Politécnico de Leiria
Escola Superior de Tecnologia e Gestão
Departamento de Engenharia Informática
Licenciatura em Eng.ª Informática

FACE DETECTION AND EMOTION RECOGNITION 4 AAL

CARLA SOFIA CRESPO MENDES, 2191170
RAFAEL MENDES PEREIRA, 2191266

Leiria, Julho de 2022

[11 de abril de 2023 at 22:36 – versão 1.0]

Politécnico de Leiria
Escola Superior de Tecnologia e Gestão
Departamento de Engenharia Informática
Licenciatura em Eng.ª Informática

FACE DETECTION AND EMOTION RECOGNITION 4 AAL

CARLA SOFIA CRESPO MENDES, 2191170
RAFAEL MENDES PEREIRA, 2191266

Trabalho de Projeto da Unidade Curricular de Projeto Informático realizado sob orientação dos Professor Doutor António Manuel de Jesus Pereira (apereira@ipleiria.pt), Professor Doutor Roberto Aguiar Ribeiro (roberto.ribeiro@ipleiria.pt), Professor Doutor José Carlos Bregieiro Ribeiro (jose.ribeiro@ipleiria.pt).

Leiria, Julho de 2022

[11 de abril de 2023 at 22:36 – versão 1.0]

AGRADECIMENTOS

O desenvolvimento do Sistema SmartEmotion contou com o apoio e ajuda dos orientadores, professores, funcionários da instituição, familiares e amigos, de modo que o presente capítulo visa prestar os respetivos agradecimentos.

Primeiramente queremos agradecer ao Professor Doutor António Manuel de Jesus Pereira (apereira@ipleiria.pt), ao Professor Doutor Roberto Aguiar Ribeiro (roberto.ribeiro@ipleiria.pt) e ainda ao Professor Doutor José Carlos Bregieiro Ribeiro (jose.ribeiro@ipleiria.pt) por todo o apoio e ajuda incensáveis que forneceram durante todo o semestre ao longo das diversas fases: investigação, desenvolvimento, testagem e inclusive correção do presente relatório. Desde sempre apresentaram imensa disponibilidade e compreensão e sem a sua ajuda e orientação não seria possível o desenvolvimento de um projeto com esta qualidade e profissionalismo.

Deixamos um agradecimento especial também para o Professor Doutor Luís Alexandre Lopes Frazão (luis.frazao@ipleiria.pt) e Professor Doutor David Ferreira Safadinho (david.safadinho@ipleiria.pt) pela ajuda na aquisição dos equipamentos e suporte técnico dos mesmos, Professor Doutor Rolando Lúcio Germano Miragaia (rolando.miragaia@ipleiria.pt) pela ajuda com conceitos de Inteligência Artificial e verificação científica do modelo presente na nossa solução e ainda pelas ajudas níveis matemáticos. Para além disso agradecemos o apoio do Professor Doutor João Pedro Ferreira Ramos (joao.f.ramos@ipleiria.pt) pelo suporte técnico e intelectual e ainda ao Professor Doutor Ricardo Jorge Pereira Gomes (ricardo.p.gomes@ipleiria.pt) por toda a disponibilidade demonstrada e ajuda na resolução de diversos problemas técnicos e conceituais. Por último, também queremos agradecer ao Suporte DEI pela criação do domínio para a plataforma SmartEmotion e o alojamento da mesma e ainda pela impressão das caixas para os equipamentos.

Por último, queremos agradecer aos amigos e à família pelo apoio emocional fornecido ao longo de todo o semestre, por se disponibilizarem para participar na realização dos testes ao sistema SmartEmotion e pelo tempo despendido nos mesmos.

RESUMO

Este relatório de projeto enquadra-se na UC de Projeto Informático que se encontra incluída no ciclo de estudos da Licenciatura de Engenharia Informática. O projeto, denominado de SmartEmotion incluído-se no âmbito de Ambient Assisted Living (AAL), tem como principal objetivo o desenvolvimento de uma solução inteligente que seja capaz de detetar rostos e reconhecer as emoções de pessoas idosas através de técnicas de Inteligência Artificial e Machine Learning.

Nos dias de hoje familiares idosos passam muito tempo nas suas habitações sozinhos, este é um problema que se tem vindo a agravar com a pandemia COVID-19 onde os mesmos foram impossibilitados de receber visitas de familiares ou amigos. Desta forma a sua solidão e isolamento cresceram ainda mais, o que levanta questões relativas à sua segurança e bem-estar físico e mental. Consequentemente causa preocupação aos cuidadores destes idosos.

