

A estrutura foi então repensada e decidiu-se utilizar um tubo único com 3m de altura.

Um novo entrave surgiu pouco tempo depois, quando se verificou que o peso da luminária escolhida levava a uma pequena inclinação da estrutura. Um preciosismo, mas que aliado à dificuldade que seria alimentar todos os sistemas no interior de um tubo com apenas 90mm, levou à reconstrução do protótipo por completo.

Foi então desenvolvida uma terceira e última versão considerando esses detalhes e que pode ser observada na figura 4.3.



Figura 4.3: Estrutura definitiva do protótipo

Reconstruiu-se a base, aumentando a superfície de contacto com o solo, o peso e garantindo um nivelamento adequado. Assim foi possível assegurar que a estrutura aguenta ventos e chuvas fortes sem tombar.

Substituiu-se o tubo de PVC de 90mm por um tubo de 110mm de diâmetro e também com uma espessura maior, para conseguir passar melhor os cabos, reduzir a flexibilidade do material e eliminar a inclinação provocada pela luminária.

¹ junta flexível em formato de anel, criada por Neils Christensen [33]

Foram ainda realizados ajustes na forma como a base se interliga com o PVC.

Ficou assim concluída a parte mecânica do protótipo, cujo *design* permite que o sistema seja desmontado e transportado facilmente, graças à sua base removível. Também é possível remover a luminária e o tubo de 50mm que cruza o tubo principal.

4.1.2 Componentes

As tecnologias que escolhemos fazem toda a diferença no desenvolvimento de uma solução, podendo conduzi-la ao sucesso ou condená-la ao fracasso.

Os principais componentes, necessários para o desenvolvimento deste projeto, encontram-se representados no diagrama de blocos da figura 4.4.

Optou-se por implementar o sistema completo que corresponde a dois postes comunicantes, em vez de apenas um como era previsto. Isto permite que posteriormente sejam realizados testes muito mais completos, no entanto, passa a ser necessário o dobro do material. A única exceção será o radar, por uma questão de contenção de custos.

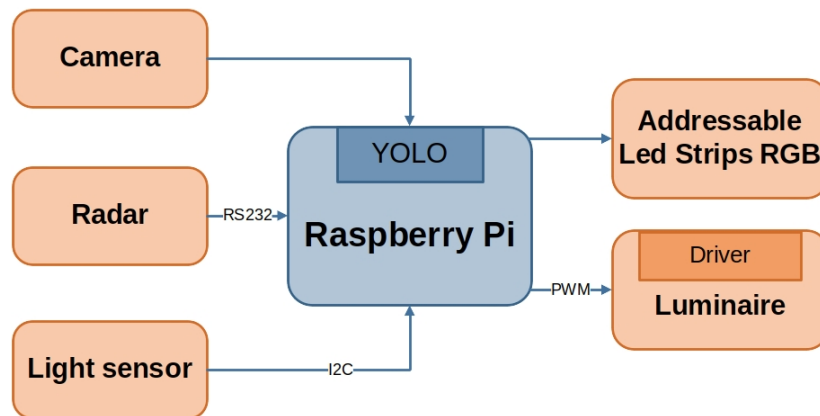


Figura 4.4: Diagrama de blocos de um poste

O processamento vai ser realizado por uma *Raspberry Pi 3B+*, por se tratar de uma placa muito versátil que permite colocar o sistema a funcionar rapidamente e pode facilmente ser integrada com outras placas.

É importante notar que o poder de processamento desta placa poderá vir a ser um fator limitativo, tendo em conta a rede neuronal que se pretende implementar. Na fase de prototipagem isto não deverá ser um problema, porque estamos cientes desta limitação. No entanto, se o objetivo for converter a solução num produto, será necessário um *upgrade* da tecnologia, recorrendo por exemplo a uma *Raspberry Pi 4*, ou outra placa com melhor Central Processing Unit (CPU) e mais memória.

Agora que definimos onde será tratada a informação, focamos as atenções nos componentes que correspondem às entradas e saídas do diagrama anterior.

Câmara:

Por uma questão de simplicidade, optou-se por iniciar os testes com uma câmara da própria *Raspberry* - figura 4.5.

A ideia seria substituí-la posteriormente se os resultados não fossem os melhores. Porém, a resposta deste módulo foi bastante boa, produzindo taxas de deteção promissoras com um alcance dentro do expectável, pelo que se decidiu manter a escolha.

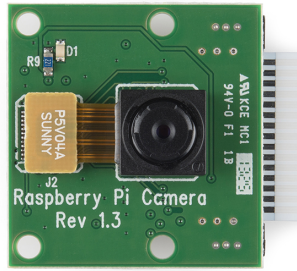


Figura 4.5: Módulo câmara para Raspberry Pi v1.3

A utilização desta câmara traz também a vantagem de uma conexão direta com placa de processamento, permitindo inclusive fazer uso de uma biblioteca específica em *python*.

Este modelo possui 5 *megapixels*, pesa 3 gramas e mede 25x24x9mm. É capaz de gerar fotografias até 2592x1944 *pixels* e vídeos com resolução de até 1080p² [34].

Radar:

Desde o início do projeto que se previa que este fosse o componente mais caro e um dos mais importantes, pelo que foi necessário realizar um estudo do mercado, em busca da alternativa mais adequada.

As características que pretendemos para o radar são as seguintes:

- Medição de velocidades até 120 km/h;
- Detecção a uma distância mínima de 150m;
- Interface RS232 ou RS485;
- Instalação fixa;
- Encapsulado.

Encontraram-se diversas opções, cujas especificações se encontram resumidas na tabela 4.1.

Empresa	Alcance (m)	Gama (km/h)	Frequência (GHz)	Interfaces	Encapsulado	Detecção contramão	Detecção paragem	Software dedicado
AMG	150	5-255	24,15-24,25	RS232, RS485, UART	×	×	✓	×
GMH	300	0,8-480	35,40-35,60	Relé	✓	×	×	×
Stalker	-	1,6-321	34,60-34,80	RS232, RS485	✓	×	×	✓
Capsys	60, 100, 150, 180	1-200	24,125	RS232, RS485, 2 Relés	✓	✓	×	×
Stalker II	400	1-321	24,00-24,20	RS232, RS485, USB, Relé	✓	✓	×	✓
Siemens	150	1-200	24,05-24,25	CAN, Ethernet, RS485, 8 Relés	✓	✓	✓	✓

Tabela 4.1: Alternativas de radar para deteção de velocidade

²resolução horizontal de 1920 *pixels*, vertical de 1080 e varredura progressiva

As primeiras três acabaram por ser descartadas, por não terem todas as características pretendidas. O radar da *AMG* [35] por não ser encapsulado, o da *GMH* [36] devido ao interface limitado e o da *Stalker* [37] por não apresentar informações que garantissem o alcance desejado. Além disso, nenhuma destas soluções permitia a deteção nos dois sentidos, algo que não sendo um requisito, pode vir a ser útil no futuro.

Estudaram-se as três alternativas restantes com maior detalhe. Foram pedidos orçamentos e mais informações às respetivas empresas, tendo-se concluído que embora o radar da *Siemens* [38] fosse o que apresentava as melhores características, era demasiado dispendioso para a aplicação pretendida.

Sobraram então duas opções, comparadas em detalhe na tabela 4.2.

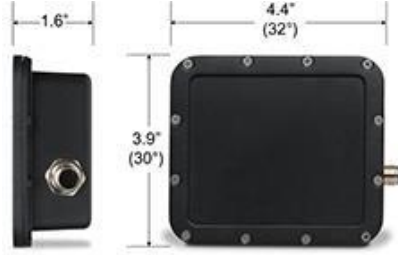

	Stalker Stationary Speed Sensor II	Capsys RXC Series
		
Custo	Kit desenvolvimento: 705€ + Portes Custo de cada sensor adicional: 440€	Uma unidade: 749€ Custo de cada sensor adicional: 485€
Resolução	0,16km/h Precisão: 0,8km ± 0,4%	16 limites de velocidade ajustáveis
Alimentação	10 a 45V DC	24/230V AC 12/24V DC (não <i>standard</i>)
Saídas	RS485 <i>full duplex</i> ; RS485 <i>half-duplex</i> ; RS232 sem controlo de fluxo.	Relé e opto-acoplador; RS485 (consultando o fornecedor); RS232 (Sem a saída de relé).
Extras	O <i>kit</i> de desenvolvimento inclui: cabo RS232, adaptador RS232/USB, software de configuração e teste, suporte para montagem do sensor, <i>calibration fork</i> e manuais.	Comando de infravermelhos opcional para configurações à distância.
Comentários adicionais	Dispõem de 11 protocolos de <i>streaming</i> de dados; O interface gráfico permite atualizar o <i>firmware</i> .	O modo de 1 relé deteta apenas velocidade superior ou inferior; O modo de 2 relés deteta a passagem e velocidades excessivas; Montagem simples e posição ajustável.

Tabela 4.2: Comparação complementar entre os radares *Capsys* e *Stalker*

Os dois radares cumprem com os requisitos do projeto e os preços também são bastante semelhantes. No entanto, a solução da *Stalker* [39] inclui um *kit* de desenvolvimento completo que pode ser integrado de diversas formas e com múltiplas tecnologias, fornecendo dados estatísticos do tráfego. Já o radar da *Capsys* [40], sendo muito mais limitado, tem a vantagem de ser extremamente mais simples.

Sendo este um projeto de engenharia e perspetivando a possibilidade de se desenvolverem

novos projetos no âmbito rodoviário, a escolha mais lógica acabou por ser o *Stalker Stationary Speed Sensor II*, como se verifica na figura 4.6.

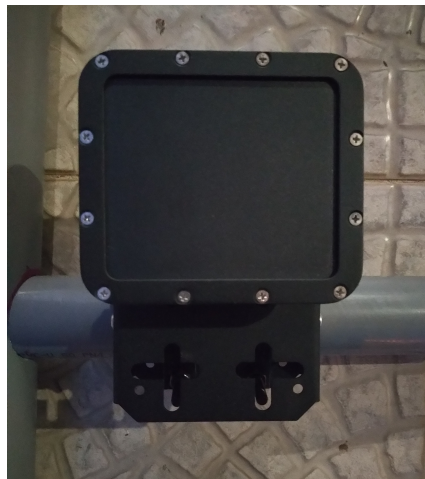


Figura 4.6: Radar Stalker Stationary Speed Sensor II

Sensor de luminosidade:

Numa fase inicial, a *Microio* forneceu um sensor de luminosidade da *Adafruit* designado TSL2591 (ver figura 4.7), que havia sido utilizado no âmbito de outro projeto. O objetivo seria dar os primeiros passos no desenvolvimento da solução, começando por controlar a intensidade de alguns LEDs consoante a luminosidade.



Figura 4.7: Sensor de luminosidade TSL2591

Trata-se de um dispositivo que utiliza Analog-to-Digital Converter (ADC)s para converter, simultaneamente, a luz recebida por dois fotodiodos em sinais digitais. A comunicação é posteriormente efetuada com recurso ao protocolo Inter-Integrated Circuit (I2C) [41].

Este sensor permite ler valores entre os $118\mu lx$ e os $88.000 lx$, mais do que suficiente no âmbito do projeto, pelo que acabou por ser implementado no protótipo.

Tiras de LED:

O aluno do DFIS, responsável pela projeção da sinalização na via, não cumpriu com o desenvolvimento da sua parte da solução.

Como tal, ao contrário do que tinha sido previsto, tomou-se a decisão de embutir tiras de LED nos postes que permitissem confirmar visualmente o funcionamento do sistema.

Escolheram-se LEDs Red, Green, Blue (RGB)³ porque são necessárias pelo menos duas cores diferentes.

Definiu-se a cor vermelha, cujo código RGB é o (255,0,0), para representar situações de perigo, em que o observador em questão deve parar. Já os excessos de velocidade, onde o objetivo é fazer os condutores reduzir a velocidade, são sinalizados pela cor laranja (255,50,0).

Estes perfis serão utilizados ao longo de toda a dissertação. Porém, pela mesma razão que decidimos excluir a cor verde, podemos precisar de substituir o vermelho no futuro. É aqui que entra a vantagem de utilizar RGB, porque nos permite alterar os perfis luminosos a qualquer altura, sem precisar de fazer alterações de *hardware*.

Optou-se pelas tiras WS2812B com revestimento em silicone (IP65), de 30 LEDs por metro e alimentação 5V DC - Figura 4.8. Estes LEDs são Surface Mount Device (SMD) 5050 e apresentam um consumo de 9 watts por metro.

A diferença desta tira para as mais comuns está no facto de ser endereçável, permitindo controlar cada LED de forma individual, através de um sinal digital único. Este sinal utiliza um protocolo de comunicação Non-Return to Zero (NRZ).



Figura 4.8: Tira de LEDs RGB endereçável WS2812B 5050

Luminárias:

À imagem do que aconteceu com o radar, verificou-se a necessidade de proceder a um estudo mais intensivo das alternativas do mercado, antes de escolher as luminárias mais adequadas.

As características pretendidas são as seguintes:

- IP65 ou semelhante;
- Boa intensidade luminosa (cerca de 50W);
- Tempo de vida útil de 50.000h ou superior;
- Controlo do fluxo luminoso por *dimming*;

Foram então ponderadas as opções da tabela 4.3.

³sistema que combina vermelho, verde e azul, para obter qualquer cor do espectro visível

Empresa	Alimentação (V)	Potência (W)	Tipo de Proteção	Tempo de vida útil (h)	Eficiência (lm/W)	Dimming
Brooklyn	24 DC	40	IP65	> 50.000	95	×
Solarcupid	24 DC	48	IP65	> 50.000	100	×
Ledenvo	36 DC	30	IP66	> 50.000	120	×
DLeds	230 AC	18 a 73	IP66	> 100.000	107 - 126	✓
Lux Concept	230 AC	23 a 360	IP66	> 100.000	130	✓
Indo	230 AC	30	IP66	-	132	✓

Tabela 4.3: Alternativas de luminárias para iluminação de presença

As primeiras três opções (*Brooklyn* [42], *Solarcupid* [43] e *Ledenvo* [44]) foram descartadas por não permitirem controlar o fluxo luminoso.

Estudaram-se as restantes alternativas em detalhe, pedindo orçamentos e informações às empresas. No entanto, a escolha acabou por ser mais simples do que o previsto.

Ao entrar em contacto com a *Lux Concept*, empresa sediada em Aveiro e especialista na produção e instalação de iluminação LED, esta apresentou uma proposta que permitia adquirir as luminárias a título definitivo ou por empréstimo. A proximidade das instalações e flexibilidade que a empresa demonstrou, levou a que esta fosse a escolha mais lógica.

As luminárias foram então produzidas de acordo com as características que pretendíamos. Optou-se pela série *Roma* que utiliza tecnologia da *OSRAM*, com as seguintes propriedades: 18 LEDs, 50W de potência, *driver MeanWell* com opção de dimming e temperatura da cor 4000°K. O resultado pode ser visto na figura 4.9.



Figura 4.9: Luminária LED “Roma”

4.1.3 Fluxograma funcional

Depois de definir e adquirir o material, podemos avançar com a programação do sistema. Mas para isso é necessário estabelecer uma sequência lógica de operações a realizar, de modo a que nenhuma ação seja menosprezada. A forma mais lógica de esquematizar essa sequência operacional é recorrendo a um fluxograma.

As comunicações entre postes vão ser deixadas para segundo plano, pelo menos por agora, já que neste momento o principal objetivo é ter um dos postes 100% funcional.

As figuras 4.10, 4.11 e 4.12 traduzem o código que se pretende implementar no poste principal (aquele onde será instalado o radar).

O programa utiliza *threads* para executar tarefas concorrentemente. A divisão é feita da seguinte forma:

Designação	Cor	Funcionalidade	Condições de arranque
Fluxo principal	Azul	Estabelece a ordem de execução; Efetua cálculos de luminosidade; Gere a iluminação de presença.	- Início do programa
Tarefa 1 (T1)	Laranja	Deteta peões	- Após inicializações; - Se não forem detetados peões; - Após T3 arrancar.
Tarefa 2 (T2)	Verde	Deteta e trata os excessos de velocidade	- Após T1 arrancar, se T2 e T3 não estiverem em execução.
Tarefa 3 (T3)	Cinza	Executa o procedimento de travessia de peões	- Após T1 e T2 terminarem, se forem detetados peões e T3 não estiver em execução.

Tabela 4.4: Divisão do código em tarefas

De notar as seguintes propriedades:

- As tarefas T2 e T3 utilizam ambas os LEDs de sinalização, pelo que nunca podem ser realizadas em simultâneo;
- A tarefa T1 tem de ser executada sempre que possível, ocorrendo em paralelo com T2 ou T3, para que a deteção de peões se mantenha atualizada;
- É utilizado multiprocessamento dentro da tarefa T2.

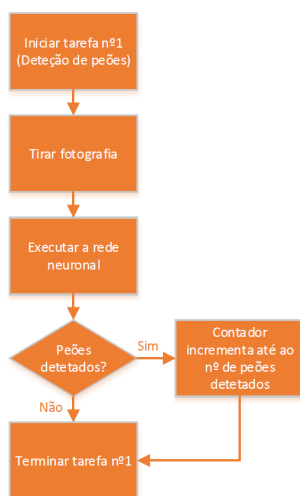


Figura 4.10: Fluxograma funcional da tarefa T1

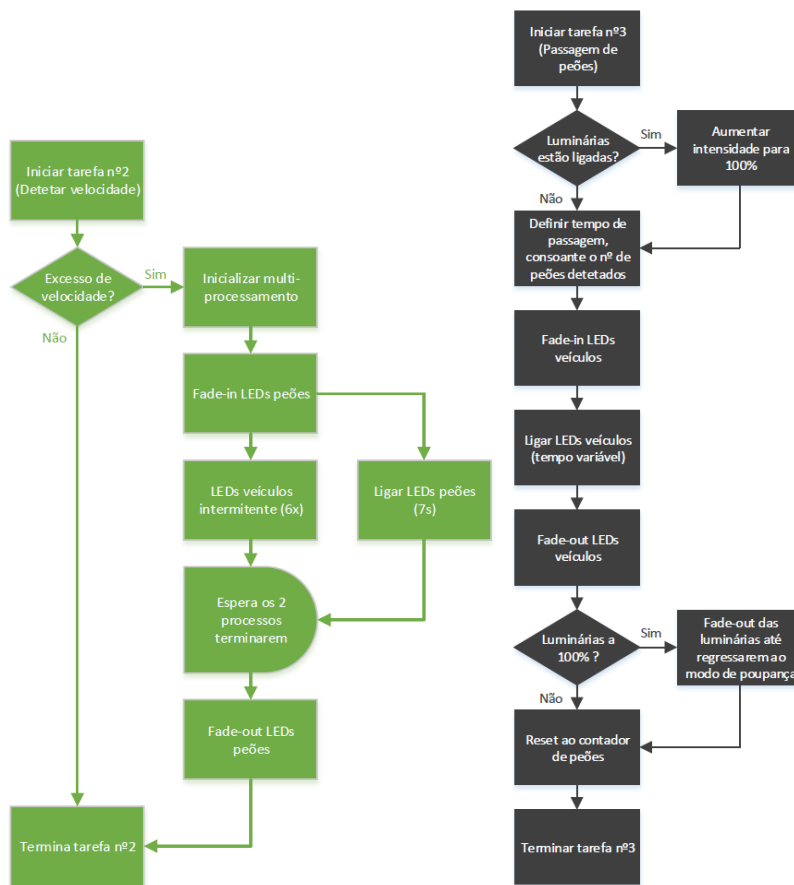


Figura 4.11: Fluxograma funcional da tarefa T2 (esquerda) e T3 (direita)

No que diz respeito à duração dos alertas, considerando que os veículos são detetados a 150m das passadeiras, calculou-se durante quantos segundos se deve manter, em média, o aviso de excesso de velocidade ativo:

$$50km/h \simeq 13,9m/s \quad 120km/h \simeq 33,3m/s$$

$$\frac{150}{13,9} = 10,8s \quad \frac{150}{33,3} = 4,5s$$

Duração média: 8s

Este é o tempo aproximado que a tarefa T2 demora a ser concluída. O sinal intermitente é atuado a uma frequência de 1 vez por segundo, em paralelo com o sinal de perigo para peões, que dura 7s. Somam-se ainda o *fade in* e *fade out* dos LEDs direcionados aos peões, que demoram cerca de 0,5s cada, de modo a que a transição não seja tão abrupta.

Por outro lado, o tempo de passagem para os peões varia consoante o número de pessoas detetadas. Ou seja, se o fluxo de pedestres for grande num certo instante, podemos ter um intervalo de tempo maior para escoar os peões em segurança. Se o fluxo for pequeno, podemos encurtar o tempo de espera dos condutores.

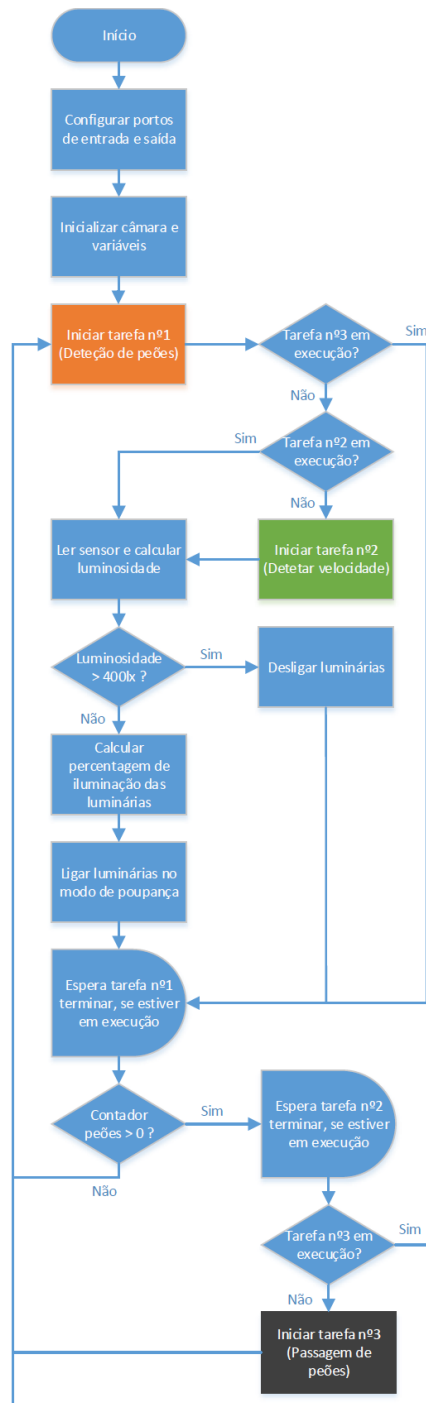


Figura 4.12: Fluxograma funcional do fluxo principal

De qualquer modo é fundamental estimar o tempo médio de passagem, pelo menos para o caso mais comum (deteção de uma pessoa), para que os valores não sejam lançados ao acaso.

Supondo uma passadeira com 7m de comprimento e velocidade dos peões entre os 3 e os 4,5 km/h, temos que:

$$3\text{km/h} \simeq 0,83\text{m/s} \quad 4,5\text{km/h} \simeq 1,25\text{m/s}$$

$$\frac{7}{0,83} = 8,4\text{s} \quad \frac{7}{1,25} = 5,6\text{s}$$

Duração média: 7s

Este é o tempo que serve de base para a travessia de uma pessoa. A partir daqui, definiram-se os seguintes valores:

Número de peões	Tempo (s)
1 ou 2	7
3 a 5	10
6 a 9	13
> 10	16

Tabela 4.5: Tempos de passagem consoante os peões detetados

A programação do segundo poste é em tudo igual ao que foi apresentado até aqui. A única diferença é que como não se implementou um segundo radar, a deteção de excessos de velocidade é feita apenas num dos sentidos, podendo-se remover a tarefa T2 do código do poste secundário. Esta configuração poderá ser útil nos casos em que as passadeiras se encontram nas proximidades de um cruzamento, onde de facto apenas é necessário um radar.

4.1.4 Eletrónica adicional

Para interligar corretamente os componentes à *Raspberry Pi*, foi necessário desenvolver alguma eletrónica básica, tanto para fornecer as alimentações necessárias, como para amplificar os sinais provenientes deste controlador.

Construiu-se então uma pequena placa como mostra a figura 4.13.



Figura 4.13: Placa de alimentação e amplificação de sinais

A placa é composta por três circuitos distintos, com os seguintes objetivos:

1. Amplificar os sinais digitais de controlo das tiras de LEDs de 3.3V para 5V;
2. Amplificar o Pulse-Width Modulation (PWM) responsável pelo *dimming* da luminária;
3. Fornecer alimentação de 10V DC ao amplificador do PWM.

O primeiro circuito utiliza um conversor de nível lógico bidirecional de dois canais [48] que converte 3,3V em 5V e vice-versa - figura 4.14. A alimentação deste componente é fornecida pela própria *Raspberry*.

Os sinais digitais, também provenientes da *Raspberry*, entram nas portas de *low-voltage* (TXI) um por cada canal, passam por um Metal Oxide Semiconductor Field Effect Transistor (MOSFET) e saem nas portas de *higher voltage* (TXO).

Foi adicionado um condensador de $100\mu F$ entre a alimentação das tiras de LED (5V e GND), para prevenir que uma corrente inicial danifique os *pixels*, e uma resistência de 270Ω no caminho de cada sinal, para reduzir o ruído na linha.

Uma última nota em relação às tiras de LED vai para os fios utilizados que devem ser o mais curtos possíveis, para minimizar as perdas de tensão.