A proposta de solução passa por permitir aos cuidadores acompanharem remotamente em tempo-real as emoções das pessoas idosas e alertar os cuidadores quando ocorrer uma alteração do estado emocional do idoso, de acordo com a configuração das notificações, quando estão sozinhos nas suas casas. Desta forma procura-se garantir a saúde e bem-estar dos idosos, providenciando ainda um sentimento de segurança aos cuidadores.

Com vista a solucionar este problema foi desenvolvido um sistema denominado de SmartEmotion, composto por um serviço que acompanha o estado emocional do idoso e por uma plataforma que permite ao cuidador receber alertas da sua alteração. Este permite analisar o ambiente envolvente para questões de segurança física e mental, configurar o sistema tendo em conta as preferências (emoções a notificar, tempo entre as mesmas) e ainda fomentar indiretamente quais as emoções a serem detetadas no futuro, através da classificação de imagens realizada pelo cuidador. Desta forma, a solução poderá ser descrita como personalizável, low-cost e auto-evolutiva.

Com recurso a inteligência artificial foi criado um modelo que deteta emoções com auxílio de visão computacional, que permite detetar emoções personalizadas através da categorização de imagens para retreino do modelo.

Foram realizados testes num ambiente semi-controlado com a duração de um mês, durante o qual diversos rostos e emoções foram detetados e ainda foram enviados diversos *logs* e notificações. Para além disso, a plataforma permitiu aos integrantes das famílias classificarem os *frames* e visualizarem as informações da pessoa idosa ao seu cuidado. Numa das casas de testes foi atingido o limite de *frames* classificados com uma emoção personalizada. Desta forma, o retreino do modelo pôde ocorrer onde se adquiriu mais uma emoção a detetar.

Os testes realizados permitiram comprovar o bom funcionamento da solução e que a sua utilização permite combater a solidão das pessoas idosas em suas casas, na medida em que ajuda os seus cuidadores a acompanharem diariamente o estado emocional e físico da pessoa idosa em tempo real.

Os testes realizados em ambiente semi-controlado tiveram resultados bastante satisfatórios relativamente ao funcionamento do sistema, sendo ainda de ressaltar que as famílias reagiram de forma muito positiva à presença e utilização do sistema no seu dia-a-dia. O sistema SmartEmotion contribui, pois, para garantir o bem-estar e segurança do idoso e a tranquilidade dos seus cuidadores ao acompanharem o seu estado emocional em qualquer hora e em qualquer lugar.

Em suma, a solução concebida carateriza-se por ser *low-cost*, auto-evolutiva devido à capacidade de aprendizagem de novas emoções, e personalizável visto que permite aos utilizadores customizar diversas partes do sistema – onde se destaca a adaptação contínua das emoções a detetar e das caraterísticas para criação de notificações.

Palavras-chave: Machine Learning, Inteligência Artificial, Human Labelling, personalização, auto-evolução, emoções

ABSTRACT

This project report is part of the subject Informatic Project which is included in the study cycle of Computer Science Degree. The project, called SmartEmotion which is included in the concept of Ambient Assisted Living (AAL), has as its main goal the development of an intelligent solution capable of detecting faces and recognizing the emotions of an elder by using Artificial Intelligence and Machine Learning techniques.

In current times elderly family members spend a long time in their residencies alone, this is a problem that has been getting aggravated with the arrival of the COVID-19 pandemic where they were forbidden to receive visits of their relatives or friends. This factor contributed to the rise of their solitude and isolation, which raises safety and both physical and mental well-being issues.

The solution aims to allow the user to accompany remotely and in real-time the emotions of the elder people and alert their caretakers when a change in the elderly' emotional state has occured, according with the configuration of notifications, when they are home alone. In this way, we seek to ensure the health and well-being of the elderly, while also providing a feeling of security for caregivers.

Acknowledging these issues, our solution, SmartEmotion, that consists of a service that accompanies the emotion well-being of the elder person and also a platform that allows the caretaker to receive alerts of its change. This allows the analysis of the surrounding environment for physical and mental security issues, configure the system according to the user's preferences (emotions to notify, time between notifications) and also indirectly influence which emotions the model will detect, through image classification done by the caretaker. This way, the solutions can be described as personalizable, low-cost and self-evolving.

A model was built using Artificial Intelligence that detects emotions with the help of computer vision, which allows the detection of personalized emotions through the categorization of images for training the model.