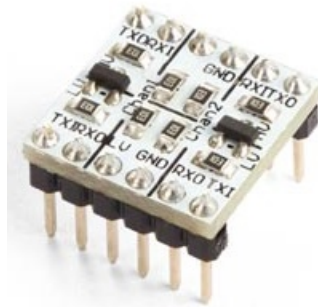


Figura 4.14: Conversor de nível lógico bidirecional [48]

O segundo circuito utiliza Bipolar Junction Transistor (BJT)s NPN para amplificar o sinal que irá controlar a intensidade da luminária.

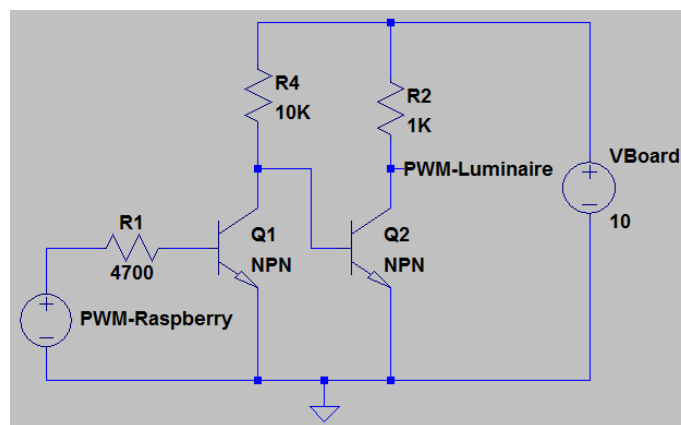


Figura 4.15: Esquema do amplificador com base em BJTs

O *driver* responsável pelo *dimming* é o NPF-60D de 60W, alimentado a 230V e com uma saída de 36V. Apresenta uma eficiência em torno de 90% e proteção IP67.

O *dimming* pode ser feito de 3 modos diferentes: aplicando um sinal 0-10V DC, um PWM de 10V DC, ou uma resistência aditiva. Optou-se pelo PWM como apresentado na figura 4.16.

© Applying additive 10V PWM signal (frequency range 100Hz ~ 3KHz):

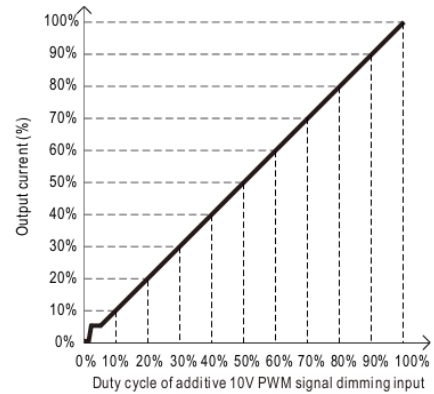
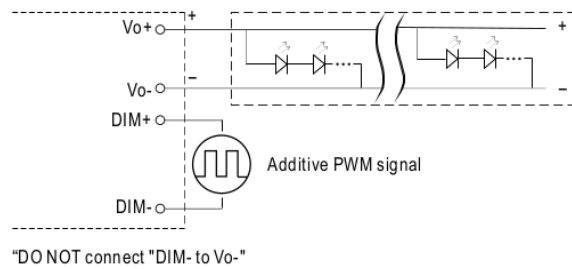


Figura 4.16: Modo de operação do driver [49]

O PWM é gerado pela *Raspberry* com uma frequência de 1 kHz e nível alto de 3,3V.

Foi utilizada a biblioteca *pigpio* em *python* [50] para controlar o porto General-Purpose Input/Output (GPIO) responsável por esta tarefa, porque a biblioteca RPi.GPIO normalmente utilizada para gerir os portos, faz o controlo dos tempos por *software* e não por *hardware*, originando *delays* que podiam fazer a luminária cintilar.

Como à entrada do *driver* (DIM+, DIM-) são esperados 10V DC, o sinal é então amplificado através de dois transístores BC547, representados na figura 4.15 por Q1 e Q2.

O resultado esperado pode ser observado no gráfico da figura 4.17, onde a verde está representado o sinal à saída da *Raspberry* (antes do amplificador) e a vermelho o sinal à entrada do *driver* (depois do amplificador).

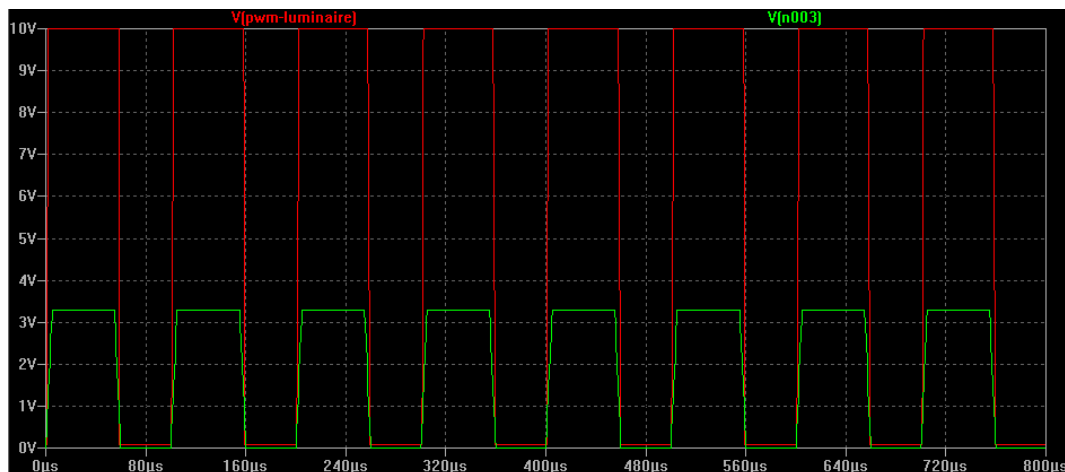


Figura 4.17: Simulação do circuito da figura 4.15

Para que isto aconteça é necessário fornecer a alimentação de 10V representada na figura 4.15 por “VBoard”. O circuito que satisfaz esta condição é o da figura 4.18.

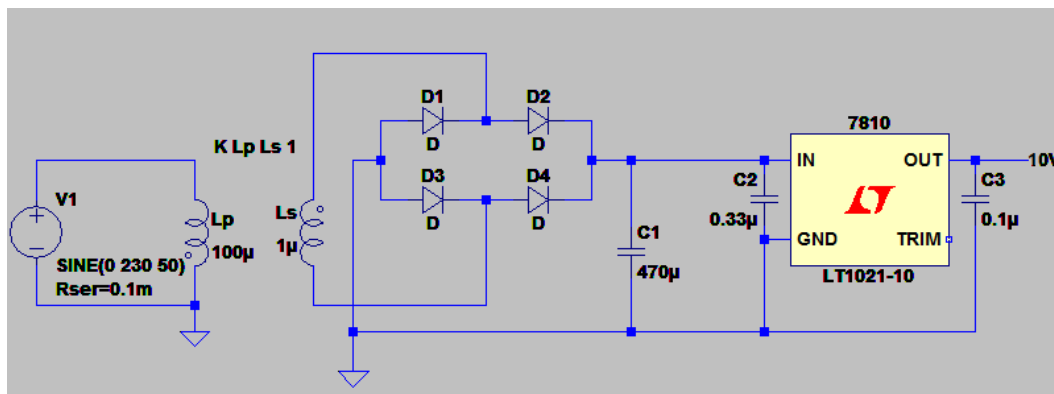


Figura 4.18: Esquema do circuito de alimentação

Fazem parte deste circuito os seguintes componentes: um transformador encapsulado TEZ 2,0/D para circuito impresso que converte 230V em 24V (2VA, 83mA) [51], uma ponte retificadora com quatro díodos 1N4007, um condensador de $470\mu F$ para filtragem e um regulador de tensão L7810 [52].

Na Printed Circuit Board (PCB) é possível encontrar um segundo regulador (L7805) que foi montado para caso fosse necessária uma alimentação complementar de 5V DC, algo que não se veio a verificar.

Este circuito foi simulado aplicando uma resistência de carga de $1K\Omega$, comprovando assim o seu funcionamento teórico (figura 4.19).

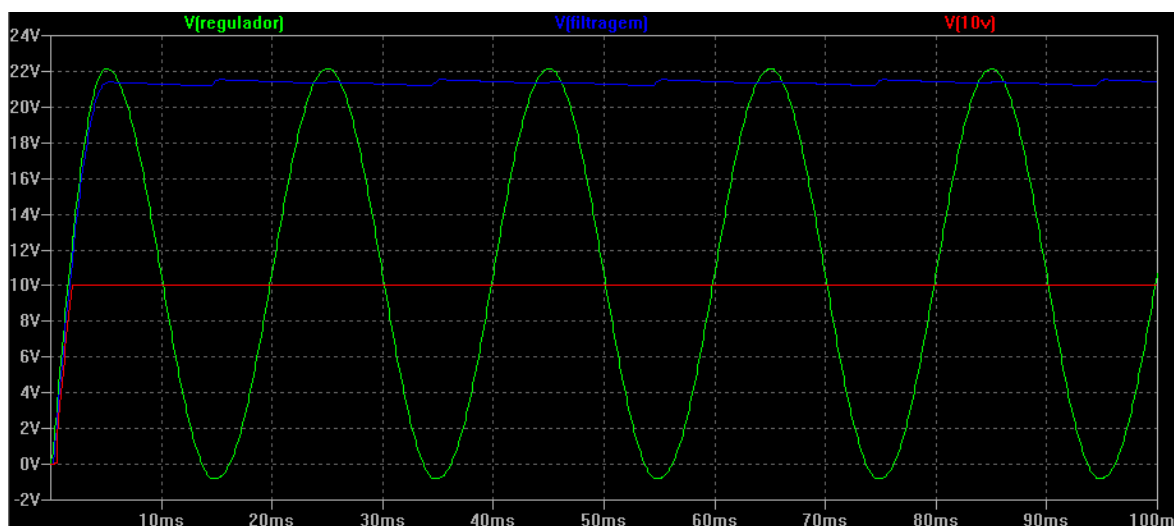


Figura 4.19: Simulação do circuito da figura 4.18

Concluída a análise dos três circuitos que compõem esta placa, resta referir que todos os circuitos foram testados e validados na prática, tendo funcionado de acordo com o que seria expetável. Foram desenvolvidas 2 placas semelhantes, uma para cada poste.

4.1.5 Instalação

Depois de construir a estrutura física que irá suportar a tecnologia, adquirir todo o material necessário, programar a *Raspberry* de acordo com os pressupostos e desenvolver a eletrónica em falta, é chegada a altura de interligar tudo isto e instalar o sistema num local apropriado, para que possa ser devidamente configurado e testado.

O local escolhido foi uma passadeira na freguesia de Pigeiros, em Santa Maria da Feira, Portugal. A figura 4.20 mostra uma fotografia do local ainda sem o sistema.



Figura 4.20: Passadeira escolhida para instalação do protótipo

Trata-se de uma passadeira sem semaforização, num local relativamente calmo, o que é ótimo para fazer ajustes no sistema, mas que ainda assim fica numa das principais estradas da região, com um fluxo de tráfego razoável nas chamadas horas de ponta (entre as 8 e as 10 da manhã e entre as 17 e as 19 da tarde), permitindo testes mais rigorosos nesses períodos.

Para instalar o protótipo será necessário, pelo menos numa primeira fase, alimentar cada um dos postes com 230V AC, o que não será um problema.

Utilizando uma abordagem lógica e crescente em complexidade, vamos começar por referir como se efetuam as ligações, seguindo-se uma análise à montagem dos componentes.

Ligações:

Começando pela câmara, basta conectar o cabo *ribbon flat*⁴ na entrada adequada, conforme o esquema da figura 4.21 e o módulo estará pronto para ser configurado.

As restantes conexões com a *Raspberry* serão através dos portos GPIO, de acordo com a estrutura definida na tabela 4.6.

As configurações dos portos e a toda a programação foram realizados por controlo remoto, a partir do computador ou do *smartphone*, recorrendo a uma Virtual Private Network (VPN)

⁴cabo flexível com diversos fios paralelos no mesmo plano

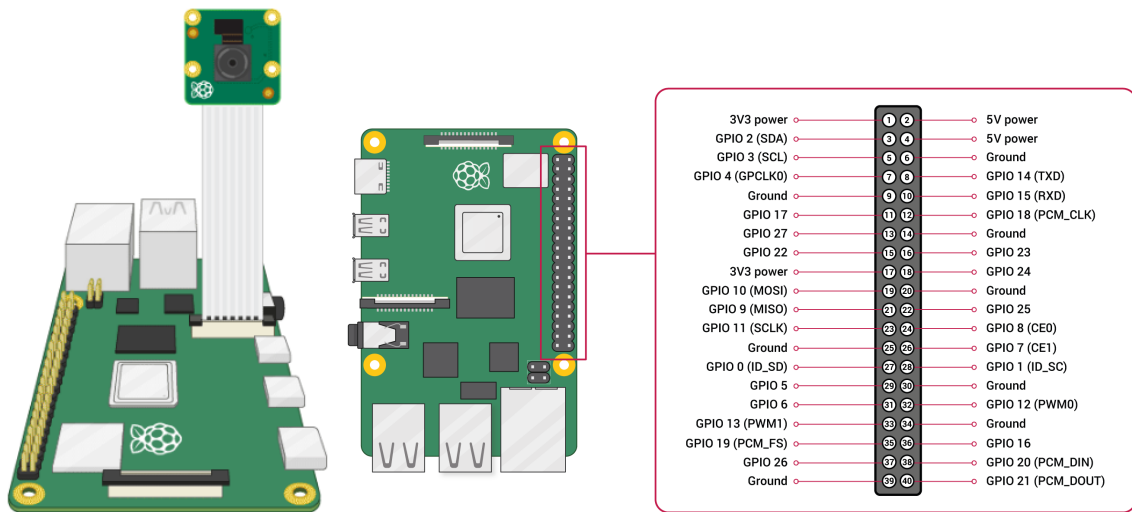


Figura 4.21: Conexão do módulo de câmara (esquerda) e pinout (direita) [53]

Posição	Designação	Funcionalidade
1	3.3V	Alimentação do sensor TSL2591 Alimentação do conversor de nível lógico
2	5V	Alimentação das tiras de LEDs Alimentação do conversor de nível lógico
3	GPIO2 (SDA)	Linha de dados I2C do sensor TSL2591
5	GPIO3 (SDL)	Linha de <i>clock</i> I2C do sensor TSL2591
6	GND	Massa das tiras de LEDs
9	GND	Massa do sensor TSL2591
11	GPIO17	Saída do PWM para o amplificador
12	GPIO18	Sinal digital de controlo da tira de LEDs direcionada aos peões, pré-amplificador
14	GND	Massa da placa eletrónica (figura 4.13)
25	GND	Massa do <i>driver</i> da luminária (DIM-)
29	GPIO5	Relé auxiliar do radar - contacto 1
30	GND	Relé auxiliar do radar - contacto 2
33	GPIO13	Sinal digital de controlo da tira de LEDs direcionada aos veículos, pré-amplificador

Tabela 4.6: *Pinout* da *Raspberry Pi* e respetivas conexões ao sistema

e ao software Virtual Network Computing (VNC), um sistema gráfico que permite a partilha e interação com o *desktop* do *Raspberry Pi*.

Falta ainda referir as ligações que têm origem na PCB referida no ponto 4.1.4. São elas:

- Sinal digital amplificado (5V), de controlo da tira de LEDs direcionada aos peões;
- Sinal digital amplificado (5V), de controlo da tira de LEDs direcionada aos veículos;
- PWM amplificado (10V), direcionado ao *driver* da luminária (DIM+).

Montagem:

Para compreender melhor a constituição do protótipo, precisamos de saber mais detalhes sobre os componentes e decidir como integrá-los devidamente.

Divulgam-se agora os esquemas com as dimensões daqueles que são dois dos componentes mais importantes, a luminária e o radar (figuras 4.22 e 4.23, respetivamente).

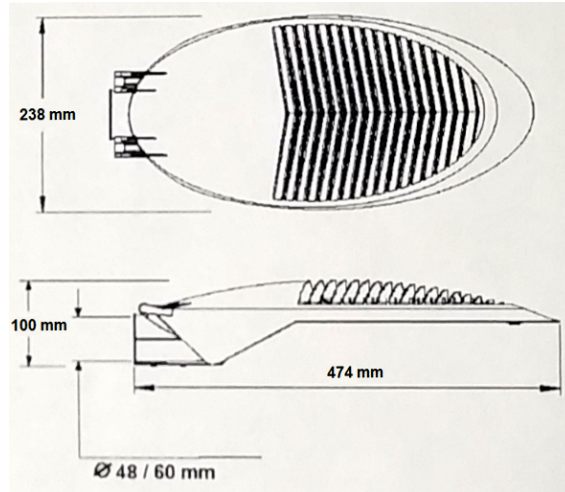


Figura 4.22: Esquema com as dimensões da luminária

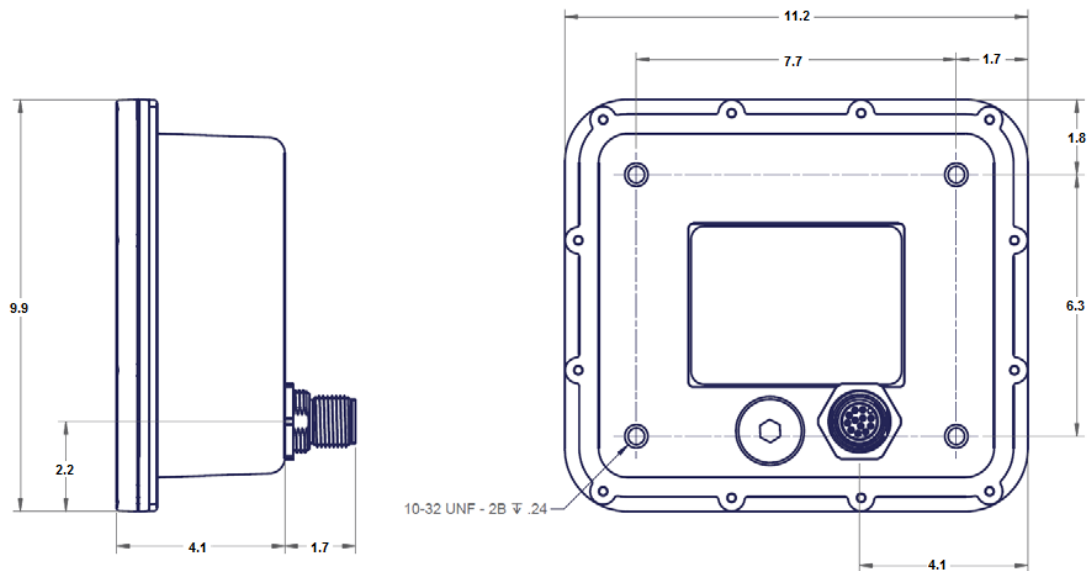


Figura 4.23: Esquema com as dimensões do radar [54]

Podemos verificar que se trata de uma luminária bastante compacta, com uma altura máxima de 10cm. O *driver* encontra-se no interior, na zona mais recuada, e próximo do encaixe ajustável que permite dois tipos de instalação, consoante o ângulo que lhe é aplicado - ver figura 4.24.

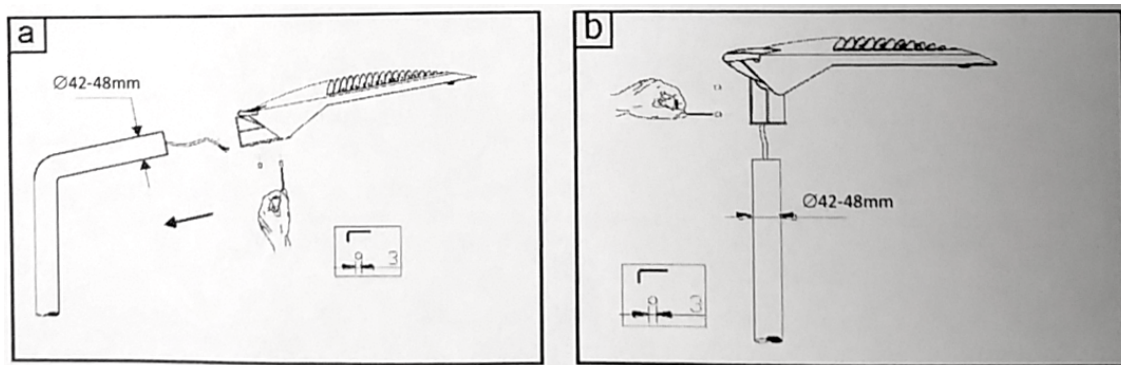


Figura 4.24: Modos de instalação da luminária: a) em braço; b) *post-top*

Por sua vez, o radar é apresentado com as suas dimensões em centímetros e possui um conector circular de nove pinos (tipo *mike*) que se estende até ao adaptador da figura 4.25.

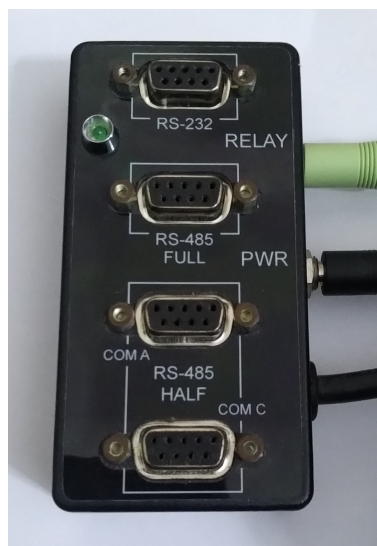


Figura 4.25: Adaptador e alimentação do radar

A alimentação é fornecida através da entrada “PWR”, proveniente de uma fonte de 12V/2A, sendo que o radar consome 85mA durante a transmissão e 40mA quando não está a transmitir.

Dependendo da configuração do radar, uma, duas, três ou quatro portas COM podem ser utilizadas em simultâneo e encontram-se dispostas conforme é indicado na tabela 4.7.

COM A	4-wire full-duplex RS-485 port
COM B	2-wire full-duplex RS-232 port
COM C	2-wire half-duplex RS-485 port
COM D	Full-duplex USB port (flow control option)

Tabela 4.7: Portas COM disponibilizadas pelo radar

Na configuração de fábrica a porta COM C encontra-se desativada e nesta versão em particular, a porta USB não se encontra disponível.

A saída designada por “Relay” é referente a um contacto auxiliar normalmente aberto, de dois fios (*mono jack* de 3,5mm), que fecha em determinadas condições. Se for programado para ser um alarme de velocidade, é possível definir um *threshold* por software, de forma a que velocidades acima ou abaixo desse valor acionem o contacto. Se por outro lado for programado para o modo de estatísticas, permite conectar um LED informativo que piscará lentamente, informando que a unidade está a monitorizar o tráfego e à medida que cada alvo é registado emite uma sequência mais rápida.

Apresentados os detalhes que caracterizam estes componentes, passamos agora à integração no poste principal. Como tem vindo a ser referido até aqui, a posição, altura e direção dos componentes é de extrema importância.



Figura 4.26: Visão lateral do poste, voltado para a via (detecção de velocidade)

No topo da estrutura e portanto a 3m do solo, encontra-se o sensor de luminosidade, posicionado de forma a que as leituras não sejam afetadas pelas luminárias.

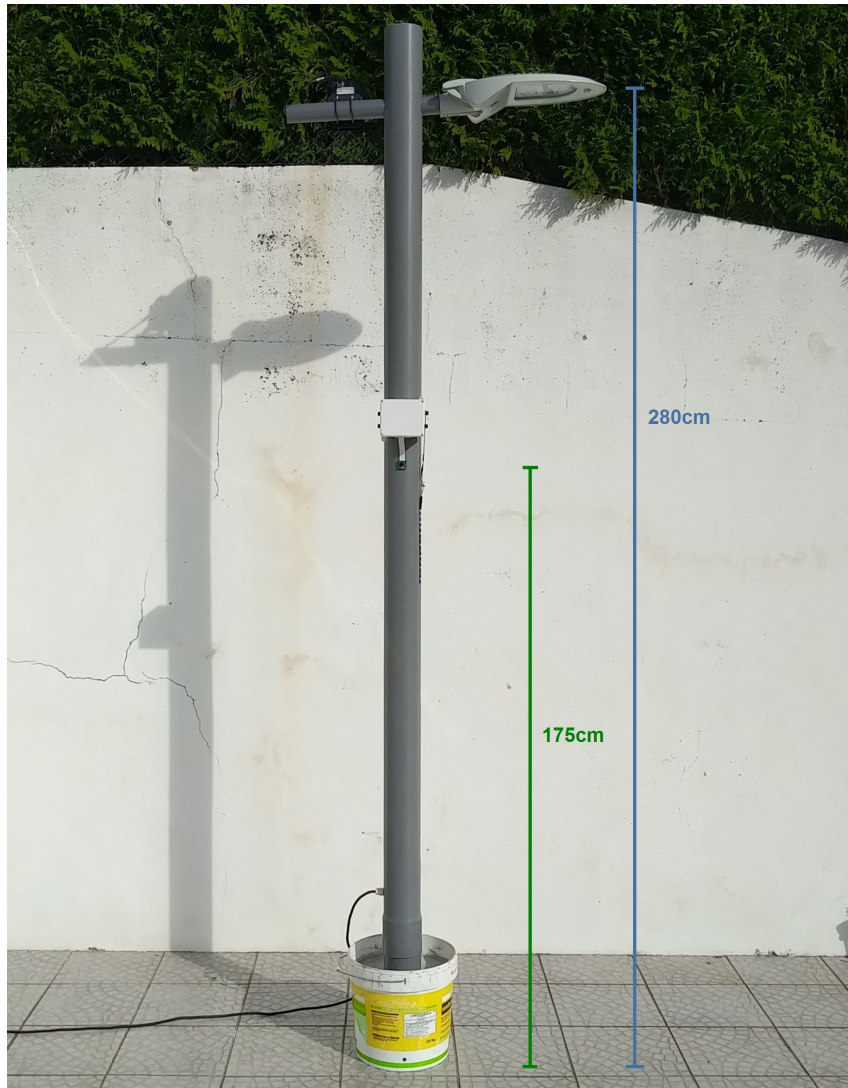


Figura 4.27: Visão lateral do poste, voltado para a passadeira (detecção de peões)

Um pouco mais a baixo temos a luminária, direcionada para a passadeira, e o radar, voltado para a zona de aproximação das viaturas. Ambos foram aplicados num tubo que atravessa perpendicularmente o poste, mas é importante notar que se utilizou uma pequena inclinação, de modo a que o fluxo luminoso seja direcionado ao centro da passadeira.