Tests were conducted under a semi-controlled environment over the course of one month, during which multiple faces and emotions were detected and were also sent various logs and notifications. Furthermore, the platform allowed the family members to classify frames and visualize the informations of the elder in their care.

In one of the houses the limit of frames classified with a personalized emotion was reached. This way, model retraining could occur where one more emotion to detect has been acquired.

The performed tests proved the proper functioning of the solution and that its usage can combat the loneliness of the elderly in their homes, as it helps their caregivers to monitor the emotional and physical state of the elderly person in real time on a daily basis.

The tests performed under a semi-controlled environment presented very satisfactory results regarding the funcioning of the system, whereas it can be emphasized that the families reacted very positively to the presence and usage of the system in their day-to-day. The SmartEmotion system consequently assures the well-being and security of the elder person and the tranquility of their caretakers by accompanying their emotional state anytime, anywhere.

To summarize, the designed solution can be described as low-cost to its users, self-evolving due to the ability to learn new emotions; and customizable considering that it allows its users to personalize multiple parts of the system, where we can highlight the continuous adaptation of emotions to detect and also the charateristics for creating notifications.

Keywords: Machine Learning, Artificial Intelligence, Human Labelling, personalization, auto-evolution, emotions

ÍNDICE

Agradecimentos i
Resumo ii
Abstract iv
Índice vi
Lista de Figuras ix
Lista de Tabelas xii
Lista de Código xiv
Lista de Abreviaturas xv

1 Introdução 1

2 Trabalho Relacionado 3
2.1 Deteção de emoções . . . 3
2.2 Abordagens de Inteligência Artificial para a deteção de emoções . . . 5
2.2.1 *Machine Learning* . . . 6
2.2.2 *Lexicon-based* . . . 7
2.2.3 *Transfer Learning* . . . 7
2.3 Ferramentas existentes no mercado . . . 8
2.3.1 Affectiva . . . 10
2.3.2 Kairos . . . 10
2.3.3 CrowdEmotion . . . 10
2.3.4 nViso . . . 11
2.3.5 EmoVu . . . 11
2.3.6 FaceReader™ . . . 11
2.3.7 API Cloud Vision . . . 12
2.3.8 Amazon Rekognition Image API . . . 12
2.3.9 Azure Media Face Detector . . . 12
2.3.10 Discussão . . . 13
2.4 *Datasets* para reconhecimento de expressões faciais . . . 14
2.5 *Data Augmentation* . . . 15
2.6 *Human Labelling* . . . 16
2.7 Pré-processamento de imagens . . . 16

3 Análise de Modelos Estado da Arte 18
3.1 Deepface 24
3.2 VGG-16 26
3.3 FaceNet 29
3.4 OpenFace 31
3.5 VGGFace 32
3.6 Discussão geral de modelos analisados 33

4 Design e arquitetura da solução 36
4.1 Sistema Elder Home 37
4.1.1 Processo de Previsão de *frames* 38
4.1.2 Processo de catalogação de *frames* 41
4.1.3 Processo de treino do modelo 41
4.2 Plataforma SmartEmotion 41

5 SmartEmotion - Uma solução para reconhecimento de emoções num contexto Ambient Assisted Living 45
5.1 Metodologias e Processo de Desenvolvimento 46
5.2 Sistema Elder Home 46
5.2.1 Processo *AAL.py* 51
5.2.2 Processo Websocket.py 58
5.2.3 Processo Retrain.py 58
5.2.4 Instalação automática do sistema Elder Home e mecanismo de *reboot* automático 59
5.3 Plataforma SmartEmotion 61