A *Raspberry Pi*, a PCB e as respetivas ligações, foram embutidas numa caixa de derivação com 120x100x60mm e foi feita uma ranhura na parte inferior para passar a câmara, afixada a uma altura de 1,75m. Tanto a câmara como a caixa estão voltadas para a entrada da passadeira.

As duas tiras de LEDs têm aproximadamente 25cm e estão a mais de 1,40m do solo. Esta altura foi considerada a mais adequada, através dos testes realizados, para permitir boa visibilidade tanto a condutores como pedestres.

Toda a cablagem passa pelo interior do tubo e deriva de um único cabo de alimentação de 230V AC, na parte inferior do poste.

4.2 Gestão de tráfego

Não basta apenas conectar e direcionar o radar para se obterem bons resultados.

Tal como é dito no manual técnico do sensor, cada instalação deve ser tratada como um caso único, existindo muitos aspetos a ter em conta durante a montagem do equipamento e todos eles têm impacto no comportamento e fiabilidade dos resultados.

Serão analisadas nesta secção as variáveis que se consideram mais relevantes, como por exemplo o posicionamento do radar e a sensibilidade. No entanto, é aconselhável a leitura dos manuais disponibilizados pela *Stalker* (manual de utilizador, manual técnico e manual de operador do sistema de estatísticas), caso se pretendam mais informações, visto que é impossível cobrir todos os tópicos neste documento.

4.2.1 Configuração do radar

Ainda antes das configurações é necessário ajustar o posicionamento do radar.

A localização mais recomendada é acima e no centro da faixa que se pretende monitorizar, apontando com uma inclinação descendente para o centro da via de trânsito.

O sensor possui uma largura de feixe de 6° na horizontal e uma largura de 26° na vertical, criando um padrão trapezoidal quando atinge a estrada, como demonstrado na figura 4.28.

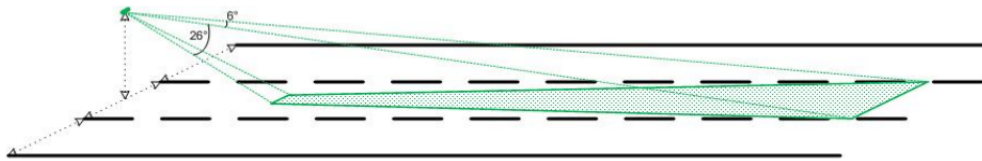


Figura 4.28: Esquema representativo da zona de deteção do radar [54]

A maioria da potência é transmitida nesta região, no entanto, é por demais evidente que alguma da potência se estende para além destas margens, apenas com menor intensidade.

Para estimar esta área precisamos fazer alguns cálculos, recorrendo à figura 4.29.

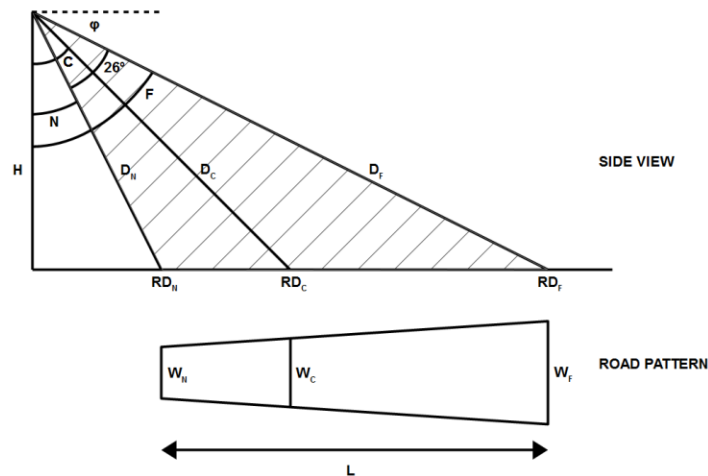


Figura 4.29: Diagrama com a geometria da instalação do radar [54]

Na figura anterior, H corresponde à altura do radar e C é o ângulo que lhe é aplicado, medido desde o centro. Daqui derivam N e F que dizem respeito ao ângulo mais próximo (*Near*) e mais afastado (*Far*), respetivamente. Podem ser calculados da seguinte forma:

$$N = C - 13^\circ \quad (4.1)$$

$$F = C + 13^\circ \quad (4.2)$$

O feixe emitido atinge a estrada a uma distância mínima RD_N do poste e máxima RD_F que se calculam:

$$RD_N = H \times \tan(N) \quad (4.3)$$

$$RD_F = H \times \tan(F) \quad (4.4)$$

No que diz respeito à largura, o feixe aumenta aproximadamente segundo a fórmula $W = 0,1 \times D$, onde D é a distância percorrida. Assim, sabendo que as distâncias D_N e D_F são dadas pelas equações 4.5 e 4.6, podemos calcular a largura mais próxima $W_N = 0,1 \times D_N$ e mais distante $W_F = 0,1 \times D_F$.

$$D_N = \sqrt{H^2 + RD_N^2} \quad (4.5)$$

$$D_F = \sqrt{H^2 + RD_F^2} \quad (4.6)$$

Aplicam-se agora estes conhecimentos à instalação que temos em mãos. Conhecem-se à partida os seguintes valores: $H = 2,80m$, $RD_F = 150m$. Utilizando a equação 4.4 é possível estimar o valor de F :

$$F = \arctan\left(\frac{RD_F}{H}\right) = \arctan\left(\frac{150}{2,8}\right) \simeq 89^\circ \quad (4.7)$$

Desta forma, pelas equações 4.1 e 4.2, temos que $C \simeq 76^\circ$ e $N \simeq 63^\circ$.

Posicionando o sensor em conformidade com os ângulos obtidos, podemos calcular todas as distâncias. Os valores obtidos encontram-se na tabela 4.8.

RD_N	RD_C	RD_F	D_N	D_C	D_F	W_N	W_C	W_F
5,5m	11,2m	160,4m	6,2m	11,5m	160,4m	0,6m	1,2m	15m

Tabela 4.8: Valores aproximados da zona de deteção do radar

A estrada onde o sistema vai ser instalado tem uma largura de 7m e a via de trânsito onde os veículos normalmente circulam, no sentido de aproximação, possui cerca de 3,5m.

Assim sendo, podemos esperar que a totalidade da via seja abrangida pelo radar, entre os 35m e os 150m de distância da passadeira.

$$W = 0,1 \times D \Leftrightarrow 3,5 = 0,1 \times D \Leftrightarrow D = 35m$$

$$RD = \sqrt{35^2 - 2,8^2} \simeq 34,9m$$

Obviamente que estes valores sofreram múltiplas aproximações e podem variar por diversas razões. Servem, no entanto, como ponto de partida para uma instalação mais consciente.

Posicionado o radar, podemos então dar início às configurações. O método mais simples nesta fase é recorrer ao *software* “*Dashboard Application*”, onde as configurações dividem-se em básicas e avançadas.

Configurações básicas:

É possível colocar o radar a funcionar alterando apenas algumas definições. São elas: o modo de transmissão, a direção dos alvos, as unidades e a resolução apresentada. Os parâmetros e os valores disponíveis encontram-se sumariados na tabela 4.9.

Parâmetro	Valores disponíveis	Default	Valor definido
Controlo da Transmissão	0 = Aguardar 1 = Transmitir 2 = Automático	1	2
Direção dos veículos	0 = Ambos os sentidos 1 = Aproximação 2 = Afastamento	0	1
Unidades	0 = mph 1 = km/h 2 = nós 3 = m/s 4 = pés/s	0	1
Resolução	0 = Unidade 1 = Décimas	0	0

Tabela 4.9: Resumo das configurações básicas do radar

O modo de transmissão define se o transmissor está ligado (1) ou desligado (0). Optou-se pelo modo automático, porque é aquele que deve ser utilizado para recolher dados estatísticos. Quando esta opção é selecionada o transmissor desliga-se sempre que não está a recolher dados, resultando num menor consumo e permitindo a recolha de informação durante mais tempo, no caso de se utilizar a memória interna do sensor.

A direção dos veículos permite definir se queremos que todos os veículos sejam analisados, ou apenas aqueles que se deslocam num sentido. Como queremos medir a velocidade de aproximação à passadeira, a escolha lógica é o valor 1.

A unidade de medida escolhida foi o km/h e os valores arredondados à unidade.

Configurações avançadas:

Entrando agora nas configurações mais avançadas, é possível caracterizar os veículos por tipo, filtrar os resultados, corrigir as leituras de velocidade, entre muitas outras opções.

O primeiro passo é fazer um ajuste na sensibilidade do sensor consoante a aplicação pretendida. Este é um dos parâmetros mais importantes, já que afeta diretamente a capacidade de reconhecer ou não determinado veículo.

É possível definir um valor de sensibilidade entre 0 e 16, sendo 16 o valor máximo e 0 a ausência de deteção. Quanto maior a sensibilidade, maior o alcance do radar e a capacidade de detetar um objeto (máximo desempenho). Permite também detetar alvos mais pequenos.

No entanto, nem sempre é desejável utilizar o valor máximo. Por vezes é muito conveniente reduzir a sensibilidade, por forma a detetar apenas os veículos mais próximos da unidade, filtrando assim o *background*.

Outra forma de filtragem é o *threshold* nos limites de velocidade inferior e superior.

No que ao limite inferior diz respeito, alterou-se a velocidade de 0 km/h para 10 km/h. Desta forma as deteções indesejadas serão ignoradas, sejam pessoas a caminhar na estrada, ou até mesmo objetos que se movam nas redondezas (por exemplo as árvores a abanar devido à passagem do vento).

Caso não seja possível instalar o radar diretamente por cima da via que se pretende monitorizar, pode-se recorrer a uma localização alternativa, como demonstra a figura 4.30.

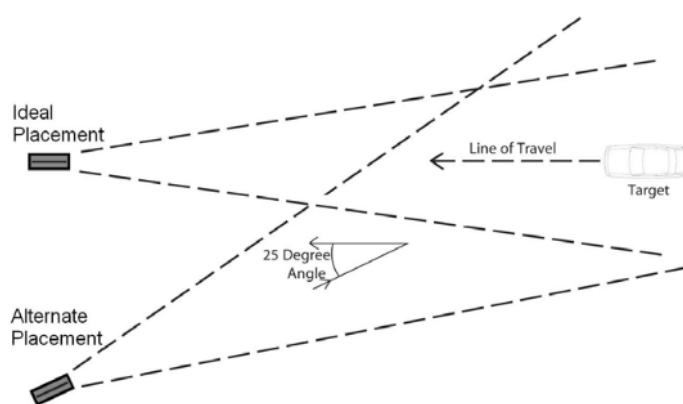


Figura 4.30: Esquema de instalação alternativa do radar [54]

No entanto, como em qualquer outro radar, uma montagem na diagonal leva a medições inferiores à velocidade real dos veículos. Se o ângulo for muito acentuado essa diferença deixa de ser desprezável, gerando um erro sistemático que não deve ser ignorado. Para corrigir estes erros é possível configurar dois parâmetros relativos ao cosseno do ângulo vertical e horizontal da instalação.

Supondo agora que a passadeira onde se pretende instalar o radar fica nas proximidades de um cruzamento (como de facto acontece na instalação que estamos a levar a cabo), e que pretendemos analisar a velocidade apenas dos veículos que circulam na estrada principal e não daqueles que surgem do cruzamento, podemos utilizar uma configuração chamada distância mínima de rastreio.

É importante notar a distância mínima de rastreio é o valor, expresso em *pés*, dentro do qual não serão considerados veículos, a menos que estes já estejam a ser detetados desde uma distância superior. Assim sendo, como na direção para a qual pretendemos apontar o radar, existe um cruzamento a menos de 15m, definiu-se a distância mínima de rastreio como 50 (50 pés \simeq 15,24 metros).

As restantes configurações estabelecem por exemplo, como alterar o relógio interno do sensor, os tempos de leitura dos dados e o momento a partir do qual se deixa de detetar um veículo, mas como não serão alvo de discussão, resumem-se na tabela 4.10 apenas os pontos que já foram abordados.

Parâmetro	Valores disponíveis	Default	Valor definido
Sensibilidade	0 - 16	16	12
Limite de velocidade inferior	0 - 322	0	10
Limite de velocidade superior	0 - 322	200	322
Cosseno do ângulo horizontal	0 - 70	0	0
Cosseno do ângulo vertical	0 - 70	0	0
Distância mínima de rastreo	0 - 500	200	50

Tabela 4.10: Resumo de algumas das configurações avançadas do radar

4.2.2 Protocolos de comunicação

Todas as portas COM do sensor são configuradas para comunicações série assíncronas de 10 bits, com 1 bit de início, 8 bits de dados, 1 bit de paragem e sem paridade (8N1).

A porta escolhida para conectar o radar foi a COM B, ou seja RS-232 *full-duplex*⁵, composto por dois sinais, um de transmissão (Tx) e um de recepção (Rx), relativos à massa (GND).

Todas as configurações foram realizadas a partir de um computador a correr *software Stalker*. Depois de configurado, o sensor pode ser desconectado, ficando a trabalhar de forma independente e armazenando a informação na memória interna. Passado algum tempo podemos voltar a conectar o radar e recolher as informações estatísticas para o computador.

Outra opção seria manter a conexão após as configurações e receber os dados em tempo real, como demonstra a figura 4.31.

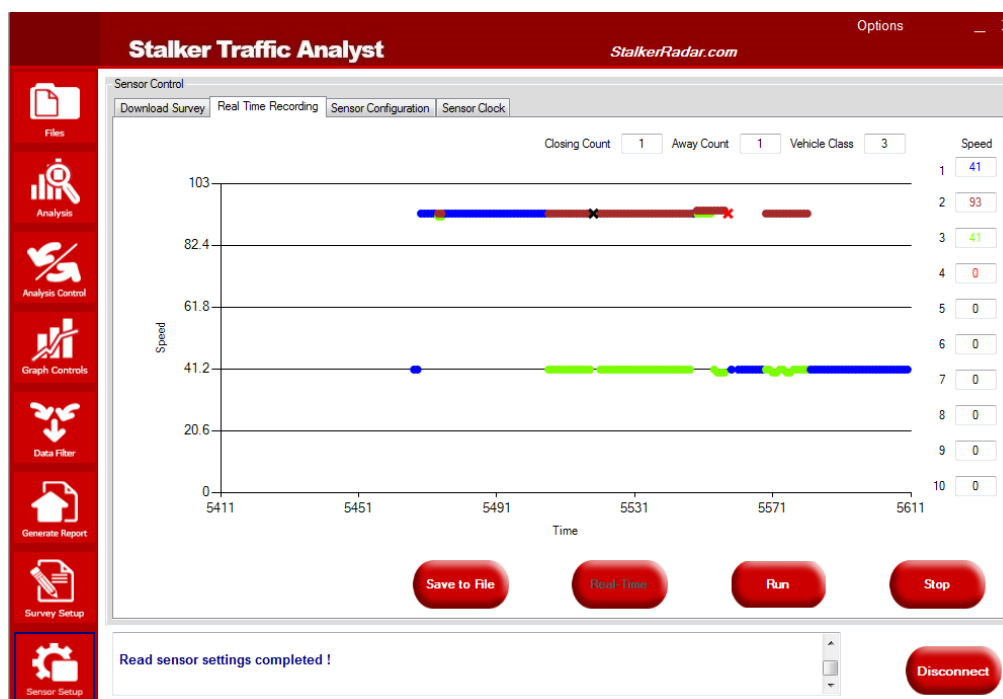


Figura 4.31: Leitura de dados do radar em tempo real [54]

⁵ os dois sinais de dados podem circular em ambos os sentidos sem se corromperem mutuamente

Se o objetivo for ligar o radar a um microcontrolador ou utilizar um computador que não possua o *software* de análise de tráfego, pode-se utilizar uma aplicação *standard* como o *RealTerm* ou o *PuTTY*, bastando para isso configurar um dos onze tipos de protocolos de *streaming* disponíveis para o envio de mensagens periódicas.

Cada protocolo tem características diferentes, alterando tanto o conteúdo como o ritmo de transmissão das mensagens. A tabela 4.11 apresenta todos os formatos suportados.

Formato	Código	Conteúdo da trama	Dimensão (bytes)	Baud rate
A	ASCII	Velocidade mais intensa ou mais rápida	4	≥ 1200
B	ASCII	Todas as velocidades (<i>strong, faster, locked</i>) e condições em que foram detetadas	16	≥ 4800
D0	ASCII	Velocidade mais intensa e direção	5	≥ 1200
D1	ASCII	Velocidade mais intensa, direção e <i>checksum</i>	6	≥ 2400
D2	ASCII	Velocidade mais intensa no formato decimal e direção	7	≥ 2400
D3	ASCII	Velocidade mais intensa no formato decimal, direção e amplitude relativa	12	≥ 4800
D4	HEX	Velocidade mais intensa	7	≥ 2400
F0	ASCII	Velocidade mais intensa	6	≥ 2400
F1	ASCII	Velocidade mais intensa, velocidade média e direção	36	≥ 9600
F2	ASCII	Velocidade mais intensa, velocidade média numa janela definida pelo utilizador e direção	25	≥ 9600
G	ASCII	Velocidade mais intensa ou mais rápida e indicador para contagem de veículos	6	≥ 2400
Gs	ASCII	Velocidade mais intensa ou mais rápida e indicadores para contagem e intensidade	6	≥ 2400
Enhanced Output	HEX	Todas as velocidades (<i>strong, faster, locked</i>) e condições em que foram detetadas	21	≥ 4800
S	ASCII	Velocidade mais intensa e mais rápida, e condições em que foram detetadas	19	≥ 4800
BT	ASCII	Registo de data e hora do relógio interno	16	≥ 4800
DT	ASCII	Registo de data e hora do relógio interno	23	≥ 9600
DBG1	ASCII	Informação estatística de todos os veículos detetados	33 ou 39	≥ 115200
LOG	ASCII	Informação estatística dos veículos após saírem da zona de deteção	60 ou 66	≥ 115200

Tabela 4.11: Protocolos de *streaming* suportados pelo radar

Uma explicação mais detalhada dos protocolos pode ser encontrada no manual ([54]).

Os protocolos mais utilizados no âmbito do projeto foram o B, o D0 e o LOG, para os quais se desenvolveu código em *Python* para leitura das *strings*.

Uma última nota vai para o facto de ser possível gerar relatórios com todos os dados estatísticos, diretamente no *software* disponibilizado pela *Stalker*, filtrando os valores pretendidos e apresentando-os em gráficos ou tabelas. Esta é uma ferramenta muito útil e que será explorada mais à frente.

Segue-se agora um capítulo dedicado à rede neuronal responsável pela deteção de peões, cobrindo assim todos os aspetos de implementação da solução, antes de passarmos à validação e apresentação de resultados obtidos.

Capítulo 5

Rede neuronal

Aborda-se agora a estratégia empregue no âmbito da deteção de peões. Ao longo dos últimos capítulos referiu-se a utilização de uma pequena câmara e uma rede neuronal, sendo que neste momento o objetivo passa por detalhar o procedimento associado a esta tarefa.

Dedicou-se todo um capítulo a este tema por se considerar ser uma parte nuclear da solução e merecer uma análise particularizada.

Em primeiro lugar serão especificados um conjunto de conceitos básicos. Segue-se uma análise à evolução da tecnologia de deteção de objetos escolhida, tendo como principal objetivo compreender o funcionamento da sua versão mais recente. Por fim, dá-se a conhecer o procedimento de treino e implementação da rede neuronal.

5.1 Conceitos-chave

O YOLO é um sistema de deteção de objetos em tempo real, cuja função é determinar a posição dos objetos dentro de uma imagem e classificá-los. Ao contrário da maioria dos métodos conhecidos, o YOLO não se baseia numa arquitetura *pipeline* onde as tarefas são executadas por etapas, mas sim numa rede neural única. Torna-se desta forma mais rápido e pelo menos, em teoria, mais fácil de treinar e otimizar [55].

Resumidamente é utilizada uma imagem como entrada do sistema (neste caso uma fotografia) passando-a por uma rede neural semelhante a uma Convolutional Neural Network (CNN), de modo a obter-se os vetores das caixas delimitadoras e prever a classe dos objetos na saída.

5.1.1 Rede neural convolucional

Para compreender melhor a arquitetura YOLO é necessário ter algumas noções básicas sobre Artificial Neural Networks (ANN) e em particular sobre CNNs, uma subclasse deste tipo de redes, em português designadas por redes neuronais convolucionais.

Os avanços na visão por computador com recurso à aprendizagem automática, devem-se em grande parte a este tipo de algoritmos, especialmente úteis para a classificação e reconhecimento de imagens.

É exigido pouco pré-processamento em comparação com outros algoritmos de classificação, já que estas redes tem a capacidade de “aprender” os filtros/características que teriam de ser implementados manualmente num algoritmo tradicional. No entanto, para que isso aconteça,

é necessário mostrar uma enorme quantidade de imagens à rede, antes que ela seja capaz de prever objetos em imagens que nunca viu.

As CNNs dividem-se em duas partes: a extração de características, realizada nas camadas ocultas, e a classificação, realizada nas camadas totalmente conectadas, onde se determina a probabilidade de essas características serem o objeto que se previa [56].

O procedimento é o seguinte: quando uma imagem é colocada à entrada da rede, é convertida numa matriz de valores com três dimensões (largura, altura e profundidade). Segue-se então a extração das características, feita com recurso a camadas convolutivas e de *pooling*.

Na camada convolutiva é executada uma operação de convolução¹, através de um filtro que percorre toda a imagem para extrair os recursos de alto nível. É utilizado um filtro diferente em cada operação, produzindo diferentes mapas de características que são posteriormente reunidos. O esquemático da figura 5.1 ajuda a compreender o funcionamento.

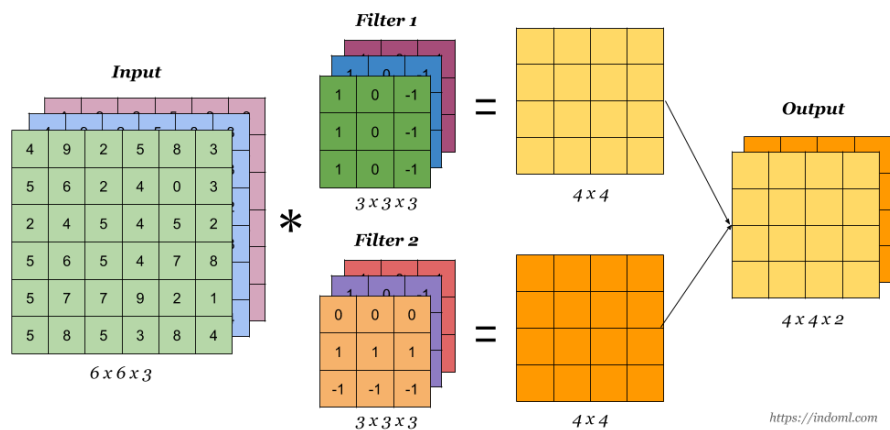


Figura 5.1: Camada convolutiva [57]

Existem 2 parâmetros que podem modificar o comportamento de uma camada convolucional, o *stride* e o *padding*. O *stride* corresponde ao número de células que o filtro se desloca de cada vez, na extração de um mapa de características. Já o *padding* consiste em adicionar zeros à volta da matriz de entrada e é o que nos permite realizar várias operações de convolução, sem que o volume à saída diminua mais do que gostaríamos.

A camada de *pooling*, por sua vez, é responsável por reduzir o tamanho espacial, de modo a diminuir a capacidade computacional necessária para processar os dados. Existem dois métodos de implementação diferentes: o *max pooling* que devolve o valor máximo, funcionando também como filtro ao ruído, e o *average pooling* que apenas retorna a média (ver figura 5.2).

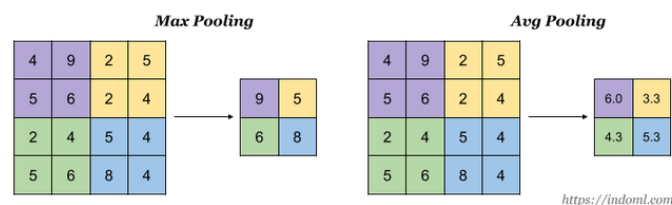


Figura 5.2: Metodologias de *pooling* [57]

¹combinação matemática de duas funções para produzir uma terceira

Aumentando o número de camadas convolucionais e de *pooling* obtêm-se mais detalhes de baixo nível, isto no entanto requer maior poder computacional, sendo necessário assumir um certo compromisso.

Segue-se a parte de classificação, onde a informação deve ser unidimensional. Convertem-se então os dados, organizando-os em vetores e atribuindo-lhes as probabilidades.

5.1.2 Precisão média

Uma das medições mais referenciadas na detecção de objetos é designada por Mean Average Precision (mAP). Surge, portanto, a necessidade de conhecer esta nomenclatura, como forma de compreender a sua importância e implicações.