6 Testes e Resultados 71
6.1 Testes à qualidade da câmara 71
6.2 Cenário de testes 74

7 Conclusões 84

Bibliografia 86

Apêndices

A Apêndice A 93

Declaração 95

LISTA DE FIGURAS

Figura 1 Pontos críticos do rosto humano. 4
Figura 2 Hierarquia das diversas arquiteturas, imagem adaptada de [2]. 5
Figura 3 Diferenças entre um treino de raiz e um treino usando *Transfer Learning*, imagem retirada de [1]. 8
Figura 4 Congelamento de camadas em *Transfer Learning*. 9
Figura 5 Análise dos estados mentais e emocionais de um condutor, imagem retirada de [5]. 10
Figura 6 *Landmarks* de um rosto, referenciado em [22]. 12
Figura 7 *Landmarks* devolvidas após análise de rosto pela ferramenta Azure Media Face Detector referenciado em [37]. 13
Figura 8 *Data Augmentation*, imagem retirada de [16]. 15
Figura 9 Ajustar tamanho da imagem, imagem retirada de [34]. . . . 17
Figura 10 Ajustar escala de cores da imagem, imagem retirada de [46]. 17
Figura 11 Esquema relativo ao retreino de modelo estado da arte. . . . 19
Figura 12 Esquema relativo ao *dataset* de treino dos modelos estado da arte. 20
Figura 13 Função de ativação *softmax*. 22
Figura 14 Função de ativação *sigmoid*. 23
Figura 15 Gráficos de perda e precisão do modelo DeepFace relativos ao teste 6. 25
Figura 16 Resultados obtidos da avaliação do modelo DeepFace com o *dataset* CK+ usando o modelo criado no teste 6. 25
Figura 17 Esquema relativo ao *dataset* de treino dos modelos estado da arte. 26
Figura 18 Imagem do *dataset* PAINFUL Database com rotação de rosto. 27
Figura 19 Imagem do *dataset* PAINFUL Database onde foi detetada emoção positiva. 27
Figura 20 Gráficos de perda e precisão do modelo VGG-16 relativos ao teste 10. 28
Figura 21 Resultados obtidos da avaliação do modelo VGG16 com o *dataset* CK+ usando o modelo criado anteriormente. 29
Figura 22 Gráficos de perda e precisão do modelo FaceNet. 30

Figura 23 Resultados obtidos da avaliação do modelo FaceNet com o *dataset* CK+ usando o modelo criado anteriormente. 30
Figura 24 Gráficos de perda e precisão do modelo OpenFace. 31
Figura 25 Resultados obtidos da avaliação do modelo OpenFace com o *dataset* CK+ usando o modelo criado anteriormente. 32
Figura 26 Protótipo de deteção das regiões de interesse e *landmarks*. . 36
Figura 27 Arquitetura do Sistema Elder Home com hardware especificado. 37
Figura 28 Arquitetura do Sistema Elder Home. 38
Figura 29 Algoritmo do processo de previsão resultante numa iteração de *frames*. 39
Figura 30 Algoritmo do processamento dedos até despoletar a notificação no microcontrolador. 40
Figura 31 Algoritmo da catalogação de *frames* no microcontrolador. . 41
Figura 32 Algoritmo do processo de treino do modelo no microcontrolador. 42
Figura 33 Arquitetura de camadas do treino do modelo no microcontrolador. 43
Figura 34 Arquitetura da Plataforma SmartEmotion. 43
Figura 35 Proposta de implementação. 45
Figura 36 Diagrama de GANTT representativo do processo de desenvolvimento e investigação da solução. 47
Figura 37 Protótipo de deteção das regiões de interesse e *landmarks*. . 48
Figura 38 Protótipo de deteção de emoções usando *Machine Learning*. 48
Figura 39 Microcontroladores Arduino e Raspberry Pi. 49
Figura 40 Arquitetura do Módulo do sistema IoT. 49
Figura 41 Modelo do DeepFace conjuntamente com *Transfer Learning*. 52
Figura 42 Região de interesse. 54
Figura 43 Algoritmo para o *top10Frames*. 56
Figura 44 Esquema do envio das notificações. 57
Figura 45 Esquema do processo Retrain.py. 60
Figura 46 Raspberry Pi 4 integrado com ventoinha. 61
Figura 47 Raspberry Pi 4 integrado com ventoinha. 61
Figura 48 Tecnologias usadas na Plataforma SmartEmotion. 62
Figura 49 Protocolo de comunicação assíncrona - Websockets na Plataforma SmartEmotion. 63
Figura 50 Autenticação à plataforma SmartEmotion. 64
Figura 51 *Dashboard* da plataforma SmartEmotion. 64
Figura 52 Menus laterais da plataforma SmartEmotion, Administrativo e Cuidador respetivamente. 65