O termo *precision* mede a percentagem de previsões corretas, ou seja, dentro do universo de positivos obtidos (verdadeiros ou falsos), aqueles que de facto são verdadeiros. Temos também o termo *recall* que mede o quão bem se encontram esses positivos, ou seja, aqueles que foram corretamente detetados, tendo em conta todos os elementos relevantes [58]. A figura 5.3 ajuda a compreender melhor estes conceitos, essenciais para definirmos o mAP.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \quad (5.1)$$

$$Recall = \frac{TruePositives}{TruePositives + FalseNegatives} \quad (5.2)$$

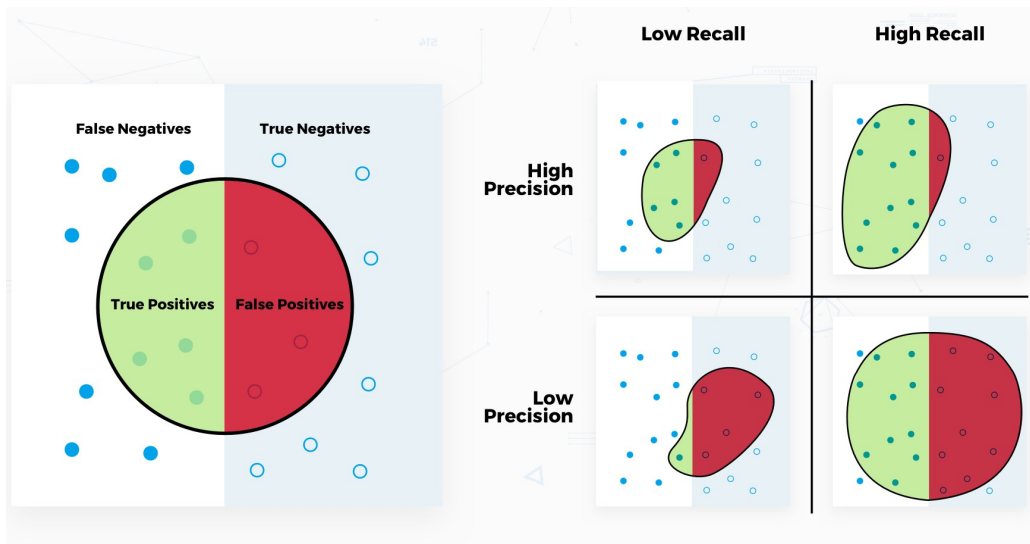


Figura 5.3: *Precision e recall* [59]

Relacionando os dois termos, define-se Average Precision (AP) como a área a baixo da curva *precision-recall*, ou seja, o integral entre 0 e 1 da função que define essa curva.

No que ao *dataset* Common Objects in Context (COCO) diz respeito, um banco de dados que será abordado mais à frente no âmbito do treino da rede neuronal, não há diferença entre os termos AP e mAP.

5.1.3 Interseção sobre união

Outra métrica usada para avaliar a detecção de objetos é chamada Intersection over Union (IoU) e mede a precisão de um detetor sobre determinado *dataset*. Qualquer algoritmo que forneça uma previsão de *bounding boxes* (caixas delimitadoras) à saída, pode ser avaliado por esta métrica.

São necessárias as caixas delimitadoras reais e as previstas pelo modelo, calculando-se a relação entre a área de sobreposição destas caixas e a área de união das mesmas, tal como mostra a figura 5.4.

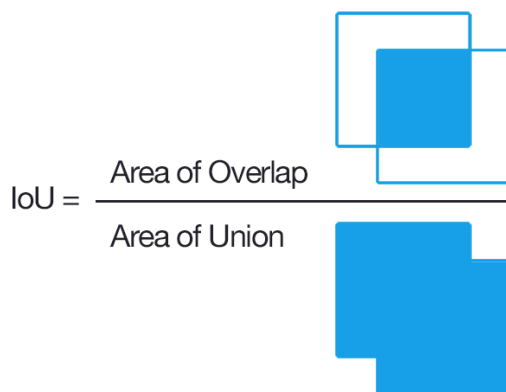


Figura 5.4: Intersection over Union [60]

Por norma, considera-se uma previsão boa se o valor de IoU for superior a 0,5.

No YOLO o conceito de IoU tem um papel fundamental, já que a confiança de uma predição é dada por: $Pr(\text{Object}) \times IoU(\text{pred}, \text{truth})$.

5.2 You Only Look Once

Vamos agora percorrer cada versão do YOLO, de modo a compreender o seu funcionamento e evolução ao longo do tempo.

Esta arquitetura prevê simultaneamente várias caixas delimitadoras e as respetivas probabilidades de classe, sendo capaz de reter informações relativas ao contexto das mesmas, já que durante o treino consegue ver as imagens completas.

A *framework* Darknet que foi escrita em C e em Compute Unified Device Architecture (CUDA), está na base desta arquitetura. O código é *open source* e suporta computação em CPU e Graphics Processing Unit (GPU) [61].

5.2.1 YOLO v1 – Unified, Real-Time Object Detection

A primeira versão possui 24 camadas convolucionais, alternando entre camadas 1x1 para reduzir o espaço dos recursos das camadas anteriores e camadas 3x3 de convolução. O pré-treino das camadas convolucionais foi realizado recorrendo ao *dataset* ImageNet1000, com imagens de resolução 224x224, duplicando-se a resolução para as tarefas de detecção. No final temos 2 camadas totalmente conectadas. A figura 5.5 esquematiza esta arquitetura.

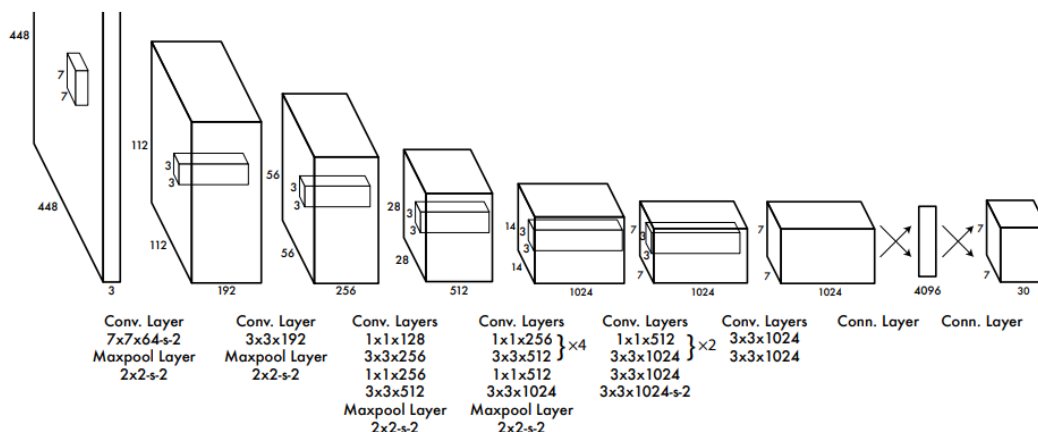


Figura 5.5: Arquitetura da primeira versão do YOLO [62]

Perante isto, o procedimento é o seguinte [62]: uma imagem de dimensões 448×448 é dividida numa grelha de $S \times S$ células e cada objeto na imagem terá uma célula responsável pela sua previsão, localizada no centro do objeto.

As células preveem um certo número de caixas delimitadoras (B), compostas por 5 componentes (x , y , w , h , confiança), onde:

- (x , y) – são as coordenadas do centro da caixa;
- (w , h) – são as dimensões da caixa;
- Confiança – varia com a IoU, como referido na secção anterior;

Se nenhum objeto for detetado, a confiança assume o valor zero nessa célula.

Depois disto, é necessário prever as probabilidades de cada classe, para as células que contêm objetos: $C = Pr(Classe(i)|Objeto)$. Adicionando estas probabilidades ao vetor de saída, o resultado é um tensor de dimensões $S \times S \times (B \times 5 + C)$.

Esta versão do YOLO quando comparada com o restante estado da arte, leva a um maior número de erros de localização, no entanto, têm uma menor probabilidade de gerar falsos positivos no *background*. A sua principal limitação está relacionada com a deteção dos objetos mais pequenos.

5.2.2 YOLO v2 – Better, Faster, Stronger

Poucos meses depois, foi lançada a segunda versão do YOLO, com o objetivo de implementar um conjunto de melhorias significativas. As principais alterações foram [63]:

- *Batch normalization*: esta técnica melhora a velocidade, performance e estabilidade, tendo sido utilizada para normalizar todas as camadas convolucionais. Isto levou a uma melhoria do mAP em 2% e reduziu o *overfitting*².
- *High resolution classifier*: a resolução das imagens para o treino com o *dataset* ImageNet, passou de 224×224 para 448×448 , o que aumentou o mAP em 4%.

²erro em novas entradas; resultado da rede memorizar o treino mas não aprender a generalizar as situações

- *Convolutional with anchor boxes*: foram removidas as camadas totalmente conectadas e utilizadas caixas de ancoragem para prever as caixas delimitadoras. Isto levou a um decaimento na precisão, mas aumentou a sensibilidade, o que significa que o modelo passou a ter mais espaço para melhorias e consegue prever mais caixas delimitadoras em simultâneo (mais de 1000, em vez de 98 que era o máximo que conseguíamos prever na versão anterior).
- *Dimension clusters*: Em vez de escolher as dimensões das caixas à mão, passou a correr *k-means* no treino para *clustering* (agrupar automaticamente os dados segundo o grau de semelhança que apresentam).
- *Direct location prediction*: Para corrigir a instabilidade do modelo, em particular nas primeiras iterações, a rede prevê agora 5 caixas delimitadoras em cada célula do mapa, com 5 coordenadas para cada caixa (tx, ty, tw, th, to). Ao restringir a previsão da localização, a parametrização torna-se mais fácil de aprender e consequentemente mais estável.
- *Fine-grained features*: As previsões são agora feitas num mapa de características 13x13. Isto permite localizar melhor objetos pequenos, mantendo a eficiência nos objetos maiores.
- *Multi-scale training*: Na primeira versão havia problemas em detetar objetos de tamanhos diferentes daqueles que haviam sido treinados. Agora a rede escolhe aleatoriamente a dimensão da imagem à entrada, entre um mínimo de 320x320 e um máximo de 608x608. Isto força a que a rede aprenda a prever corretamente em várias resoluções.

Nesta versão passou a utilizar-se uma arquitetura designada por Darknet-19, com 19 camadas convolucionais e 5 de *max pooling*.

Durante o treino foram combinadas imagens de duas bases de dados diferentes, sendo que o YOLO aprende a encontrar os objetos nas imagens recorrendo ao *dataset* COCO e aprende a classificar uma grande variedade destes objetos utilizando o ImageNet.

O resultado global foi um sistema mais rápido, capaz de detetar mais de 9000 objetos diferentes.

5.2.3 YOLO v3 – An Incremental Improvement

Esta é a versão mais recente do sistema e aquela que foi implementada nas passadeiras.

A arquitetura é agora chamada de Darknet-53, por possuir 53 camadas convolucionais. Foram adicionadas outras 53 camadas para tarefas de deteção, resultando num total de 106 camadas, como apresentado na figura 5.6. A rede continua a usar sucessivas camadas convolucionais 3x3 e 1x1, mas agora possui alguns atalhos, sendo mais potente e eficiente [64].

O aumento no número de camadas é a principal razão para as melhorias de precisão, mas levou também a uma diminuição da velocidade em comparação com a versão anterior.

As deteções passaram a ser feitas em 3 escalas diferentes que advêm da redução da amostragem das imagem de entrada em 32, 16 e 8. Essa deteção é feita em três locais distintos da rede, aplicando um *kernel* de dimensão $1 \times 1 \times (B \times (5 + C))$ [65].

Tomando como exemplo uma imagem de resolução 416x416 e tendo a rede sido treinada com o *dataset* COCO (B=3 e C=80), temos um *kernel* de dimensão 1x1x255, de onde resulta na primeira deteção, realizada na camada 82, um mapa de características de dimensões

13x13x255. As próximas detecções realizam-se nas camadas 94 e 106, produzindo mapas de características 26x26x255 e 52x52x255, respetivamente.

A camada 82 é responsável por detetar os objetos maiores, a camada 94 os intermédios e a camada 106 os objetos mais pequenos.

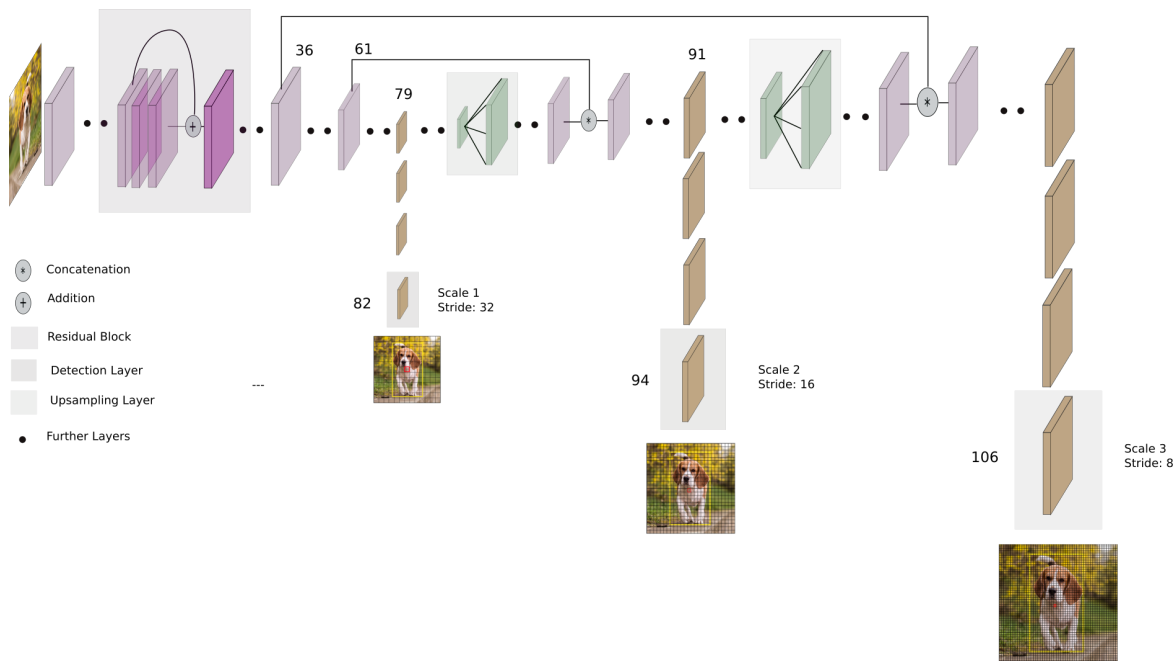


Figura 5.6: Arquitetura da terceira versão do YOLO [65]

As classes são agora previstas utilizando uma classificação multi-label, em vez de se optar pela classe com classificação superior. O método anterior assumia que as classes eram mutuamente exclusivas, ou seja, que um objeto pertencendo a uma classe não poderia pertencer a outra, o que é falso, já que um objeto pode ser associado a duas ou mais classes, dependendo do *dataset* utilizado.

A classificação passou a ser feita recorrendo a regressões logísticas³ e definindo um *threshold* a partir do qual as classes com pontuação superior são atribuídas à caixa delimitadora correspondente.

As *bounding boxes* utilizam o mesmo método que a classes, mas neste caso é atribuído o valor 1 à caixa que mais se sobrepõe ao objeto, para que daqui resulte apenas uma caixa por objeto. São previstas, no entanto, dez vezes mais caixas do que na versão anterior, muito por culpa das previsões serem realizadas em três escalas diferentes.

Quando comparado com outros detetores, verificamos que em termos de mAP esta versão do YOLO consegue ser tão exata quanto o método Single Shot MultiBox Detector (SSD) e as suas variantes, ficando no entanto atrás do RetinaNet, como comprova a tabela 5.1 (adaptada de [64]).

É importante notar, contudo, que apesar do RetinaNet ser normalmente mais preciso, demora cerca de 3,8 vezes mais tempo para processar uma imagem, como se verifica nos gráficos da figura 5.7.

³técnica que permite estimar a probabilidade da ocorrência de um evento, onde a variável dependente é de natureza binária

Designação		AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
SSD513	ResNet-101-SSD	31,2	50,4	33,3	10,2	34,5	49,8
DSSD513	ResNet-101-DSSD	33,2	53,3	35,2	13,0	35,4	51,1
RetinaNet	ResNet-101-FPN	39,1	59,1	42,3	21,8	42,7	50,2
RetinaNet	ResNeXt-101-FPN	40,8	61,1	44,1	24,1	44,2	51,2
YOLOv2	Darknet-19	21,6	44,0	19,2	5,0	22,4	35,5
YOLOv3 (608x608)	Darknet-53	33,0	57,9	34,4	18,3	35,4	41,9

Tabela 5.1: Comparação da precisão de diferentes detetores

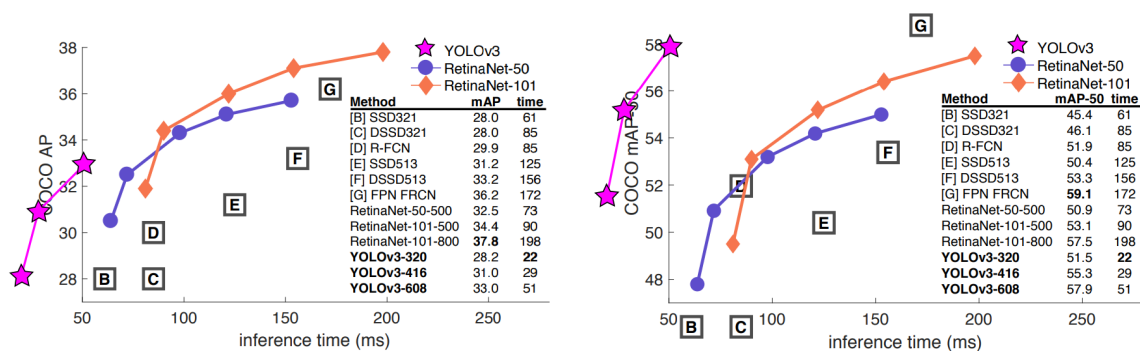


Figura 5.7: Velocidade/precisão de diferentes métodos - COCO AP (média e 0,5 IoU) [64]

5.3 Deteção em tempo real

Conhecidos os conceitos e o funcionamento da rede, podemos finalmente passar à sua implementação. Inicialmente, por uma questão de simplicidade, foi utilizado o modelo previamente treinado e disponibilizado pelos criadores do YOLO, seguindo os passos presentes no seu *website* [66]. Posteriormente, foram realizadas as alterações necessárias para otimizar os resultados.

Sabemos à partida que o modelo descrito na secção anterior é muito pesado para implementar numa *Raspberry Pi*, pelo que vamos configurar e testar também uma versão mais leve chamada “YOLOv3-tiny”.

O YOLOv3-tiny nada mais é do que uma versão simplificada do YOLOv3. É muito mais pequena do que a original, o que significa que ocupa menos memória e tem menos necessidades de hardware, permitindo o seu uso em ambientes mais restritos. Sendo esta uma rede mais rápida, é também, obviamente, menos precisa.

Apresenta-se agora o procedimento realizado de forma resumida e cronológica.

5.3.1 Implementação em Windows

As configurações e testes que se seguem foram efetuados numa máquina com as características apresentadas na tabela 5.2.

O primeiro passo foi a instalação da *framework* Darknet. Em seguida fez-se o *download* dos ficheiros com pesos pré-treinados e foram aplicadas diferentes imagens ao detetor.

Sistema operativo	Windows 8.1, 64-bit
Processador	Intel Core i7-4710HQ, 2.5GHz
Memória RAM	8GB
Placa gráfica	NVIDIA GeForce GTX 850M

Tabela 5.2: Características da plataforma para teste da rede neuronal

O sistema mostrou-se capaz de distinguir entre as diferentes classes, mas o CPU demorava vários segundos para processar uma imagem, além de que os resultados não eram apresentados automaticamente.

Tornou-se evidente a necessidade de instalar *software* adicional que permitisse otimizar o processo de detecção, como o OpenCV para apresentar os resultados da detecção e a plataforma CUDA para computação paralela por GPU (CUDA Toolkit V9.0, CuDNN V7.05).

Como o processo de compilação da Darknet com recurso ao GPU pode ser bastante penoso, simplificou-se o procedimento seguindo parcialmente o tutorial desenvolvido pela “Augmented Startups” [67] que implementa a última versão do YOLO e utilizando o PyTorch no Windows (Anaconda – Python 3.7) [68].

Depois de todas as configurações, correu-se de novo o detetor num conjunto de imagens de teste. As figuras 5.8 e 5.9 são exemplos dos resultados obtidos para o modelo YOLOv3 e YOLOv3-tiny.

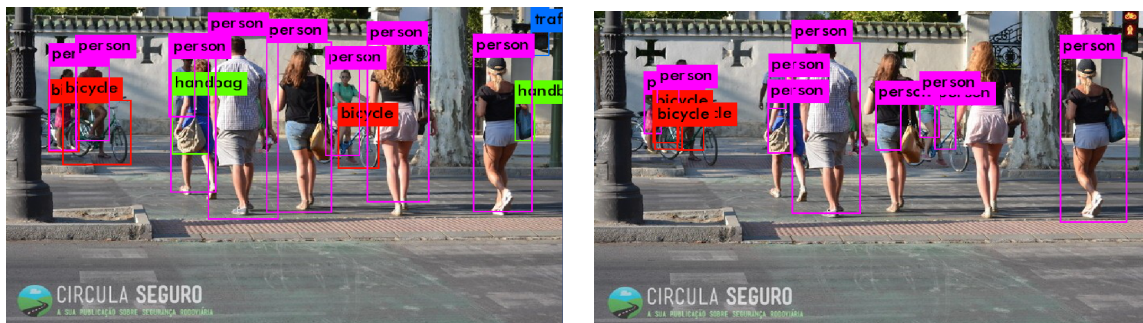


Figura 5.8: Exemplo nº1 - YOLOv3 (esquerda) vs YOLOv3-tiny (direita)



Figura 5.9: Exemplo nº2 - YOLOv3 (esquerda) vs YOLOv3-tiny (direita)

Além das imagens, testou-se também com sucesso a detecção em tempo real em arquivos de vídeo e através de uma *webcam*.

A velocidade do processamento depende de múltiplos fatores, incluindo a quantidade de objetos que a rede neuronal tem de processar. No entanto, a título de curiosidade, são exibidos na tabela 5.3 os resultados obtidos em termos de Frames Por Segundo (FPS).

	Ficheiro de vídeo (FPS)	Webcam (FPS)
YOLOv3	5,4	7,5
YOLOv3-tiny	21,3	19,6

Tabela 5.3: Velocidade média de processamento da rede neuronal

5.3.2 Implementação em Raspberry Pi

Testadas as capacidades da rede neuronal, vamos então passar à implementação na placa de desenvolvimento do protótipo. As características mais importantes da máquina que temos em mãos são as seguintes:

Sistema operativo	Raspbian, 64-bit
Processador	Broadcom BCM2837B0, Cortex-A53, 1.4GHz
Memória RAM	1GB, LPDDR2 SDRAM

Tabela 5.4: Características da *Raspberry Pi 3B+*

Como já foi referido, a *Raspberry* não é capaz de processar a rede completa, tornando-se mandatório o uso da versão *tiny*. Na base do código desenvolvido está o artigo “*Deep Learning with Raspberry Pi – Real-time object detection with YOLO v3 Tiny*” [69].

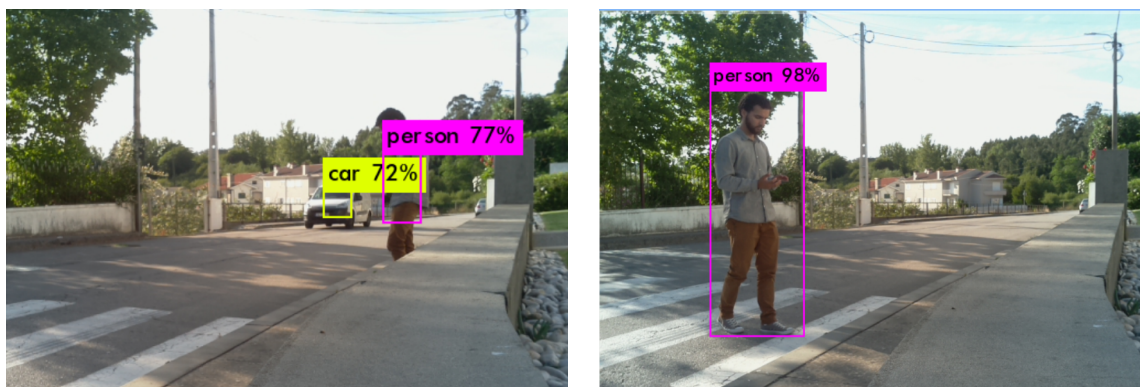


Figura 5.10: Exemplos de funcionamento do YOLOv3-tiny em *Raspberry Pi*

Foi instalada a biblioteca NNPACK com o objetivo de otimizar a Darknet. Esta implementação permite um desempenho cerca de 40% mais rápido, após execução das primeiras

frames num CPU multi-core [69], uma grande vantagem, tendo em conta que neste projeto é necessário processar imagens de forma contínua.

A par com esta biblioteca utilizou-se também uma *framework* Python chamada “PeachPy” [70] e o sistema de compilação “Ninja” [71].

Seguiu-se a configuração da câmara e o teste da rede em imagens, fotografias e vídeo, tendo-se obtido para este último resultados em torno de 1 FPS, como seria de esperar. Na figura 5.10 temos dois exemplos de deteção por câmara, no local onde o STAP será instalado.