Figura 53 Método de implementação de iterações na plataforma SmartEmotion. 66
Figura 54 Página de Iterações na plataforma SmartEmotion. 66
Figura 55 *Frame* na plataforma SmartEmotion. 67
Figura 56 Página de Frames na plataforma SmartEmotion. 68
Figura 57 Destinos das notificações, plataforma SmartEmotion, Email e SMS. 68
Figura 58 Configuração de notificações na plataforma SmartEmotion. . 69
Figura 59 Páginas de Administradores e Cuidadores respetivamente na plataforma SmartEmotion. 69
Figura 60 Página de *logs* na plataforma SmartEmotion. 70
Figura 61 Disposição de equipamentos para teste de captura de *frames* em diferentes distâncias. 72
Figura 62 Percurso dos resultados obtidos do teste à qualidade da câmara do sistema Elder Home. 73
Figura 63 Distância entre a cadeira e o equipamento de teste à qualidade da câmara, sexto teste - 5 metros. 73
Figura 64 Resultado do teste à qualidade da câmara a 5 metros. 74
Figura 65 Cenário de testes em ambiente semi-controlado. 75
Figura 66 Casa de Teste A. 76
Figura 67 Casa de Teste B. 76
Figura 68 Deteção de objetos. 78
Figura 69 Erro no envio de *frames*. 79
Figura 70 Exemplo de Iterações na habitação A. 80
Figura 71 Número de Notificações na casa A. 81
Figura 72 Exemplo de Iterações na habitação B. 82
Figura 73 Número de Notificações na casa B. 83
Figura 74 Retreino do modelo para uma emoção personalizada. 83
Figura 75 Estrutura relevante da base de dados na Plataforma SmartEmotion. 94

LISTA DE TABELAS

Tabela 1 Ações faciais, adaptado de [24]. 4
Tabela 2 Comparação de *datasets* estado da arte para deteção de expressões faciais. 14
Tabela 3 Caraterísticas da máquina A utilizada nos testes. 18
Tabela 4 Caraterísticas da máquina B utilizada nos testes. 19
Tabela 5 Distribuição de imagens por classe para treino de modelos estado da arte. 20
Tabela 6 Testes ao modelo Deepface realizados na máquina B. 24
Tabela 7 Resultados obtidos dos testes ao modelo DeepFace realizados na máquina B. 24
Tabela 8 Testes ao modelo VGG16 realizados na máquina A. 27
Tabela 9 Resultados obtidos dos testes ao modelo VGG-16 realizados na máquina A. 28
Tabela 10 Teste ao modelo FaceNet realizados na máquina B. 29
Tabela 11 Resultados obtidos dos testes ao modelo FaceNet realizados na máquina B. 29
Tabela 12 Teste ao modelo OpenFace realizados na máquina A. 31
Tabela 13 Resultados obtidos dos testes ao modelo OpenFace realizados na máquina A 31
Tabela 14 Teste ao modelo VGGFace realizados na máquina B. 32
Tabela 15 Métricas de comparação entre avaliação de modelos com o *dataset* CK+. 34
Tabela 16 Métricas dos parâmetros dos melhores testes de cada modelo para o *dataset* FER-2013. 34
Tabela 17 Métricas dos resultados dos melhores testes de cada modelo. 34
Tabela 18 Métricas dos resultados de cada modelo com *datasets* CK+ e PAINFUL Database. 35
Tabela 19 Métricas dos *datasets* de testes. 35
Tabela 20 Especificações da webcam Xiaomi Imilab Full HD. 72
Tabela 21 Resultados obtidos do teste à qualidade da câmara do sistema Elder Home. 72

Tabela 22 Resultados obtidos do teste à qualidade da câmara do sistema Elder Home. 73

LISTA DE CÓDIGO

Código 1 Parâmetros utilizados para a *Data Augmentation*. 21
Código 2 Ficheiro .env do Sistema Elder Home. 51
Código 3 Parâmetros *default* para o modelo. 53
Código 4 Objeto JSON resultante da previsão do modelo. 54

LISTA DE ABREVIATURAS

3DCNN	3D CNN.
ADAM	Adaptive Moment Estimation.
API	Application Programming Interface – Interface de Programação de Aplicações.
AU	Action Units.
CNN	Convolutional Neural Network.
DL	Deep Learning.
FACS	Facial Action Coding System.
IA	Inteligência Artificial.
LSTM	Long Short-Term Memory.
ML	Machine Learning.
NiN	Network-in-Network.
OS	Operating System.
RoI	Region of Interest.
SBP	Sparse Batch Normalization.