5.3.3 Otimização da rede neuronal

Nesta fase, a utilização desta rede para detetar peões na via pública de forma contínua, ainda é praticamente impossível. A complexidade dos cenários que podem ser detetados pelas câmaras, levariam não só a resultados pouco viáveis, como poderiam até fazer com que a placa deixasse de responder.

Para resolver este problema é necessário treinar a nossa própria rede neuronal. Todas as configurações que se seguem foram realizadas num computador com sistema operativo Ubuntu e que tal como a máquina apresentada na implementação em Windows, permite computação paralela, utilizando a extensão de linguagem CUDA da Nvidia.

A alteração mais simples que podemos fazer ainda antes treinar a rede, é alterar alguns parâmetros de forma a detetar apenas 1 classe, em vez das 80 que a versão *tiny* do YOLO consegue detetar. Isto não fará com que a rede se torne obrigatoriamente mais precisa, mas ficará pelo menos mais rápida.

Modifica-se então o ficheiro com a lista de nomes das classes, eliminando todas as entradas à exceção da primeira (*person*). Em seguida corrige-se o ficheiro *yolov3-tiny.cfg*, alterando o número de classes e as dimensões dos filtros das camadas convolucionais que surgem antes das camadas designadas por [yolo]. Originalmente estes filtros possuem o valor 255 e passam assim para 18, considerando a equação 5.3.

$$Filtros = (n_{classes} + 5) \times 3 \quad (5.3)$$

Depois de testar e confirmar que as alterações foram bem sucedidas, temos finalmente o sistema a detetar apenas pessoas e podemos passar à próxima fase que será treinar a rede especificamente para este objetivo.

O primeiro passo é escolher um bom *dataset*. Como existem diversos *datasets* muito completos no que diz respeito à deteção de pessoas, não faria sentido nenhum criar um de raiz, até porque no contexto desta dissertação, não haveria tempo suficiente para selecionar tantas imagens quantas aquelas que os *datasets* atuais fornecem e criar as respetivas anotações⁴.

Assim sendo, realizou-se uma pesquisa sobre *datasets* de pessoas, focando em particular naqueles que se dizem melhores para deteção de pedestres. Encontraram-se múltiplas alternativas, das quais se destacam as da tabela 5.5.

Excluíram-se o *CityPersons* e o *KITTI* porque se considerou que o número de amostras era muito baixo e dificilmente levariam a resultados melhores do que aqueles que já se tinham obtido. Os restantes foram considerados como hipóteses viáveis, tendo sido estudados em maior profundidade.

⁴ficheiro de texto onde são armazenadas as coordenadas das caixas delimitadoras e a classe dos objetos

	CityPersons	KITTI	CrowdHuman	Caltech	COCO (<i>Persons</i>)
Imagens	2.975	3.712	15.000	42.782	64.115
Pessoas	19.238	2.322	339.565	13.674	257.252
Regiões ignoradas	6.768	45	99.227	50.363	5.206
Pessoas/imagem	6,47	0,63	22,64	0,32	4,01

Tabela 5.5: Comparação de diferentes *dataset* de pedestres [72]

Tanto o *CrowdHuman* como o *COCO* fornecem as anotações no formato JavaScript Object Notation (JSON), incompatível com a Darknet, pelo que precisam de ser convertidos em ficheiros de texto que cumpram com o formato: <classe> <x> <y> <w> <h>.

Já o *Caltech Pedestrian* precisa não só de converter as anotações do formato “.vbb” para “.txt”, como também os ficheiros de vídeo “.seq” nas respetivas *frames* “.png”.

Todas as conversões referidas são possíveis [73, 74, 75], pelo que não levaram à exclusão de nenhuma das hipóteses.

No entanto, percebendo que a complexidade das 15 mil imagens do *CrowdHuman* não correspondem à realidade observada nas maiorias das passadeiras em Portugal, decidiu-se excluir esta alternativa. A situação inversa ocorre no *Caltech*, já que apresenta uma média de apenas 0,32 pessoas por imagem. A escolha acabou assim por recair no *dataset COCO*, por parecer o mais equilibrado dos três e possuir situações muito diversificadas.

Fez-se então o *download* da última versão (COCO 2017) no site oficial, num total de mais de 123 mil imagens, das quais 118 mil (19,3GB) são para treino e 5 mil (815MB) para validação.

Converteram-se as anotações seguindo o procedimento descrito na publicação “Getting Started with Darknet YOLO and MS COCO for Object Detection” [76], obtendo-se os ficheiros de texto que correspondem às 64.115 imagens com a classe *person*.

As imagens e as anotações foram então colocadas junto dos ficheiros da *darknet*, ficando apenas em falta a especificação da localização das mesmas e a configuração de algumas variáveis. Os parâmetros mais importantes a ajustar são os seguintes [77]:

- **Batch:** número de imagens processadas pelo GPU em cada iteração. Foi definido o valor 64, o que significa que a cada 64 imagens os parâmetros da rede neuronal são atualizados.
- **Subdivisions:** este valor surge porque o GPU pode não ter memória suficiente para processar o *batch* de uma só vez. Definiu-se o valor 16 porque valores inferiores (1, 2, 4 e 8 - devem ser sempre múltiplos de 2) levariam a um erro por falta de memória. Desta forma o GPU processa $64/16 = 4$ imagens de cada vez, até que a iteração esteja completa, perfazendo as 64 imagens.
- **Width, height, channels:** estes parâmetros especificam o tamanho para o qual as imagens serão redimensionadas antes de serem treinadas. O *channel* será obrigatoriamente 3 porque se trata de imagens RGB. Já para a largura e a altura, sabemos que quanto maior o tamanho, mais precisa será a rede e mais lenta de treinar, pelo que inicialmente tentou-se utilizar 608x608 para privilegiar a precisão.

Devido a limitações de memória não foi possível completar o treino com esta resolução. Uma alternativa seria manter a resolução e aumentar as subdivisões para 32, mas como

pequenos *minibatches*⁵ levam a uma diminuição da velocidade e da qualidade dos resultados, decidiu-se manter as subdivisões em 16 e utilizar uma resolução de 416x416.

- **Max Batches:** É o número total de iterações do processo de treino. Depende do número de classes da rede e aconselha-se que seja igual a $n_{classes} \times 2000$, mas nunca menor que 4000. Como só estamos a treinar uma classe, definiu-se o valor 4000.
- **Random:** Trata-se de uma *flag* que quando está ativa permite treinar o YOLO em diferentes resoluções. Foi colocada a 1 para assim melhorar a capacidade de deteção.

Uma nota para o facto de que os valores *batch*=64 e *subdivisions*=16 dizem respeito apenas ao treino, sendo-lhes atribuído o valor 1 durante os testes. Existem ainda outros parâmetros do ficheiro de configurações que podem ser ajustados, como por exemplo o *learning rate*, mas que não serão aqui abordados para não alongar a explicação.

Já durante o treino, se forem observadas entradas Not a Number (NaN) no campo *average loss*, significa que algo correu mal. Se essas entradas surgirem apenas em algumas linhas que não esta, o treino está a decorrer normalmente [78].

Assim que o treino terminar deve-se escolher o ficheiro “.weights” que apresente menor *average loss*. Por vezes pode não ser o último ficheiro produzido, caso ocorra *overfitting*.

Treinada a rede neuronal, podemos implementa-la no sistema e passar para a fase de testes e validação dos resultados.

⁵resultado da relação *batch/subdivisions*

Capítulo 6

Validação e resultados

Neste momento, todas as etapas de implementação foram cumpridas e portanto, é chegada a hora de avaliar a solução.

O objetivo desde capítulo não é só dar a conhecer os resultados alcançados, mas também procurar falhas e aspetos que possam ser melhorados numa próxima versão do protótipo.

Para que isso aconteça e se torne mais simples de transmitir os dados e as respetivas conclusões, dividiu-se a explicação em duas partes. Na primeira parte são descritos os testes realizados, desde o procedimento seguido até aos resultados obtidos. Na segunda parte analisam-se esses resultados, procurando realizar uma análise crítica e construtiva.

Será também utilizada uma abordagem crescente em complexidade, no sentido em que inicialmente serão efetuados testes parciais a cada tecnologia e por fim integrar-se-ão os diversos subsistemas, validando a solução como um todo.

6.1 Testes realizados

Vamos então começar por apresentar os testes que foram realizados, separando-os de acordo com a tecnologia que utilizam. São três tópicos que constituem esta secção. O primeiro diz respeito à câmara e à rede neuronal para detetar peões, o segundo ao radar para análise do tráfego rodoviário e o terceiro aos sistemas de iluminação, tanto de presença como de sinalização.

6.1.1 Detecção de peões

Para percebermos se o treino da rede neuronal produziu resultados desejáveis, nada melhor do que correr um conjunto de imagens de validação, ou seja, imagens em que se conheça à partida os objetos presentes e a sua localização. Neste caso em particular, precisamos apenas de saber a posição real das pessoas nas imagens (classe *person*). Conhecendo essas anotações, podemos recorrer aos conceitos teóricos apresentados no capítulo 5 e calcular valores de precisão, sensibilidade, mAP e IoU, para avaliar a eficácia da deteção de pessoas.

A validação é preferencialmente realizada com imagens diferentes daquelas com que se treinou a rede e, portanto, optou-se por utilizar as 5 mil imagens que o *dataset* COCO 2017 fornece especificamente para o efeito. No entanto, a maioria dos *datasets* são obtidos através de *scripts* de conversão (como aconteceu neste caso), e por vezes, as próprias anotações são

estabelecidas através de ferramentas de terceiros, pelo que é frequente surgirem objetos mal classificados e caixas delimitadoras desajustadas [78].

Decidiu-se então verificar as imagens de validação, recorrendo a uma Graphical User Interface (GUI) [79] que permite marcar/alterar as caixas delimitadoras dos objetos, gerando as anotações correspondentes. O objetivo de tal verificação passa por corrigir os erros mais flagrantes, garantindo que as anotações “reais” são o mais corretas possíveis, para que possam posteriormente ser comparadas com as obtidas pela rede neuronal.

Encontraram-se no *dataset* imagens muito diversas, sendo que se, por um lado, temos casos bem classificados e muito semelhantes aqueles que esperamos encontrar na realidade (ver figura 6.1), por outro, eram tão ou mais comuns os casos em que as anotações estavam em falta ou incorretas, muito por conta da presença de multidões ou plateias (ver figura 6.2).

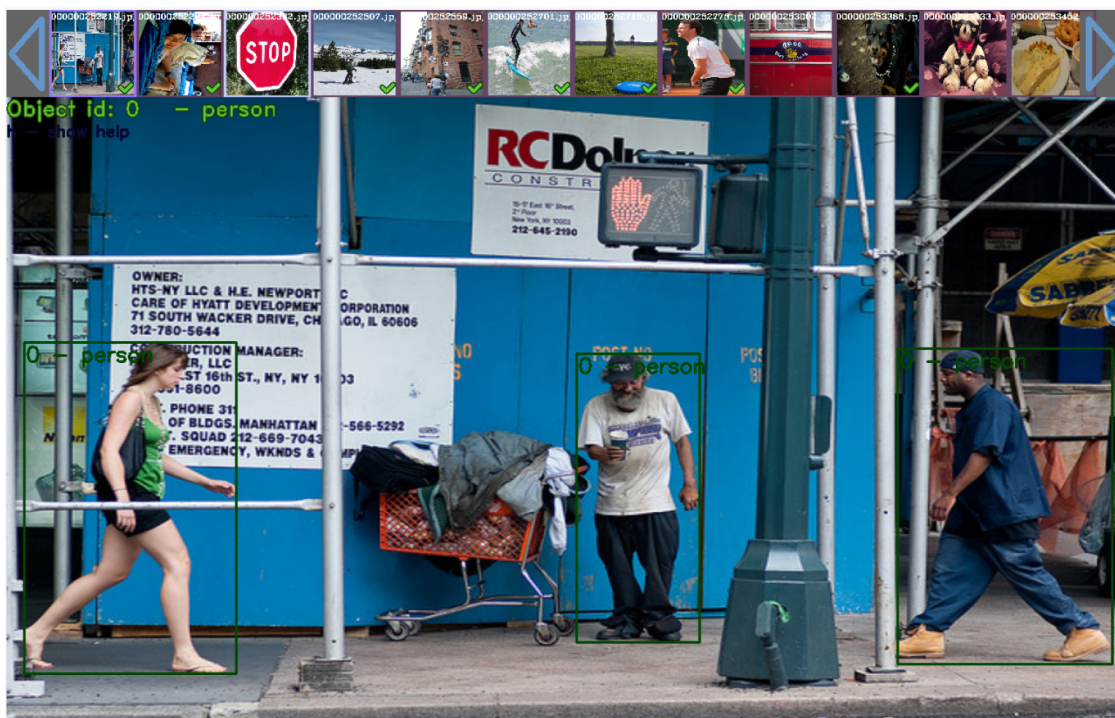


Figura 6.1: Exemplo de anotação correta de objetos da classe *person*

Corrigiu-se então parcialmente as anotações, eliminando as caixas delimitadoras que surgiam em posições erradas e criando novas em torno das pessoas que não estavam sinalizadas.

Foram também removidas 34 imagens devido à sua complexidade. Tratavam-se de enormes multidões onde as anotações não faziam o menor sentido e que levariam imenso tempo a corrigir por completo, sem trazer grandes vantagens em termos práticos para a avaliação da capacidade de detetar pedestres.

Com o *dataset* melhorado, podemos finalmente avaliar as capacidades da rede neuronal. Decidiu-se realizar uma análise comparativa que permitisse facilmente perceber se o treino descrito na secção 5.3.3, levou a uma otimização ou a um decaimento das capacidades da rede, relativamente à versão pré-treinada disponibilizada pelos criadores¹.

¹YOLOv3-tiny 608x608

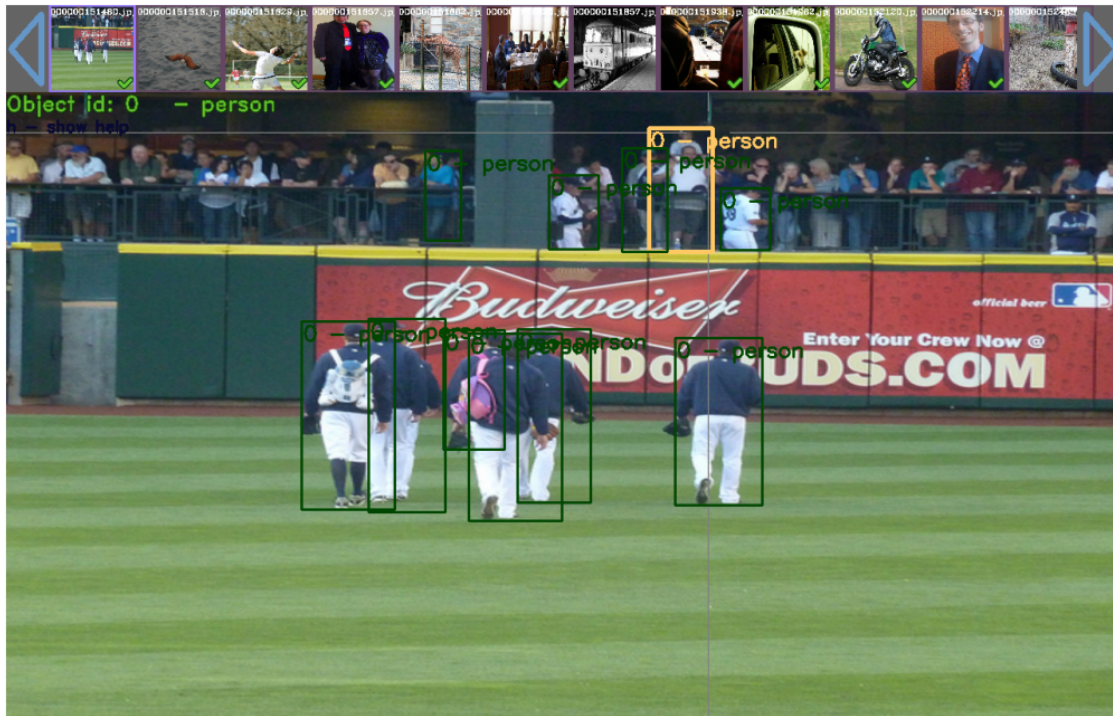


Figura 6.2: Exemplo de anotação parcialmente errada de objetos da classe *person*

A avaliação do desempenho das duas versões será efetuada recorrendo ao código presente no repositório [80] que segue os critérios mAP definidos na competição “PASCAL Visual Object Classes (VOC) 2012”. O procedimento é o seguinte:

- 1) Os resultados são ordenados por ordem decrescente de confiança e a sua localização comparada com a das anotações;
- 2) Se a previsão e a anotação forem classificadas com a mesma *label* e a IoU for maior ou igual a 50%, a deteção é considerada correta. Apenas é possível atribuir uma previsão por anotação, evitando assim deteções múltiplas;
- 3) Obtém-se a curva precisão/sensibilidade;
- 4) Calcula-se por integração numérica a área por baixo da curva, de modo a obter o mAP;
- 5) Por fim, o número de previsões por classe (verdadeiras e falsas) é representado num histograma horizontal.

Assim sendo, aplicaram-se à entrada da rede original e da versão pós-treino as 4.966 imagens e foram guardados os resultados das suas previsões. Por *default* apenas objetos detetados com uma confiança igual ou superior a 25% são armazenados, sendo que os restantes foram descartados. A resolução das duas redes foi definida como 608x608 nos respetivos ficheiros “.cfg”, com o objetivo de aumentar a precisão [78].

Comparando os resultados com as anotações verdadeiras, obtiveram-se para a classe *person*, as curvas precisão/sensibilidade representadas na figura 6.3.

Calculando o mAP obteve-se 17,15% para a versão original e 37,32% para a versão treinada, como se observa nos gráficos da figura 6.4.

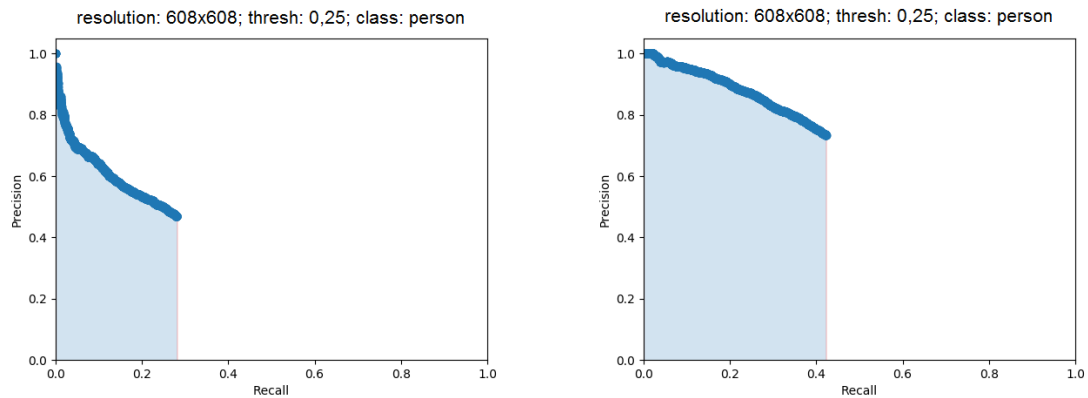


Figura 6.3: Curva *precision/recall*: YOLOv3-tiny (esquerda) e versão melhorada (direita)

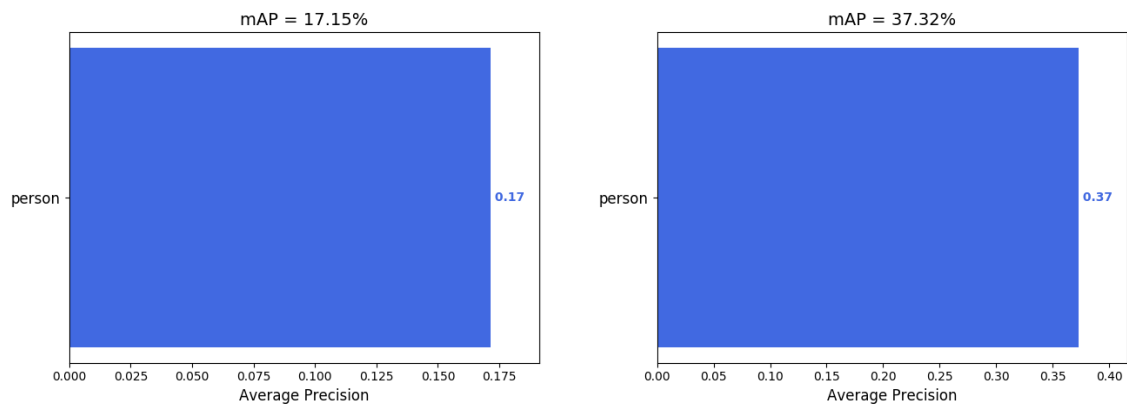


Figura 6.4: mAP para a classe *person*: YOLOv3-tiny (esquerda) e versão melhorada (direita)

Em termos do número de pessoas presentes no *dataset*, eram esperadas 10.875 detecções, distribuídas por 2.659 das 4.966 imagens (figura 6.5).

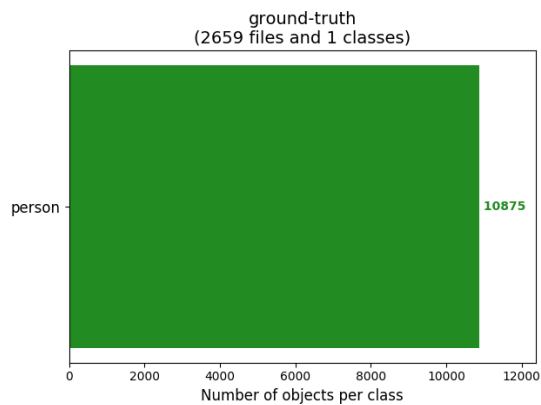


Figura 6.5: Número de objetos a detetar da classe *person*

Este seria o caso ideal, considerando que o *dataset* não possui nenhum erro nas caixas delimitadoras e que todas as detecções teriam uma confiança superior a 0,25 e $\text{IoU} \geq 50\%$. Como sabemos que essas condições nem sempre se vão verificar e que o próprio *dataset* apresenta erros, os resultados reais são os apresentados na figura 6.6.



Figura 6.6: Objetos detetados: YOLOv3-tiny (esquerda) e versão melhorada (direita)

Além da diferença no número de detecções entre as duas redes, é possível observar que a rede fornecida pelos criadores do YOLO detetou 77 das 80 classes para as quais está programada. Como só nos interessa comparar a capacidade de detetar pessoas, essas detecções foram filtradas, apresentando-se apenas os resultados que dizem respeito à classe pretendida.

Ao executar a rede para se obter as previsões, também é possível definir o *threshold* referente à confiança. Este parâmetro pode ser alterado na linha de comandos ao incluir a *flag* “-thresh <valor>”. Para termos uma ideia do efeito que uma alteração destas produz, repetiu-se o processo anterior para uma confiança igual ou superior a 10%, mantendo os restantes parâmetros.

Os resultados podem ser observados nas figuras 6.7, 6.8 e 6.9, onde se apresentam as curvas precisão/sensibilidade, os valores de mAP e os resultados da detecção, respetivamente.

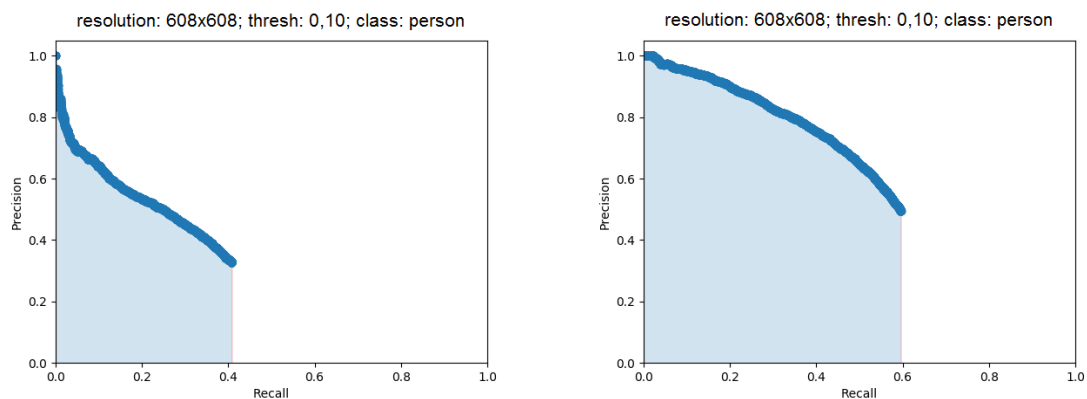


Figura 6.7: Curva *precision/recall*: YOLOv3-tiny (esquerda) e versão melhorada (direita)

Por fim, a tabela 6.1 resume o desempenho das duas redes.

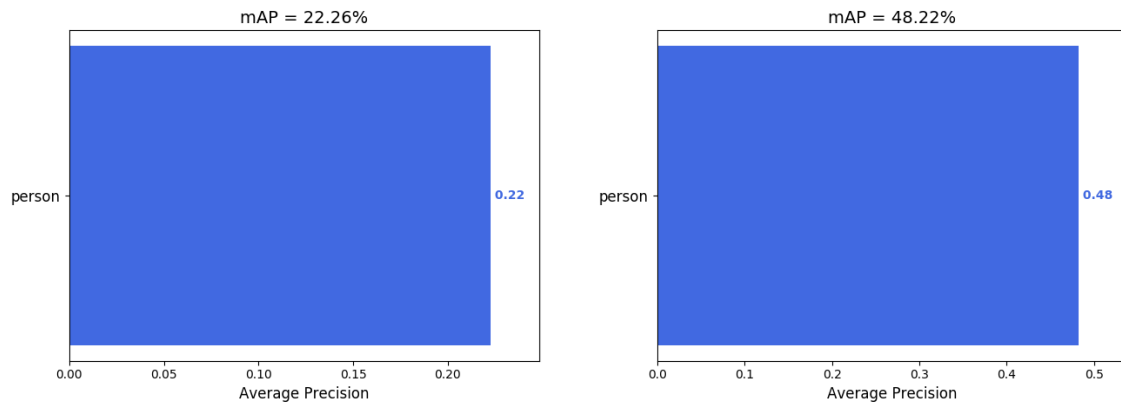


Figura 6.8: mAP para a classe *person*: YOLOv3-tiny (esquerda) e versão melhorada (direita)



Figura 6.9: Objetos detetados: YOLOv3-tiny (esquerda) e versão melhorada (direita)

	YOLOv3-tiny thresh = 0,25	YOLOv3-tiny thresh = 0,10	My network thresh = 0,25	My network thresh = 0,10
Deteções acertadas	3.058	4.437	4.595	6.479
Falsas deteções	3.469	9.085	1.662	6.633
mAP (%)	17,15	22,26	37,32	48,22

Tabela 6.1: Resumo do desempenho da rede neuronal no âmbito da deteção de pessoas

A avaliação da rede de acordo com estas métricas de precisão é importante, porque permite prever o seu comportamento em diversos cenários. Mas tão ou mais importante do que estes cálculos, é observar o seu funcionamento na prática. No nosso caso, é fundamental testar a deteção através da *Raspberry Pi*, para perceber se esta é capaz de produzir bons resultados e aguentar a carga computacional em funcionamento contínuo.

A placa de desenvolvimento ficou a executar o programa que implementa a rede durante vários dias, para comprovar a sua estabilidade. Foram efetuadas deteções *indoor* e *outdoor*, com baixa e elevada luminosidade (dia e noite), a curta e longa distância, dando prioridade

às deteções *outdoor* a curta distância.

Um exemplo de deteção da versão melhorada da rede neuronal, implementada no sistema STAP e como tal, processada através da *Raspberry Pi*, pode ser observado na figura 6.10.

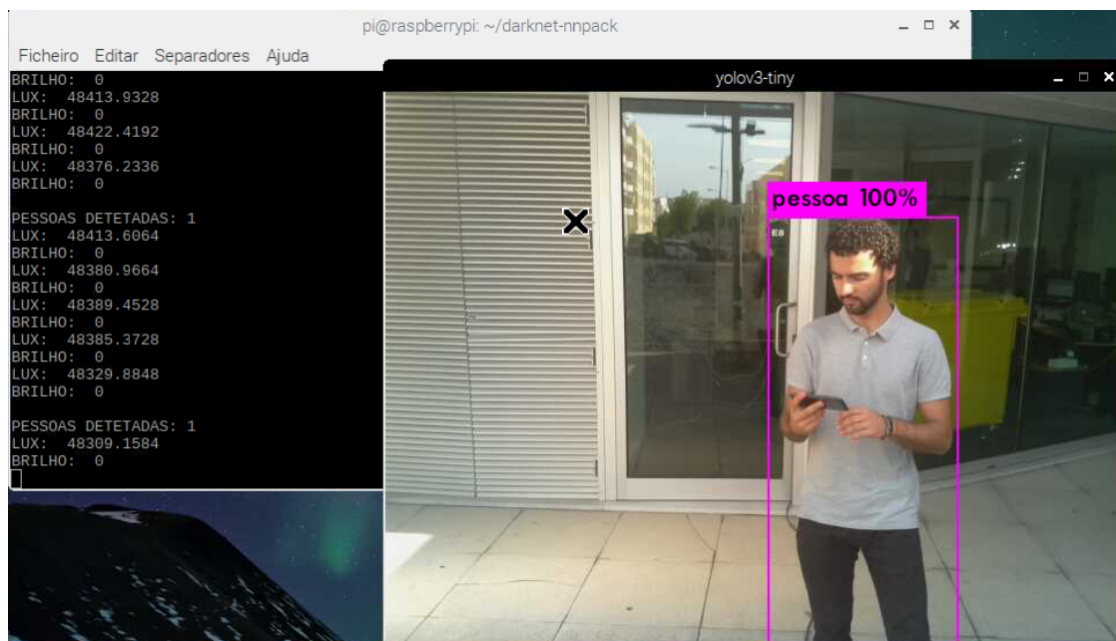


Figura 6.10: Exemplo de funcionamento da versão melhorada da rede neuronal

6.1.2 Análise de tráfego

Relativamente ao radar, decidiu-se numa primeira fase testar globalmente as suas capacidades. Para isso privilegiou-se a utilização do *software* da *Stalker*, tanto para a configuração como para a recolha de dados, por uma questão de simplicidade. São quatro os programas disponibilizados, cujas funcionalidades primordiais se descrevem em seguida:

- **Dashboard application:** Apresenta as configurações básicas presentes no radar e as velocidades detetadas em tempo real (*strong*, *fast*, *lock* e *vehicle*).

É uma ótima opção para fazer um *fork test*, ou seja, utilizar o *tuning fork* presente no *kit* de desenvolvimento para garantir o funcionamento adequado do sensor de velocidade. Certificado de acordo com os *standards* do National Institute of Standards and Technology (NIST), emite um sinal de calibração a uma frequência de 2.899 ± 5 Hz (21°C). O objetivo é simular o efeito de Doppler criado por um objeto em movimento e será lido por um radar K-Band² a funcionar a 24,15 GHz, como 40 milhas por hora (64 km/h).

- **Sensor Wizard:** Permite alterar todas as configurações do sensor, desde as mais básicas até às mais avançadas.

Se o objetivo for configurar o radar para funcionar com algum protocolo de *streaming*, ou melhorar o desempenho quando é instalado num local que não é o ideal, este é o

²definido pelo Institute of Electrical and Electronics Engineers (IEEE) como a gama de frequências entre os 18 e 27 GHz

programa que devemos explorar. As alterações devem ser efetuadas com cuidado, visto que as configurações avançadas são destinadas apenas a utilizadores experientes e podem resultar numa diminuição da precisão do sensor e/ou falhas na recolha de dados.

- **Easy Analyst:** Ferramenta fácil de utilizar que permite importar os dados estatísticos e manusear a informação para gerar relatórios (gráficos e tabelas).

É compatível com o “Traffic Analyst”, facilitando a gestão dos dados recolhidos. Possibilita a inserção de imagens, a especificação da localização (endereço, coordenadas Global Positioning System (GPS), vista do Google Maps) e a alteração no modo como os dados são apresentados, já que disponibiliza 7 formatos de gráficos *standard*.

- **Traffic Analyst:** Aplicação mais completa de todas e que tem como principal objetivo a recolha de dados estatísticos. Além de permitir configurar parcialmente o sensor, apresenta uma série de funcionalidades adicionais que serão muito utilizadas.

É possível atualizar o relógio interno do sensor consoante o relógio do computador; utilizar a opção “Real Time Recording” para que o programador tenha uma representação visual do tráfego à medida que este é detetado, e possa confirmar se o sensor está bem configurado antes de o deixar a recolher dados por um longo período de tempo; criar um novo *survey*³ para ser armazenado na memória interna do sensor; e fazer *download* dos dados de um *survey* para o computador, depois da informação ter sido coletada.

Criou-se então um *survey* específico, de modo a deixar o radar a trabalhar em *standalone* por algum tempo, recolhendo posteriormente os dados para análise - figura 6.11. Este teste decorreu no dia 28/05/2019 entre as 13:42h e as 19:30h.

Survey Details		
Survey Location	Rua Além do Rio, 4505-562, Santa Maria da Feira,	
Survey GPS Coordinates	40.95°N, 8.48°W	
Survey Dates	Start	Stop
	Tuesday, 28/05/2019 13:42	Tuesday, 28/05/2019 19:30
Posted Speed Limit	50 (km/h)	
Traffic Zone	Normal	
File Name	SurveyDataFile@2019-05-28_19-37-40.xml	
Device Type	000	
Device Serial Number	SNAA000000	
Survey Note	Survey from home	
Total Vehicle Count	1285	
<i>Traffic Direction</i>	Undefined (Closing)	Undefined (Away)
<i>Vehicle Count</i>	1285	0

Figura 6.11: Descrição do *survey*

O primeiro passo foi colocar o radar junto à passadeira da figura 4.20, instalado a uma altura de 2,20m, afastado 1,45m da linha de cedência de passagem e 2,40m da zona de passagem. Seguiram-se as configurações do relógio e da memória interna do sensor.

Sabendo que o limite de velocidade naquela estrada é de 50 km/h, foi dada uma margem de 5 km/h, considerando assim que uma velocidade é excessiva se for igual ou superior a 55 km/h. Velocidades inferiores a 10 km/h não serão consideradas.

³conjunto de configurações e especificações definidas com o objetivo de coletar dados

As condições atmosféricas quando se deu início ao teste eram de céu limpo, 26°C, 0% de precipitação e 34% de humidade. Foram considerados apenas os veículos que se deslocavam no sentido de aproximação da passadeira, obtendo-se os resultados das figuras 6.12 a 6.17.

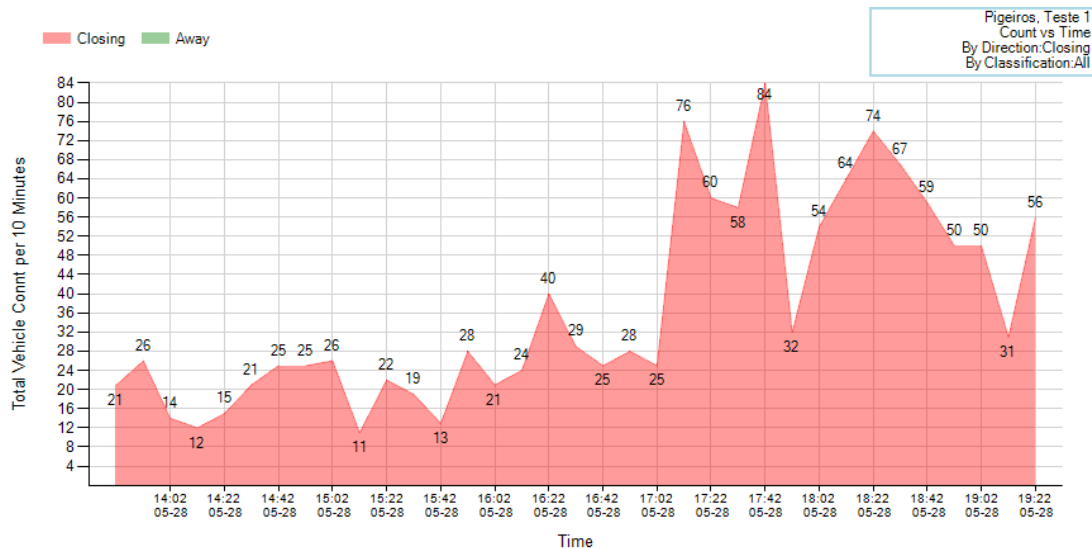


Figura 6.12: Teste ao radar - Contagem de veículos

Na figura 6.12 podemos observar o número de veículos que passaram pela passadeira no sentido sul-norte, a cada 10 minutos.

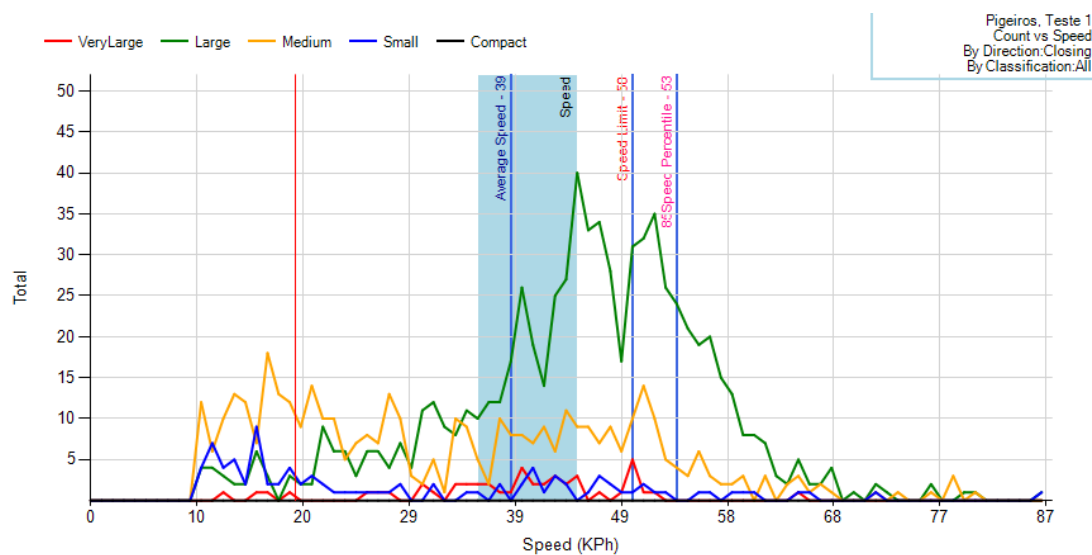


Figura 6.13: Teste ao radar - Velocidades mais frequentes por tipo de viatura

Já a figura 6.13, permite perceber quais foram as velocidades mais frequentes de acordo com a classificação atribuída a cada viatura. Neste contexto, a classe 1 corresponde aos veículos maiores (*very large*) e a classe 5 aos menores (*compact*). A divisão é feita consoante a intensidade do sinal de retorno.

Passamos agora para as figuras 6.14 e 6.15 que dizem respeito aos limites de velocidade.

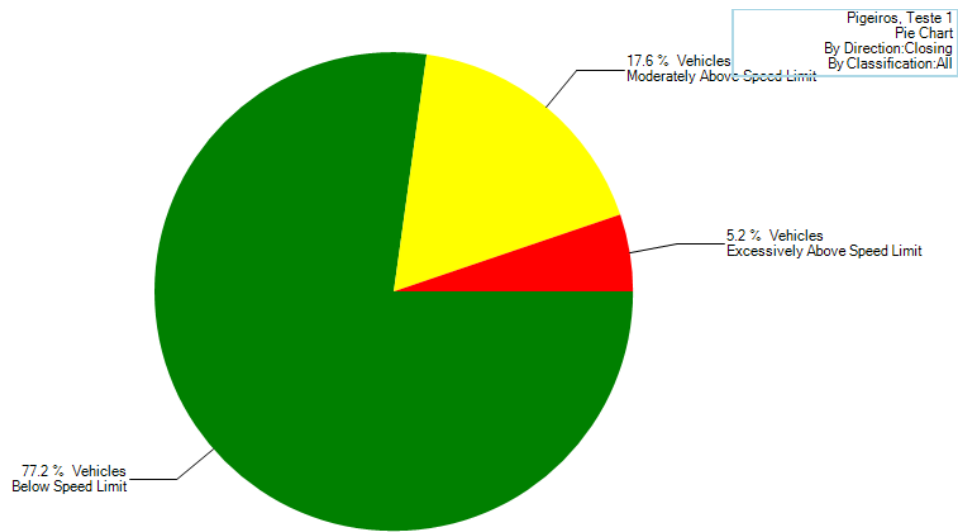


Figura 6.14: Teste ao radar - Cumprimento dos limites de velocidade

O gráfico circular reflete a percentagem de veículos que cumpriram e que excederam os limites. De entre um universo de 1285 amostras, 992 (77,2%) passaram abaixo dos 50 km/h, 226 (17,6%) aproximaram-se da passadeira entre os 50 e os 55 km/h, e os restantes 67 (5,2%) excederam os 55 km/h.

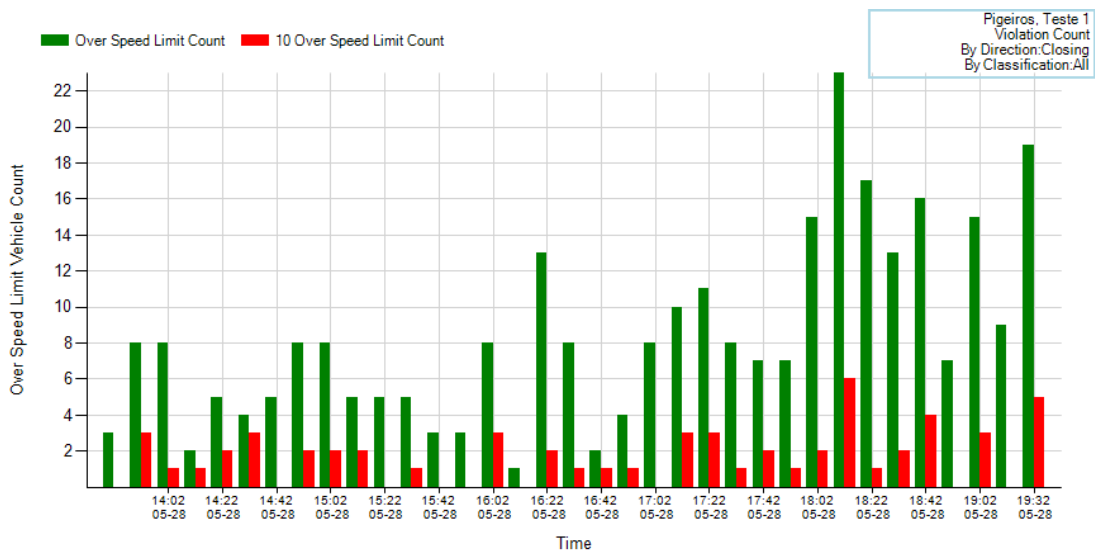


Figura 6.15: Teste ao radar - Excessos de velocidade detetados

O histograma por sua vez, retrata apenas os excessos de velocidade e o momento em que estes ocorreram. A verde temos os 293 veículos acima dos 50 km/h e a vermelho aqueles que, dentro deste universo, excederam os 60 km/h (19,8%).

Os últimos dois gráficos que se seguem, surgem como formas alternativas de representar os dados.

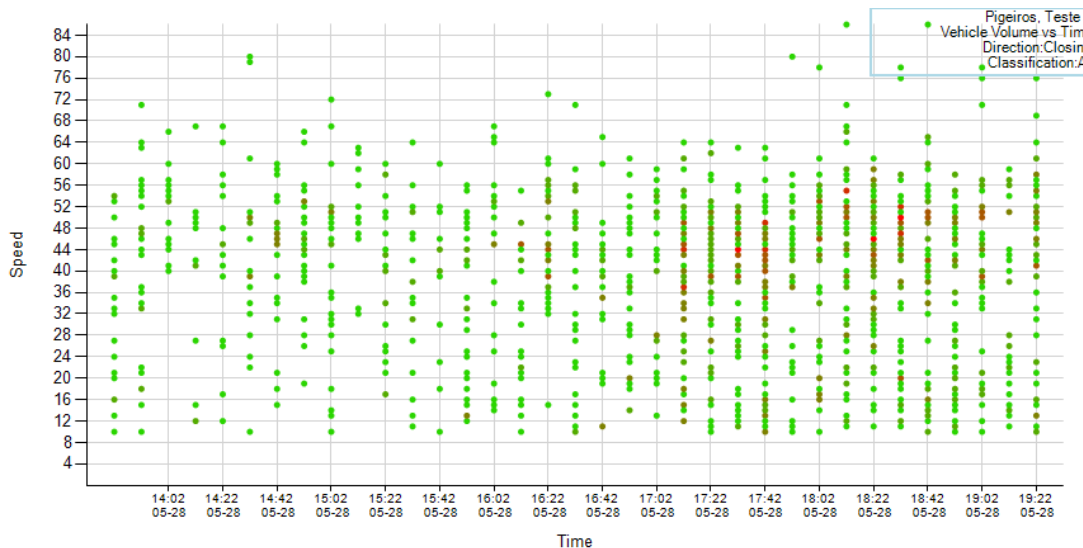


Figura 6.16: Teste ao radar - Marcação no tempo de todas as velocidades

A figura 6.16 espelha todas as velocidades registadas pelo sensor a cada 10 minutos.

Já a figura 6.17 divide as velocidades em intervalos, para que se perceba qual a gama mais frequente a cada meia hora.

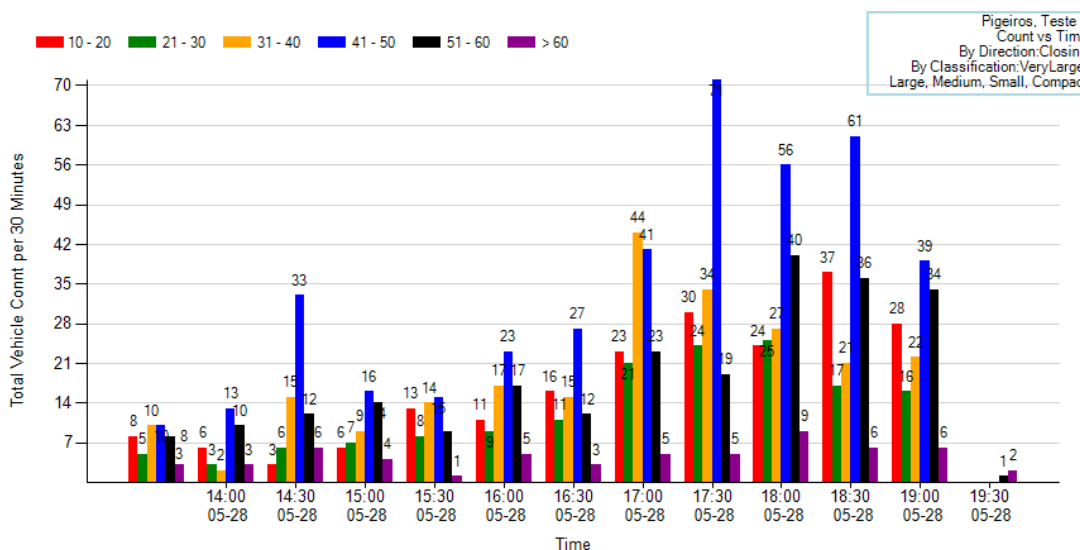


Figura 6.17: Teste ao radar - Número de deteções a cada meia hora

Concluindo o teste, resumiam-se na tabela da figura 6.18 os resultados em termos de deteções e velocidades.

Seguem-se agora testes mais específicos de contagem e classificação, efetuados com recurso à ferramenta “Real Time Recording” para confirmação visual da eficácia do radar.

Survey Result	
Total Vehicle Count	1285
Traffic Direction	Closing
Vehicle Count	1285
Posted Speed Limit	
<i>Vehicles Under the Speed Limit Count</i>	992
<i>Vehicles Under the Speed Limit Percentage</i>	77,2%
<i>Vehicles Over the Speed Limit Count</i>	293
<i>Vehicles Over the Speed Limit Percentage</i>	22,8%
Excessive Speed Threshold	
<i>Vehicles Over the Excessive Speed Count</i>	162
<i>Vehicles Over the Excessive Speed Percentage</i>	12,61%
<i>Average Speed Over Limit</i>	56,92 (km/h)
Speed Profile	
Average Speed	39,24 (km/h)
Minimum Speed	10 (km/h)
Maximum Speed	86 (km/h)
85th Percentile Speed	53 (km/h)
Standard Deviation	33 (km/h)
10 km/h Pace	35-44 (km/h)
In Pace Count	296
Data Recording Limits	
Highest Speed Allowed	120 (km/h)
Lowest Speed Allowed	10 (km/h)
Minimum Following Time	2 Second(s)

Figura 6.18: Resumo dos resultados do teste ao radar

Contagem

Foram realizados dois testes de contagem de veículos, no mesmo local e sentido que os testes anteriores. A diferença é que desta vez se confirmaram visualmente os alvos, tirando notas das ocorrências. Segue-se um breve resumo de ambos.

O primeiro teste foi realizado a 05/08/2019, tendo como hora de início as 16:20h e de fim as 17:00h. As condições atmosféricas eram de céu pouco nublado a limpo e vento moderado. O sensor foi instalado numa estrutura temporária com 2,35m de altura, a 1m da faixa de rodagem.

Mais importante do que obter resultados muito precisos, é perceber como é que as alterações nas configurações podem modificar o comportamento do radar. Este processo de análise é fundamental para compreender melhor o sistema, tendo em conta a sua complexidade. Definiram-se então os parâmetros que se seguem.

Configurações básicas:

- Unidades: km/h
- Direção: ambas (aproximação e afastamento)
- Precisão: 1 km/h
- Controlo de transmissão: automático

Configurações avançadas:

- Sensibilidade: 1 (mínimo detetável)

- Distância mínima detecção: 50 ft (aproximadamente 15 metros)
- Ângulo vertical: 30° para baixo
- Ângulo horizontal: 50° para a direita
- COM: RS-232 full-duplex, COM B
- LOG messages: enable
- Protocolo *streaming*: D0 (mas apenas o LOG é relevante na marcação dos pontos)
- Limite velocidade mínimo: 10 km/h
- Limite velocidade máximo: 120 km/h

Foram observados 124 veículos e a tabela 6.2 sumaria os resultados:

	Veículos no sentido de aproximação	Veículos no sentido de afastamento
Detecções corretas	58	50
Não detetados	10	6
Detecções múltiplas	0	0
Falsos positivos	0	1
Total real	68	56

Tabela 6.2: Resultados do 1° teste de contagem de veículos

Podemos assim garantir que 108 veículos foram corretamente detetados, perfazendo uma percentagem de detecção de 87,1%.

O segundo teste foi realizado a 06/08/2019, tendo como hora de início as 18:00h e de fim as 18:40h, verificando-se um maior fluxo de tráfego e entropia, por se tratar de hora de ponta. As condições atmosféricas eram de céu nublado e vento fraco.