1 INTRODUÇÃO

A solidão dos idosos desde cedo constitui um problema para a sociedade, sendo este agravado pela pandemia vivida recentemente, onde as pessoas idosas ao redor do mundo passam os seus dias sozinhas e isoladas sem qualquer supervisão ou visitas. Isto implica que nos momentos em que o idoso apresente dor, tristeza, angústia ou stresse (emoções negativas diretamente ligadas à saúde mental e física) este não possuí nenhum suporte ou apoio que esteja fisicamente ao seu alcance. O facto de os idosos apresentarem uma preferência de continuar a residir em suas casas, ao invés de passarem a morar num lar de idosos onde os níveis de solidão diminuiriam devido ao contacto com outros utentes e funcionários destas instituições, contribui para o agravamento da sua solidão. Mantendo, no entanto, o seu conforto de continuar a residir na habitação onde passaram toda a sua vida.

Existe um grande laço entre a tecnologia e a saúde, estas duas grandes áreas têm-se cruzado ao longo dos anos por motivos óbvios, a tecnologia consegue fornecer uma rapidez e eficácia enorme, poupando recursos humanos. E no desenvolvimento de uma solução para o problema apresentado não será diferente; com a ajuda da tecnologia pretende-se oferecer uma solução que permita acompanhar o estado emocional do idoso e alertar quando ocorrer uma alteração negativa.

Os objetivos deste trabalho consistem no desenvolvimento de um serviço que permita efetuar reconhecimento facial e deteção de emoções, mas que também permita notificar o cuidador de modo a zelar pelo bem-estar físico e mental do idoso ao seu encargo. Tendo isto em conta surgem outros objetivos: nomeadamente a parametrização das emoções a notificar, a categorização das imagens detetadas de forma a que o cuidador defina quais as emoções a serem detetadas no futuro, e a criação e testagem do modelo que permita detetar emoções.

Como resultado deste trabalho foi desenvolvido o sistema SmartEmotion, que permite acompanhar com tecnologias de visão computacional o estado emocional dos idosos e alertar os seus cuidadores de eventos que se considerem ser alvo de atenção por parte dos cuidadores. Este sistema incluí dois módulos: um que estará presente nas habitações dos idosos com o objetivo de detetar emoções expressadas e efetuar o treino de modelos para uma personalização de sistema à medida do idoso;

e uma plataforma que permite tanto a idosos como a cuidadores parametrizar o sistema e visualizar a evolução das emoções do idoso. Este sistema foi testado em ambiente semi-controlado, tendo a sua aplicabilidade verificada num cenário real.

Deste projeto resultou um conjunto de contribuições relevantes, com destaque para: a deteção de emoções em tempo real com um modelo de *machine learning* e visão computacional; a criação de uma infraestrutura que permite retreinar o modelo para a personalização/adaptação ao idoso; o desenvolvimento de uma plataforma para alertar sobre alterações e monitorização do estado emocional do idoso; a proposta de um sistema IoT *low-cost* preparado para ser exposto a cenários reais e futuras expansões. Considera-se que esta solução constitui um avanço significativo no sentido de proporcionar autonomia aos idosos, uma vez que lhes permite permite passar mais tempo em suas casas sendo que, caso algo de inesperado ou anormal aconteça, os seus cuidadores serão alertados para que possam agir em conformidade.

O restante documento encontra-se organizado conforme descrito em seguida. O Capítulo 2 focará em conceitos relativos ao reconhecimento facial e deteção de emoções, de seguida serão apresentadas algumas das abordagens para a resolução deste problema. Ainda neste capítulo serão analisadas as ferramentas que existem no mercado que apresentem semelhanças técnicas com o sistema que se pretende desenvolver. Por último, este capítulo irá fornecer alguma informação relativa a técnicas e conceitos que serão de elevada importância nos capítulos que se seguem. O Capítulo 3 aborda os testes realizados aos modelos estado da arte e as respetivas conclusões para a implementação da solução. O foco do Capítulo 4 consiste em introduzir a solução desenvolvida, abordar o processo de desenvolvimento, demonstrar a sua estrutura, tecnologias e análise da implementação. Os requisitos, arquitetura e funcionalidades presentes que constituem a solução desenvolvida são detalhados ainda neste capítulo. O próximo Capítulo 5 apresenta os detalhes de implementação através da análise de cada módulo que constituí a solução concebida. O Capítulo 6 consiste na análise e descrição dos testes realizados em ambiente semi-controlado e dos resultados obtidos. O último Capítulo deste documento, 7 abordará a conclusão do trabalho.

2 TRABALHO RELACIONADO

O problema em causa: reconhecimento facial e deteção de emoções implica a aquisição de alguns conhecimentos relevantes, tanto ao longo deste capítulo como do restante documento de forma a que seja possível compreender a solução concebida e todo o processo de desenvolvimento. Desta forma, inicialmente será detalhado o conceito da deteção de emoções, seguindo-se a análise das arquiteturas existentes que suportam o problema. Posteriormente serão abordadas algumas das ferramentas comercializadas que já satisfazem um problema semelhante. Tendo em conta a grande relevância dos *datasets* para a construção de modelos serão apresentados e caraterizados alguns dos mais conhecidos. Os conceitos de *data augmentation*, *human labelling* e pré-processamento de imagens serão abordados ao longo deste capítulo devido à sua relevância para a proposta de solução posteriormente apresentada.

2.1 DETEÇÃO DE EMOÇÕES

A demonstração de um sentimento pode ser feito de várias maneiras como por exemplo, por voz, escrita ou expressões faciais. Nesta secção serão abordados conceitos fundamentais da deteção de emoções com o objetivo de conhecer melhor o problema apresentado.

Sentiment Analysis (ou *opinion mining*) é uma técnica de processamento de linguagem natural ou imagens de rostos usada para determinar quando é que a expressão dos dados é positiva, negativa ou neutra. É bastante usada para avaliar o *feedback* dos clientes das empresas [4] [64] [51].

Cada pessoa tem a sua forma de expressar uma emoção isto leva a que seja extremamente difícil (até para um humano) caracterizar todas as expressões faciais [63], um computador de forma a identificar uma emoção, precisa de compreender a forma como os humanos identificam as emoções [23]. A definição de expressões faciais é baseado num sistema, denominado de FACS - *Facial Action Coding Systems*, onde cada movimento do rosto é associado a um código. Desta forma as emoções são baseadas em ações, como se encontra representado na Tabela 1.

Emoção	Ação facial
Raiva	Sobrancelhas baixas e cavadas; Olhar intenso; Queixo Levantado.
Alegria	Cantos da boca levantados em um sorriso; Cantos dos olhos enrugados.
Surpresa	Queixo caído/Boca aberta; Sobrancelhas levantadas; Olhos bem abertos.
Medo	Sobrancelhas levantadas; Pestanas superiores levantadas; Pestanas inferiores tensas.
Dor	Sobrancelhas levantadas; Pestanas superiores levantadas; Pestanas inferiores tensas; Lábios ligeiramente abertos.
Tristeza	Pálpebra superior caída; Perda de foco ocular; Esquinas dos lábios levemente voltadas para baixo.
Ansiedade	Morder o lábio.

Tabela 1: Ações faciais, adaptado de [24].

Estas ações faciais podem ser calculadas com o uso dos pontos críticos de um rosto humano como mostra a Figura 1. A abertura da boca é um excelente exemplo duma ação facial calculável visto que pode ser obtida através da distância entre os pontos centrais da boca.

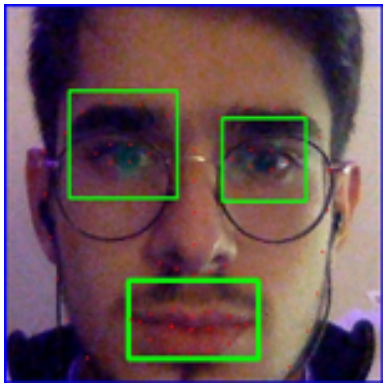


Figura 1: Pontos críticos do rosto humano.

Em suma, compreende-se que a complexidade existente ao redor da deteção de emoções, representadas por expressões faciais, deve-se à unicidade que cada indivíduo tem para expressar sentimentos, e existem vários graus que podem classificar uma emoção (por exemplo dor pode ser representada por uma dor leve, e sucessivos graus até uma dor extrema).

Identifica-se como o principal objetivo da solução a este problema a deteção de dor com uso de técnicas de Inteligência Artificial e visão computacional. A deteção desta emoção é crucial para a prevenção de acidentes irreversíveis, sendo que toda a investigação, solução e desenvolvimento irá ter girar em torno deste requisito.

2.2 ABORDAGENS DE INTELIGÊNCIA ARTIFICIAL PARA A DETEÇÃO DE EMOÇÕES

Esta secção tem como objetivo a introdução e análise das diversas arquiteturas, provenientes da área de Inteligência Artificial. Através da observação da Figura 2, pode-se constatar que a análise de emoções é um tipo de problema que diversas arquiteturas diferentes permitem resolver, sendo que as arquiteturas dividem-se em duas grandes categorias distintas: abordagens de *Machine Learning* e abordagens baseadas em léxico. A estrutura desta secção seguirá a introdução das categorias anteriormente referidas pela mesma ordem. Finalizando com a abordagem de *Transfer Learning* pertencente à categoria menor de outras abordagens, sendo esta é uma abordagem explorada devido à sua relevância para o desenvolvimento da solução permitindo a transferência de conhecimento entre modelos.