Fizeram-se alterações nas configurações avançadas, com o objetivo de tentar melhorar a capacidade de detecção. As configurações básicas e a instalação foram mantidas, assim como a metodologia de análise.

Configurações avançadas:

- Sensibilidade: 6
- Distância mínima detecção: 50 ft (aproximadamente 15 metros)
- Ângulo vertical: 0° (apesar de fisicamente estar cerca de 30° para baixo)
- Ângulo horizontal: 0° (apesar de fisicamente estar cerca de 40° para a direita)
- COM: RS-232 full-duplex, COM B
- LOG messages: enable
- Protocolo *streaming*: B (permite observar a evolução da velocidade)
- Limite velocidade mínimo: 10 km/h
- Limite velocidade máximo: 200 km/h

	Veículos no sentido de aproximação	Veículos no sentido de afastamento
Deteções corretas	131	89
Não detetados	7	8
Deteções múltiplas	2	4
Falsos positivos	0	0
Total real	138	97

Tabela 6.3: Resultados do 2º teste de contagem de veículos

Registou-se a passagem de 235 viaturas, cujos resultados se apresentam na tabela 6.3.

De entre os veículos observados, 220 foram detetados corretamente pelo radar, correspondendo a uma percentagem de deteção de 93,6%.

Classificação

Para conseguir expor o procedimento e os resultados, é necessário compreender primeiro como é que funciona a classificação. Embora este radar não permita a medição direta do tamanho dos veículos, é capaz de fornecer indicações aproximadas consoante a intensidade dos sinais de retorno. Isto porque, na maioria dos casos, estes valores são proporcionais às dimensões reais dos objetos.

De acordo com o manual de utilizador [82], a montagem ideal para operações estatísticas é a da figura 6.19, porque o ângulo extremo faz com que apenas os veículos mais próximos sejam analisados, reduzindo os efeitos da entrada e saída dos alvos do campo de visão do radar. Este desvio deve ser compensado, inserindo o cosseno do ângulo nas configurações avançadas, mantendo assim uma medição precisa das velocidades.

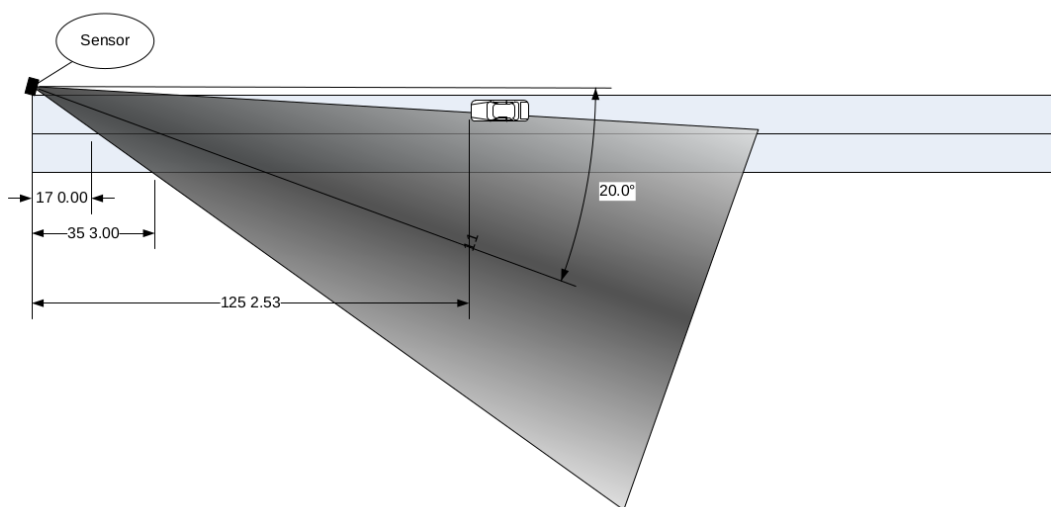


Figura 6.19: Montagem em ângulo para tarefas de classificação [82]

Em teoria, as 5 classes a que um veículo pode ser atribuído, definem-se da seguinte forma:

- **Classe 1 - Muito maiores que a média:** Autocarros e camiões de grande dimensão;
- **Classe 2 - Maiores que a média:** Camiões, carrinhas e Sport Utility Vehicles (SUV);
- **Classe 3 - Média:** Pequenas carrinhas, *crossovers* e carros;
- **Classe 4 - Menores que a média:** Carros pequenos como por exemplo os *smarts*;
- **Classe 5 - Muito menores que a média:** Motociclos, bicicletas e pedestres;

Existem 5 *thresholds* que permitem fazer a distinção entre as classes, sendo que, para cada sentido do fluxo de tráfego, podem ser estabelecidos *thresholds* distintos, definidos pelo programador ou obtidos automaticamente através de um processo de treino.

Um treino de classificação consiste em monitorizar 100 veículos por cada sentido analisado, calculando as intensidades observadas e ajustando posteriormente os parâmetros em conformidade. Observou-se que de fábrica os *thresholds* eram os seguintes:

Nomenclatura	Threshold
Classe 1	84
Classe 2	68
Classe 3	53
Classe 4	37
Classe 5	0

Tabela 6.4: Valores mínimos da intensidade de reflexão do sinal antes do treino

Foram realizados dois testes e por conseguinte dois treinos, em instalações e dias diferentes.

O primeiro treino foi efetuado a 27/09/2019, entre as 17:00h e as 18:00h. A instalação foi a mesma que se utilizou no sistema STAP, ou seja, o poste descrito no capítulo 4 e o tráfego foi analisado no sentido sul-norte (em aproximação à passadeira). É importante referir que sendo uma região fortemente industrializada e tendo o treino decorrido a uma sexta-feira ao final do dia, foram observados diversos veículos de grande porte, como camiões e autocarros.

O segundo treino decorreu a 23/11/2019 entre as 16:30h e as 17:15h, também focado no trânsito em aproximação à passadeira, mas desta vez no sentido norte-sul. O radar foi colocado a uma altura de 3,40m e a 5,50m da estrada. Este treino foi efetuado a um sábado à tarde e portanto, a maioria do tráfego era composto por veículos ligeiros.

Nomenclatura	Thresholds - Treino 1	Thresholds - Treino 2
Classe 1	89	78
Classe 2	79	71
Classe 3	68	63
Classe 4	58	56
Classe 5	0	0

Tabela 6.5: Valores mínimos da intensidade de reflexão do sinal após os treinos

Os *thresholds* obtidos são apresentados na tabela 6.5.

Após o primeiro treino foram observados 76 veículos, verificando a classe que lhes era atribuída. O processo foi repetido no final do segundo treino para 57 veículos. A tabela 6.6 exibe os resultados dessa análise.

Classificação	Atribuições - Teste 1	Atribuições - Teste 2
Classe 1	3	0
Classe 2	36	10
Classe 3	34	38
Classe 4	3	7
Classe 5	0	2

Tabela 6.6: Número de veículos observados e respetiva classificação

6.1.3 Sistemas de iluminação

Foram avaliadas duas funcionalidades distintas no que diz respeito às tecnologias LED: a sinalização e a iluminação das passadeiras.

Em relação à sinalização, considerou-se importante ter um interface gráfico que permitisse verificar o funcionamento das tiras RGB a qualquer altura, alterando a cor e intensidade dos LEDs através de qualquer dispositivo conectado pela rede à *Raspberry Pi*. Isto atribui uma maior flexibilidade à solução, caso haja necessidade de testar novos perfis luminosos.

Relativamente às luminárias, pretende-se perceber se os níveis de iluminação são adequados para uma passadeira de peões. Com esse objetivo em mente, compararam-se medições de luminância com as informações presentes no relatório da Federal Highway Administration (FHWA), no âmbito de um programa de Research and Development (R&D) sobre visibilidade [21]. Este relatório foi previamente analisado na sequência do estado da arte desta dissertação.

Começando pelo interface gráfico, recorreu-se a um *driver* da Neopixel Node-RED⁴ para *python* [81]. Não vou entrar nos detalhes da configuração do Node-RED, mas depois da sua instalação na *Raspberry Pi* é possível configurar um *dashboard* para acionar as tiras de LED através de um *browser*. O resultado pode ser visto na figura 6.20.

Além do controlo de todos os LEDs em simultâneo, através da opção “select strip color”, é possível atribuir uma cor diferente a cada um deles, bastando para isso selecionar o LED que se pretende alterar e escolher a cor pretendida no “color picker”. Também é possível correr durante alguns segundos um efeito de “arco-íris” e apagar todos os LEDs. Todas as configurações são facilmente alteráveis, seja gravar novas cores, definir novos efeitos, modificar os tempos de atuação, ou alterar o número de LEDs.

Em relação aos níveis de intensidade luminosa nas passadeiras, o teste consistiu em colocar um dos postes em funcionamento normal durante a noite, e portanto, com a intensidade ajustável através do sensor de luminosidade da figura 4.7. Em seguida, utilizou-se um segundo sensor de luminosidade conectado a uma *Raspberry Pi* para efetuar leituras de luminância vertical.

⁴ferramenta de programação para conectar dispositivos de hardware, Application Programming Interfaces (API) e serviços online

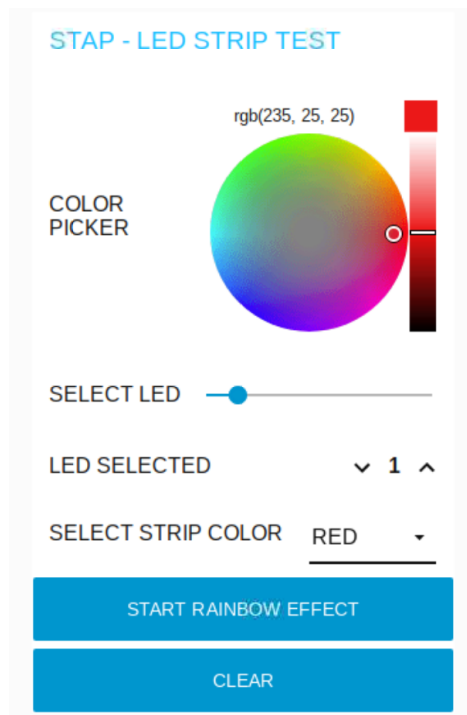


Figura 6.20: Interface para controlo das tiras de LED RGB

A medição foi efetuada a 1,50m de altura, no centro da zona de passagem da passadeira. Sabendo que a instalação completa é composta por duas luminárias e que cada luminária é responsável por iluminar metade da passadeira, estaremos a obter valores um pouco inferiores aos reais e que representam o pior cenário possível.

Foram realizadas duas medições, uma primeira sem a presença de peão (modo de poupança de energia) e uma segunda com presença de peão (intensidade máxima). Os resultados podem ser consultados na tabela 6.7.

	Intensidade mínima (50%) Poupança de energia	Intensidade máxima (100%) Peão detetado
Valor mínimo	11 lx	22 lx
Valor máximo	13 lx	25 lx

Tabela 6.7: Valores de luminância vertical no centro da passadeira

6.2 Análise de resultados

Agora que se descreveram os procedimentos de teste e os resultados obtidos, é chegada a altura de avaliar a solução. Esta secção divide-se em duas partes, uma inicial em que comento individualmente o que se observou nos diferentes testes e uma segunda em que apresento alguns exemplos de funcionamento do sistema, integrando todas as tecnologias.

Rede neuronal

Através do treino da rede neuronal, foi possível observar melhorias muito significativas e promissoras, quando comparado com a versão *tiny* do YOLO. A rede não só se tornou mais rápida, mas também mais precisa, como comprovam os testes de mAP.

Relativamente às deteções no *dataset* de validação, verificou-se um aumento no número de deteções acertadas e uma diminuição dos falsos positivos. Ainda assim, é possível observar um elevado número de falsos positivos quando a confiança se diminui para 10%, cuja principal causa aparenta ser a ocorrência de casos de deteções múltiplas. Como sabemos, apenas uma previsão pode ser atribuída por validação, o que faz com que essas deteções de menor confiança sejam declaradas como falsos positivos.

Em termos práticos, as deteções múltiplas não são críticas, porque não se tratam de falsas deteções propriamente ditas. O único inconveniente é que o número de pessoas detetadas pelo sistema pode, por vezes, ser superior ao número real de pedestres. Para evitar essa situação e obter o número exato de peões, aumentou-se a confiança necessária para um objeto ser considerado correto para 30%. Isto permite maiores garantias de que de facto aquela deteção corresponde a uma pessoa e funciona também como um filtro do *background*, para que não sejam detetados pedestres a grandes distâncias e que não pretendam atravessar a passadeira.

Depois da instalação das câmaras nos postes, comprovou-se que a deteção ocorria sem problemas a curtas distâncias (até 6m, como especificado no capítulo 3), sendo normal a obtenção de taxas de confiança elevadas, tanto de dia como de noite, sempre que não existiam obstáculos à deteção.

Mas como nem tudo são aspetos positivos, constatou-se que as sombras e iluminação direta da câmara podem afetar a capacidade de deteção da rede. A medida mais eficaz que se tomou para tentar minimizar esse problema, foi a instalação da câmara por baixo da caixa de derivação onde se encontra a eletrónica. Isto permitiu reduzir a luz solar que atinge diretamente a câmara e garantir que a imagem não é afetada pela luminária durante a noite.

Uma última nota para o pequeno *delay* observado nas tarefas de deteção, devido ao tempo de processamento de imagem da rede neuronal. Quando as zonas de deteção forem muito estreitas isto pode vir a tornar-se problemático. Para resolver este problema, acredito que seja possível otimizar ainda mais o código, mas considerando todas as melhorias já implementadas neste sentido, penso que a hipótese mais viável é mesmo um *upgrade* da placa de controlo, como se referiu em capítulos anteriores.

Radar

O teste às capacidades do radar demonstrou que o este é capaz de contar, classificar, medir velocidades, detetar excessos e anotar os tempos das ocorrências.

Como seria de esperar, observa-se no gráfico 6.12 um aumento no fluxo de tráfego na hora de ponta, comprovando que as medições estão em concordância com a realidade.

No gráfico seguinte (6.13), não foram detetados veículos de classe 5, apesar de terem sido analisadas mais de mil e duzentas viaturas. Isto demonstra uma falha de classificação, que ocorreu porque ainda não se tinha procedido ao treino do radar, de modo a ajustar os *thresholds* à instalação.

Seguindo agora para os testes de contagem, estes permitiram perceber muito melhor o funcionamento do radar. No entanto, como na altura essa compreensão era limitada, certas configurações levaram a um decaimento da precisão.

No primeiro teste ocorreram dois erros cruciais que justificam a maioria das falhas.

O primeiro erro está associado ao ajuste dos ângulos vertical e horizontal. O objetivo era compensar um possível decaimento da velocidade devido ao ângulo de montagem, mas como os valores inseridos estavam desajustados, as velocidades medidas foram muito superiores às reais. Como o limite máximo era de 120 km/h, os veículos que passavam um pouco mais depressa foram então ignorados pelo sensor. Situação que foi corrigida no segundo teste.

O segundo erro foi o ajuste da sensibilidade para o mínimo. Apesar desta alteração filtrar os movimentos de fundo, elimina também o reconhecimento de objetos mais pequenos, fazendo com que as motos e bicicletas fossem ignoradas pelo radar.

No segundo teste já se obtiveram melhores resultados, mas a sensibilidade um pouco baixa, ainda foi a causa de um terço das não deteções, mais uma vez por conta dos motociclos. Além disso, surgiram deteções múltiplas que podiam ter sido reduzidas/eliminadas com um reajuste da posição e do ângulo do radar.

Um problema transversal aos dois testes e que ainda estou à procura de corrigir, são as falhas nas deteções causadas por sequências de veículos muito próximos. Coloca-se a hipótese de que o problema esteja relacionado com os parâmetros “target acquisition density” e “target acquisition span”, mas só através da realização de novos testes é que será possível confirmar esta teoria.

Dá-se por terminada esta análise com algumas notas sobre a classificação de veículos. Aqui o mais importante é não tanto o número de veículos observados durante os testes, mas sim os valores dos *thresholds* obtidos nos treinos.

Observou-se que a média das dimensões dos 100 veículos analisados no primeiro treino era muito superior à do segundo, o que obviamente levou a *thresholds* mais elevados. No entanto, as duas situações correspondem a casos um pouco extremos: um em que quase não existiam veículos de pequena dimensão, e outro em que não se observaram veículos de grande porte.

Para um ajuste mais adequado ao trânsito que normalmente flui naquela estrada, podemos optar por uma de duas opções. Ou estabelecemos os limites como uma média ponderada das medições anteriores, ou realiza-se um novo treino durante aquilo que em tráfego se chama um dia típico (entre terça e quinta-feira).

Ainda assim foi possível comprovar em ambos os testes, que de facto o radar é capaz de distinguir perfeitamente entre os veículos maiores e menores. É, no entanto, um pouco mais difícil distinguir entre as classes intermédias, até porque nesses casos não é fácil definir claramente que veículo deve corresponder a que classe.

Iluminação LED

Sobre a aplicação que serve de interface às tiras de LED, basta realçar que funciona sem problemas, permitido testar o funcionamento de todos os LEDs a qualquer momento, controlando-os em simultâneo ou individualmente.

Uma análise mais interessante diz respeito à iluminação de presença.

Sabemos à partida que são recomendados no mínimo 20 *lx* verticais, medidos a 1,5m de altura da estrada, de modo a fornecer iluminação suficiente à passadeira. Assim garantem-se níveis de luminosidade suficientes para uma boa deteção dos pedestres, por parte dos condutores, a uma distância que lhes permita um tempo de resposta adequado [21].

A tabela 6.7 comprova que, no momento em que um peão é detetado, o nível mínimo de luminosidade para as condições referidas é de 22 *lx*. Esta medição, ao ser realizada na zona central da passadeira, dá-nos boas garantias, porque movimentos para qualquer um dos lados

levam sempre a uma aproximação de uma das luminárias e portanto, a uma maior intensidade luminosa.

Se por um lado conseguimos garantir luminosidade suficiente para a detecção por parte dos condutores, também foi possível comprovar visualmente que o modo de poupança de energia fornece iluminação suficiente, garantindo boa visibilidade nas proximidades da passadeira e um funcionamento correto da detecção de peões por parte da rede neuronal.



Figura 6.21: Vista de topo durante o teste de luminosidade - modo de poupança de energia

6.2.1 Integração dos subsistemas

Ao longo dos testes realizados neste capítulo foram registados alguns momentos, após a instalação do poste principal no local, que ajudam a visualizar o funcionamento do projeto como um todo.

Ao executar o programa principal na *Raspberry Pi*, a rede neuronal é inicializada e são lidos os valores de luminosidade atuais.

Como estes testes em particular foram realizados ao final da tarde, o sensor de luminosidade informa a luminária de que deve ajustar gradualmente a sua intensidade, até um máximo de 50%, caso a luminosidade ambiente seja em torno de zero.

Suponhamos agora que um veículo se aproxima da passadeira a uma velocidade superior a 50 km/h. Como o radar está continuamente a analisar o tráfego, essa informação vai ser transmitida de imediato para a placa controladora. É então acionado, por precaução, um sinal vermelho para os peões, informando-os que não devem atravessar neste momento. Simultaneamente, é emitido um sinal laranja intermitente para os veículos, alertando-os que devem reduzir a velocidade.

A situação agora descrita pode ser observada na figura 6.22, onde um veículo que seguia no sentido sul-norte foi detetado em excesso de velocidade, despoletando o alerta. O sistema manteve-se a funcionar com baixa intensidade, tal como seria de esperar.



Figura 6.22: Detecção de um excesso de velocidade

Por outro lado, se um peão passar na frente da câmara, como acontece na figura 6.23, será registado de imediato. São tiradas fotografias constantemente, a uma frequência superior a 1 vez por segundo, fazendo com que qualquer movimento seja processado e detetado pela rede neuronal. Nenhuma fotografia é armazenada após processamento.

Detetado o peão, a intensidade luminosa aumenta para os 100%, captando a atenção dos intervenientes e melhorando a visibilidade. É também emitido um sinal de paragem de cor vermelha aos condutores, permitindo ao peão atravessar em segurança - figura 6.24.

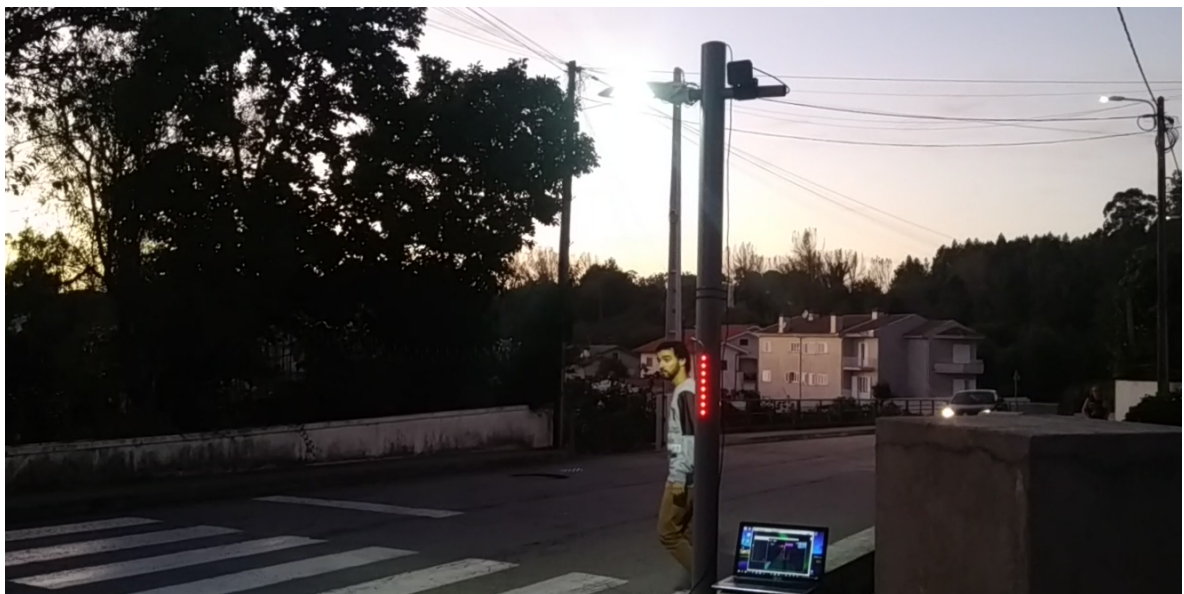


Figura 6.23: Detecção de um peão que pretende atravessar a via

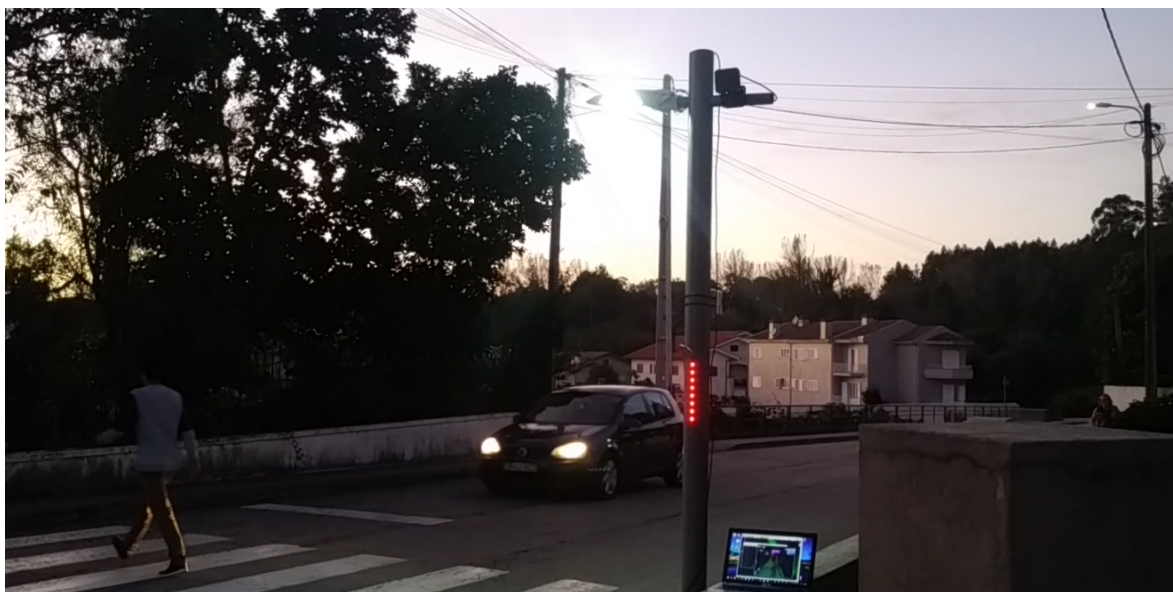


Figura 6.24: Travessia em segurança com o sistema STAP

Terminada a travessia, a sinalização é desligada e a luminária regressa ao modo de poupança de energia. Todas as transições de sinais luminosos são efetuadas de modo suave, privilegiando a segurança e considerando os tempos de reação necessários ao ser humano.

Conclui-se assim a validação do projeto e a análise dos resultados, sendo que no próximo capítulo serão tiradas conclusões sobre o trabalho realizado e enumeradas as melhorias que se pretendem implementar num futuro próximo.

Capítulo 7

Considerações finais

Chegados ao fim do projeto é necessário revisitar um pouco todo o trabalho desenvolvido, desde a especificação de requisitos até à validação dos resultados, passando pela implementação da solução. Só assim é possível compreender se o trabalho foi ou não bem feito e se os objetivos foram cumpridos.

Este capítulo divide-se em duas partes, uma primeira onde se fala sobre os resultados alcançados e uma segunda mais focada no futuro do projeto STAP. Sendo a última secção da dissertação, não deixa de ser uma das mais importantes, porque é daqui que se pode partir no sentido de melhorar a solução e transformar um protótipo funcional num produto comercializável.

7.1 Conclusões

Podemos afirmar que este era um projeto ambicioso, já que foi desenvolvido do zero no âmbito desta dissertação. Envolve várias tecnologias que por si só dariam um projeto, como os radares de trânsito ou as redes neuronais, mas ainda assim conseguiram-se alcançar os objetivos propostos. Relembrando que os requisitos do sistema definidos no ponto 3.2 do documento, eram os seguintes:

- Detetar peões e veículos;
- Medir velocidade instantânea dos veículos;
- Informar os condutores em excesso de velocidade;
- Sinalizar situações de perigo nas proximidades;
- Iluminar a passadeira durante a noite;
- Manter uma instalação fácil e rápida.

Estes requisitos foram cumpridos e foi desenvolvido com sucesso um protótipo em tamanho real que permite combater problemas de visibilidade, excessos de velocidade e distrações por parte dos pedestres.

Todas as decisões tomadas foram devidamente fundamentadas, procurando sempre recorrer a estudos e trabalhos desenvolvidos pela comunidade científica na área em questão.

É importante realçar novamente que este era um projeto para ser desenvolvido em parceria com o DFIS, responsável pela projeção da sinalização na estrada no âmbito de outra

dissertação. A substituição da projeção por tiras de LED nos postes, foi uma decisão estratégica no sentido de conseguir concluir um protótipo funcional.

Uma funcionalidade adicional que gostaria de ter implementado ainda no contexto desta dissertação era a comunicação entre os postes. Tal ainda não foi possível, no entanto, considerando que apenas era proposta a prototipagem de um dos postes e tendo sido desenvolvidos os dois que compõem a solução completa, estamos mais próximos de atingir esse novo objetivo.

É esta busca por melhorar e complementar a solução atual que nos leva ao próximo e último tópico desta dissertação.

7.2 Trabalho futuro

Enumeram-se agora um conjunto de alterações que se podem aplicar à solução, com o objetivo de a otimizar:

- Como já se referiu, o objetivo inicial era projetar a sinalização na estrada, delimitando as passeadeiras. Esta opção ainda pode vir a ser considerada, aumentando a probabilidade de pessoas distraídas ao telemóvel conseguirem ver a sinalização.
- A rede neuronal, apesar dos resultados promissores, ainda tem uma grande margem de manobra para melhorias. Para isso é necessário um novo treino da rede.

Tal como aconteceu com o *dataset* de validação, é necessária uma correção das anotações do *dataset* de treino, sendo que neste momento já foram verificadas 17.000 das 80.000 imagens presentes no COCO 2017. Devemos também manter no *dataset* as imagens que não possuam objetos a detectar, bem como os respetivos ficheiros “.txt” vazios [78].

Finalmente, também seria interessante experimentar alterar a estrutura da rede, conforme descrito no artigo “An improved tiny-yolov3 pedestrian detection algorithm” [83].

- Melhorar a rede neuronal, por si só, servirá de pouco se continuarmos a utilizar um controlador limitado. Deve-se portanto substituir a *Raspberry Pi 3B+* pela versão 4 (4GB de Random Access Memory (RAM)), ou por outra placa similar com capacidades superiores à atual.
- Como complemento ou em substituição da sinalização laranja intermitente, podemos aplicar no poste um painel informativo com a velocidade instantânea.
- Por fim, para tornar o protótipo completamente à prova dos elementos, devemos proteger o módulo de câmara com um material IP65 transparente.

Além das alterações já descritas, podemos ainda desenvolver as novas funcionalidades que se seguem:

- Integrar tecnologia LoRa para comunicação sem fios entre os postes.
- Adicionar um painel solar e uma bateria para tornar a solução ainda mais prática, descartando a necessidade de alimentação externa.
- Criar uma plataforma *web* com recolha de informações para uma base de dados (dados de tráfego, fluxo de pedestre, situações de perigo detetadas, entre outras).

- Integrar sensores de pluviosidade e inundação que ajudem a adequar os níveis de iluminação, não só consoante a luminosidade, mas também em concordância com as condições climatéricas. Pode ainda ser gerado um alerta de cor diferente ou transmitida a informação através do painel informativo, sempre que existir risco de *aquaplaning*.
- Utilizar as capacidades da rede neuronal para detetar pessoas com mobilidade reduzida, adequando assim o tempo de passagem de acordo com o tipo de peão.

Bibliografia

- [1] ANSR, 2018. Sinistralidade Rodoviária. *Vítimas no local* [Em linha]. Disponível em: <http://www.ansr.pt/Estatisticas/RelatoriosDeSinistralidade/Pages/default.aspx> [Consult. 07 de agosto de 2019].
- [2] PRP, 2017. Estudo da PRP revela que mais de 15% dos peões distraem-se com o telemóvel enquanto atravessam a estrada. *Comunicado de Imprensa* [Em linha]. Disponível em: <http://prp.pt/wp-content/uploads/2017/06/2-Distracao-Telemovel-e-Peoes.pdf> [Consult. 07 de agosto de 2019].
- [3] International Symposium on Ambient Intelligence and Embedded Systems [Em linha]. Disponível em: <http://amies.international-symposium.org/index.html> [Consult. 14 de setembro de 2019].
- [4] World Health Organization, 2018. The top 10 causes of death. [Em linha]. Disponível em: <https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death> [Consult. 18 de setembro de 2019].
- [5] Ana Tiemi, 2019. Você conhece o verdadeiro impacto dos custos dos acidentes de trânsito? *Prevenção de acidentes* [Em linha]. Disponível em: <https://veltec.com.br/custos-dos-acidentes-de-transito/> [Consult. 18 de setembro de 2019].
- [6] *Lighted Zebra Crossing* [Em linha]. Apeldoorn (Holanda): Lighted Zebra Crossing B.V.. Disponível em: <http://www.lightedzebracrossing.com/product/?lang=en> [Consult. 5 de setembro de 2019].
- [7] Pedro Venâncio, 2018. Empresa de Bodegraven testa semáforo de pavimento para peões. *Transportes em revista* [Em linha]. Disponível em: <http://www.transportesemrevista.com/Default.aspx?tabid=210&language=pt-PT&id=57770> [Consult. 31 de julho de 2019].
- [8] Jeffrey Heller, 2019. Tel Aviv tries out new crosswalk lights for cellular addicts. *Technology news* [Em linha]. Disponível em: <https://www.reuters.com/article/us-israel-smartphone-zombies/tel-aviv-tries-out-new-crosswalk-lights-for-cellular-addicts-idUSKCN1QV2QP?il=0> [Consult. 31 de julho de 2019].
- [9] *Smart Crosswalk IRWL System* [Em linha]. Santa Rosa (Califórnia - EUA): LightGuard Systems, Inc..

- Disponível em: <https://www.lightguardsystems.com/smart-crosswalk-in-roadway-warning-light-irwl-system/> [Consult. 21 de setembro de 2018].
- [10] Aterriza en España el paso de peatones «inteligente», 2016. *La Voz de Cádiz* [Em linha]. Disponível em: https://www.lavozdigital.es/motor/reportajes/abci-ateriza-a-espana-paso-peatones-inteligente-201604200004_noticia.html [Consult. 20 de setembro de 2018].
- [11] *Proyecto “Urban Lab” de Pasos de Peatones Inteligentes en Cambrils* [Em linha]. Tarragona (Espanha): Llumtraffic. Disponível em: <http://www.llumtraffic.com/cliente?id=12> [Consult. 20 de setembro de 2018].
- [12] *Passadeiras seguras e inteligentes com novo sistema instalado em Guimarães*, 2018 [Em linha]. Guimarães: Notícias DDE. Disponível em: https://www.cm-guimaraes.pt/pages/1418?news_id=3738 [Consult. 19 de setembro de 2018].
- [13] *Passadeira Inteligente* [Em linha]. Matosinhos: Habidom - Sinalização Rodoviária, Lda. Disponível em: <https://www.habidom.com/pt-PT/produtos/7/sistemas-integrados/8/passadeira-inteligente> [Consult. 19 de setembro de 2018].
- [14] *Paso de peatones Inteligente*, 2014 [Em linha]. Badajoz (Espanha): SICE. Disponível em: <https://www.sice.com/actualidad/paso-de-peatones-inteligente> [Consult. 20 de setembro de 2018].
- [15] Martes, 2014. El Ayuntamiento retirará plazas de aparcamiento para eliminar obstáculos en los pasos de peatones. *Hoy* [Em linha]. Disponível em: <https://www.hoy.es/badajoz/201412/16/ayuntamiento-retirara-plazas-aparcamiento-20141216153655.html> [Consult. 1 de outubro de 2019].
- [16] Minwoo Park, 2019. South Korea radar and thermal camera system warns ‘smartphone zombies’ of traffic. *Technology news* [Em linha]. Disponível em: <https://www.reuters.com/article/us-southkorea-smartphones-crossing-idUSKCN1R0029> [Consult. 27 de agosto de 2019].
- [17] *Starling CV*, 2017 [Em linha]. Londres (Inglaterra): Umbrellium Ltd. Disponível em: <https://umbrellium.co.uk/products/starling-cv/> [Consult. 20 de setembro de 2018].
- [18] Mako, E. Szakonyi, P. 2016, *Evaluation of Human Behaviour at Pedestrian Crossings*. [livro eletrônico] Transportation Research Procedia (Vol. 14, 2121-2128). Elsevier B.V. Disponível em: <https://doi.org/10.1016/j.trpro.2016.05.227> [Consult. 19 de setembro de 2018].
- [19] Beckwith, D. Hunter-Zaworski K. 1997, *Passive Pedestrian Detection at Unsignalized Crossings*. [livro eletrônico] Transportation Research Record (No. 1636). Disponível em: <https://pdfs.semanticscholar.org/0b8c/7f892dbcf4b3ec9ead09bdef4d99b5f57372.pdf> [Consult. 27 de setembro de 2018].

- [20] Czajewski, W. Dabkowski, P. Olszewski, P. 2013, *Innovative solutions for improving safety at pedestrian crossings*. [livro eletrônico] Archives of Transport System Telematics (Vol. 6, 16-22).
Disponível em: <http://yadda.icm.edu.pl/yadda/element/bwmeta1.element.baztech-d8e6be07-89d4-43dd-85d4-bbdaea12999c> [Consult. 21 de setembro de 2018].
- [21] Gibbons, R. Edwards, C. Williams, B. Andersen C. 2008, *Informational Report on Lighting Design for Midblock Crosswalks*. [livro eletrônico] U.S. Department of Transportation Federal Highway Administration (No. FHWA-HRT-08-053).
Disponível em: <https://www.fhwa.dot.gov/publications/research/safety/08053/> [Consult. 14 de agosto de 2019].
- [22] Fotios, S. Gibbons, R. 2017, *Road lighting research for drivers and pedestrians: The basis of luminance and illuminance recommendations*. [livro eletrônico] Lighting Research & Technology (Vol. 50, 154-186).
Disponível em: https://www.researchgate.net/publication/322020461_Road_lighting_research_for_drivers_and_pedestrians_The_basis_of_luminance_and_illuminance_recommendations [Consult. 3 de agosto de 2019].
- [23] *Green Energy*, 2018 [Em linha].
Disponível em: <https://www.engineergirl.org/108043/Green-Energy> [Consult. 18 de outubro de 2018].
- [24] *How PIRs Work* [Em linha].
Disponível em: <https://learn.adafruit.com/pir-passive-infrared-proximity-motion-sensor/how-pirs-work> [Consult. 18 de outubro de 2018].
- [25] *Ultrasound Expert Witness: What Ultrasound is and How It Works* [Em linha].
Disponível em: <https://riversonicsolutions.com/ultrasound-expert-witness-what-ultrasound-is-and-how-it-works/> [Consult. 3 de outubro de 2019].
- [26] *HB100 Miniature Microwave Motion Sensor* [Em linha].
Disponível em: <https://www.itead.cc/hb100-miniature-microwave-motion-sensor.html> [Consult. 4 de outubro de 2019].
- [27] *Flexidome IP starlight 7000 VR* [Em linha].
Disponível em: https://resource.boschsecurity.com/documents/NIN_733_Data_sheet_enUS_10263325835.pdf [Consult. 18 de outubro de 2018].
- [28] *Conheça os conceitos luminotécnicos básicos*, 2018 [Em linha].
Disponível em: <https://www.vivadecora.com.br/pro/iluminacao/conceitos-luminotecnicos/> [Consult. 9 de outubro de 2019].
- [29] *Biologia e aplicações*, 2013 [Em linha]. São Paulo (Brasil): Pereira, W.
Disponível em: http://www.crbiodigital.com.br/01/williamroberto?c=.&ac=P_Teses&txt=3277333332 [Consult. 11 de julho de 2019].
- [30] Seco, A. Macedo, J. Costa, A. 2008, *Manual do Planejamento de Acessibilidade e Transportes*. [livro eletrônico] Comissão de Coordenação e Desenvolvimento Regional do Norte. Peões (Capítulo 8).

- Disponível em: http://www.estgv.ipv.pt/PaginasPessoais/vasconcelos/Documentos/ManualdeAcessibilidades/ManuaisCCDRNmiolo_AF/08Peoes_AF.pdf [Consult. 10 de outubro de 2019].
- [31] Rui Oliveira, 2015. Distância de paragem: distância de reacção. *ANSR* [Em linha]. Disponível em: http://www.ansr.pt/SegurancaRodoviaria/ArtigosTecnicos/Documents/Artigo_absolute%20motors.pdf [Consult. 10 de outubro de 2019].
- [32] Rui Oliveira, 2015. Distância de paragem: distância de travagem. *ANSR* [Em linha]. Disponível em: [http://www.ansr.pt/SegurancaRodoviaria/ArtigosTecnicos/Documents/Dist_travagem%20\(absolute%20Motors\)%20r.pdf](http://www.ansr.pt/SegurancaRodoviaria/ArtigosTecnicos/Documents/Dist_travagem%20(absolute%20Motors)%20r.pdf) [Consult. 10 de outubro de 2019].
- [33] Nicki Hines, 2017. Adventures of an O-ring - Part 1 In The Beginning [Em linha]. Disponível em: <https://www.oringslimited.co.uk/adventures-o-ring-part-1-beginning/> [Consult. 16 de outubro de 2019].
- [34] Adilson Thomsen, 2014. Funcionamento da Câmera Raspberry Pi [Em linha]. Disponível em: <https://www.filipeflop.com/blog/modulo-camera-raspberry-pi/> [Consult. 17 de outubro de 2019].
- [35] *AMG061-002 - SPEED SENSOR* [Em linha]. Disponível em: <https://www.amg-microwave.com/-Doppler-radar-sensors-> [Consult. 19 de outubro de 2018].
- [36] *Delta DRS1000 Non-contact Speed Sensor* [Em linha]. Disponível em: http://www.gmheng.com/speed_sensor.php [Consult. 19 de outubro de 2018].
- [37] *MLR – Multi-Lane Radar detector* [Em linha]. Disponível em: <https://assets.new.siemens.com/siemens/assets/api/uuid:a319f978966348ebff9efd67998014fa13cdef55/version:0/multi-lane-radar-detector-mlr-en.pdf> [Consult. 19 de outubro de 2018].
- [38] *Stalker Stationary Speed Sensor* [Em linha]. Disponível em: <https://www.stalkerradar.com/oem/Stationary.html> [Consult. 19 de outubro de 2018].
- [39] *Stalker Stationary Statistics Sensor* [Em linha]. Disponível em: <https://www.stalkerradar.com/oem/Stationary-Statistics-Sensor.html> [Consult. 19 de outubro de 2018].
- [40] *RXC Series Microwave Radar* [Em linha]. Disponível em: <https://www.lasemaforica.com/en/products/traffic-data-survey-systems/radar/rxc-series> [Consult. 2 de novembro de 2018].
- [41] *TSL2591 Datasheet* [Em linha]. Disponível em: https://cdn-shop.adafruit.com/datasheets/TSL25911_Datasheet_EN_v1.pdf [Consult. 30 de novembro de 2018].

- [42] *Luminária LED Brooklyn 40W 24V* [Em linha].
Disponível em: <https://www.efectoled.com/pt/comprar-luminarias-led-de-iluminacao-publica/5857-luminaria-led-brooklyn-40w-24v.html> [Consult. 19 de outubro de 2018].
- [43] *Solarcupid SC-L48W-B-24V* [Em linha].
Disponível em: <http://www.solarcupid.com/index.php?c=msg&id=327&> [Consult. 19 de outubro de 2018].
- [44] *Ledenvo LED Street Light* [Em linha].
Disponível em: https://www.osram.asia/ecat/Road%20luminaires-Outdoor%20luminaires-Luminaires/asia/en/GPS01_1029355/PP_APAC_Asia_eCat/ [Consult. 19 de outubro de 2018].
- [45] *STRATOS N - DLeds* [Em linha].
Disponível em: http://www.detas.hr/wp-content/uploads/2017/02/DLEDS_catalog_street_industrial_v7.pdf [Consult. 19 de outubro de 2018].
- [46] *Luminária LED Roma* [Em linha].
Disponível em: https://luxconcept.pt/Produtos/ProdID/517/CatID/2/Lumin%C3%A1ria_LED_Roma [Consult. 6 de maio de 2019].
- [47] *INDO Products - RDC4* [Em linha].
Disponível em: http://indolighting.com/wp-content/uploads/2015/11/RDC4-Road_Rev_1_01.pdf [Consult. 26 de março de 2019].
- [48] *VMA410* [Em linha].
Disponível em: https://www.velleman.eu/downloads/29/vma410_a4v01.pdf [Consult. 15 de julho de 2019].
- [49] *60W Single Output LED Driver* [Em linha].
Disponível em: <https://www.mouser.com/datasheet/2/260/NPF-60D-SPEC-806235.pdf> [Consult. 15 de julho de 2019].
- [50] *Pigpio library - Python Interface* [Em linha].
Disponível em: <http://abyz.me.uk/rpi/pigpio/python.html> [Consult. 15 de agosto de 2019].
- [51] *TEZ - Transformer for PCB* [Em linha].
Disponível em: <http://biakom.com/pdf/TEZ-Breve.pdf> [Consult. 15 de julho de 2019].
- [52] *L7800 Series - Positive Voltage Regulators* [Em linha].
Disponível em: <https://www.hobbytronics.co.uk/datasheets/L7812.pdf> [Consult. 15 de julho de 2019].
- [53] *Getting started with the Camera Module* [Em linha].
Disponível em: <https://projects.raspberrypi.org/en/projects/getting-started-with-picamera/3> [Consult. 16 de outubro de 2019].
- [54] *Stalker Stationary Speed Sensor II Technical Manual*. [Em linha] 011-0131-00 Rev. M
Disponível em: [https://www.stalkerradar.com/oem/downloads/011-0131-00%](https://www.stalkerradar.com/oem/downloads/011-0131-00%20Rev%20M.pdf)

- 20Stalker%20Stationary%20Speed%20Sensor%20II%20Technical%20Manual.pdf [Consult. 28 de maio de 2019].
- [55] Menegaz M., 2018. Understanding YOLO. *Hackernoon* [Em linha].
Disponível em: <https://hackernoon.com/understanding-yolo-f5a74bbc7967> [Consult. 9 de abril de 2019].
- [56] Cornelisse D., 2018. An intuitive guide to Convolutional Neural Networks. *Machine Learning - freeCodeCamp* [Em linha].
Disponível em: <https://www.freecodecamp.org/news/an-intuitive-guide-to-convolutional-neural-networks-260c2de0a050/> [Consult. 23 de abril de 2019].
- [57] *Student Notes: Convolutional Neural Networks (CNN) Introduction* [Em linha].
Disponível em: <https://indoml.com/2018/03/07/student-notes-convolutional-neural-networks-cnn-introduction/> [Consult. 30 de outubro de 2019].
- [58] Hui J., 2018. mAP (mean Average Precision) for Object Detection. *Medium* [Em linha].
Disponível em: https://medium.com/@jonathan_hui/map-mean-average-precision-for-object-detection-45c121a31173 [Consult. 24 de abril de 2019].
- [59] Tang B., 2018. Image and Video Understanding: A Roadmap For Implementation. *Stradigi* [Em linha].
Disponível em: <https://www.stradigi.ai/blog/computer-vision-implementation/> [Consult. 4 de novembro de 2019].
- [60] Rosebrock A., 2016. Intersection over Union (IoU) for object detection [Em linha].
Disponível em: <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/> [Consult. 8 de maio de 2019].
- [61] Redmon J., 2016. Darknet: Open Source Neural Networks in C [Em linha].
Disponível em: <http://pjreddie.com/darknet/> [Consult. 9 de abril de 2019].
- [62] Redmon J., Divvala S., Girshick R., Farhadi A., 2016. You Only Look Once: Unified, Real-Time Object Detection. *University of Washington, Allen Institute for AI, Facebook AI Research* [Em linha].
Disponível em: <https://arxiv.org/pdf/1506.02640.pdf> [Consult. 9 de abril de 2019].
- [63] Redmon J., Farhadi A., 2016. YOLO9000: Better, Faster, Stronger. *University of Washington, Allen Institute for AI* [Em linha].
Disponível em: <https://arxiv.org/pdf/1612.08242.pdf> [Consult. 9 de abril de 2019].
- [64] Redmon J., Farhadi A., 2018. YOLOv3: An Incremental Improvement. *University of Washington* [Em linha].
Disponível em: <https://pjreddie.com/media/files/papers/YOLOv3.pdf> [Consult. 14 de março de 2019].
- [65] Kathuria A., 2018. What's new in YOLO v3? *Medium* [Em linha].
Disponível em: <https://towardsdatascience.com/yolo-v3-object-detection-53fb7d3bfe6b> [Consult. 9 de abril de 2019].
- [66] Redmon J., Farhadi A., 2018. YOLO: Real-Time Object Detection. *arXiv* [Em linha].
Disponível em: <https://pjreddie.com/darknet/yolo/> [Consult. 14 de março de 2019].

- [67] Augmented Startups, 2018. YOLOv3 WorkFlow Tutorial [Em linha].
Disponível em: <https://github.com/reigngt09/yolov3workflow> [Consult. 21 de março de 2019].
- [68] Kathuria A., 2018. A PyTorch implementation of a YOLOv3 Object Detector [Em linha].
Disponível em: <https://github.com/ayooshkathuria/pytorch-yolo-v3> [Consult. 21 de março de 2019].
- [69] Zhai X., 2018. Deep Learning with Raspberry Pi – Real-time object detection with YOLO v3 Tiny! *fun of DIY* [Em linha].
Disponível em: <http://funofdiy.blogspot.com/2018/08/deep-learning-with-raspberry-pi-real.html> [Consult. 28 de março de 2019].
- [70] Portable Efficient Assembly Code-generator in Higher-level Python (PeachPy). *x86-64 assembler embedded in Python* [Em linha].
Disponível em: <https://github.com/Maratyszczka/PeachPy> [Consult. 28 de março de 2019].
- [71] The Ninja build system [Em linha].
Disponível em: <https://ninja-build.org/manual.html> [Consult. 28 de março de 2019].
- [72] Pedestrian Detection [Em linha].
Disponível em: https://github.com/handong1587/handong1587.github.io/blob/master/_posts/computer_vision/2015-09-24-datasets.md [Consult. 3 de junho de 2019].
- [73] Darknet CrowdHuman [Em linha].
Disponível em: <https://github.com/alaksana96/darknet-crowdhuman> [Consult. 3 de junho de 2019].
- [74] Convert the format of the Caltech Pedestrian Dataset to the format that YOLO uses [Em linha].
Disponível em: <https://github.com/simonzachau/caltech-pedestrian-dataset-to-yolo-format-converter> [Consult. 3 de junho de 2019].
- [75] COCO to YOLO [Em linha].
Disponível em: <https://bitbucket.org/yymoto/coco-to-yolo/src/master/> [Consult. 3 de junho de 2019].
- [76] Getting Started with Darknet YOLO and MS COCO for Object Detection. *Comme ci, comme ça* [Em linha].
Disponível em: <https://commecica.com/2018/07/18/starting-with-darknet-yolo-coco/> [Consult. 3 de junho de 2019].
- [77] Nayak S., 2019. Training YOLOv3 : Deep Learning based Custom Object Detector. *Learn OpenCV* [Em linha].
Disponível em: <https://www.learnopencv.com/training-yolov3-deep-learning-based-custom-object-detector/> [Consult. 9 de novembro de 2019].

- [78] Yolo-v3 and Yolo-v2 for Windows and Linux [Em linha].
Disponível em: https://github.com/manishravula/yolov3_tiny_pruned#how-to-train-to-detect-your-custom-objects [Consult. 28 de março de 2019].
- [79] YOLO mark [Em linha].
Disponível em: https://github.com/AlexeyAB/Yolo_mark [Consult. 10 de novembro de 2019].
- [80] Cartucho J., 2019. mAP (mean Average Precision) [Em linha].
Disponível em: <https://github.com/Cartucho/mAP> [Consult. 10 de novembro de 2019].
- [81] Higino H., 2019. Control de tira de LED WS2812b [Em linha].
Disponível em: https://drive.google.com/file/d/1oqnBv09IA_74Xkk1MzS7KOUXmjF5CYY9/view [Consult. 28 de março de 2019].
- [82] Applied Concepts, Inc. 2018. *Stalker Traffic Statistics Sensor - User Manual*, 855 East Collins Blvd. Richardson, Texas. 011-0132-00 Rev. C.
- [83] Zhang Y., Shen Y., Zhang J., 2019. An improved tiny-yolov3 pedestrian detection algorithm. *Optik* (Vol. 183, 17-23) [Em linha].
Disponível em: <https://www.sciencedirect.com/science/article/pii/S003040261930155X> [Consult. 10 de abril de 2019].