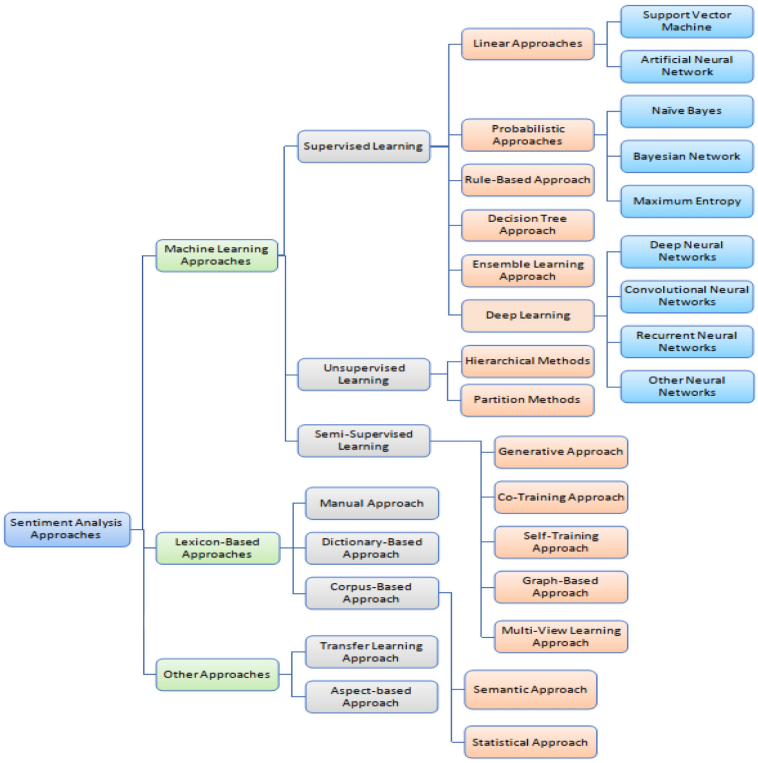


Figura 2: Hierarquia das diversas arquiteturas, imagem adaptada de [2].

2.2.1 *Machine Learning*

Machine Learning, é uma área de estudo de Inteligência Artificial que permite a criação e teste de modelos que visam a descobrir padrões entre os dados. Relativamente ao problema de análise de sentimentos *ML* constitui uma das abordagens mais utilizadas principalmente pela utilização de *datasets* de treino e teste para classificar a polaridade de sentimentos - positivo, negativo ou neutro. [10]

Através do seguimento da Figura 2 é possível concluir que a abordagem de *Machine Learning* pode ser dividida em quatro abordagens mais específicas, nomeadamente aprendizagem supervisionada, aprendizagem não-supervisionada, aprendizagem semi-supervisionada e aprendizagem por reforço.

Resumidamente, a aprendizagem supervisionada permite resolver problemas de classificação onde a caraterística objetivo, denominada de *label*, é conhecida nos dados de treino. Isto permite que durante os treinos, o modelo, que é constituído por um algoritmo, encontre padrões nos dados de forma a atingir denominada classe da caraterística objetivo; e que, em futuras previsões, ao encontrar os padrões referidos anteriormente consiga classificar a nova amostra corretamente, esta classificação designa-se também de previsão. Ainda dentro da aprendizagem supervisionada diversas são ainda as abordagens que poderiam ser usadas para a resolução do problema em questão, sendo *Deep Learning* a mais utilizada e com bastante impacto na solução apresentada por este documento.

Ao contrário das outras abordagens integradas no ramo de *Machine Learning*, onde normalmente são extraídas de forma manual os atributos faciais de uma imagem, a abordagem *Deep Learning* permite a extração dessas mesmas caraterísticas de fora automática. A abordagem *Deep Learning* consiste na utilização de Redes Neuronais Convolucionais, denominadas de CNN, que na sua constituição integram 3 tipos de camadas: *input*, *hidden* e *output*.

As camadas de *input* são também designadas por camadas convolucionais, sendo que a primeira camada convolucional requer pré-processamento mínimo e ao receber uma imagem como *input* é responsável pela aplicação de diversos filtros, o que resulta na extração automática dos atributos da imagem. A camada *hidden* (constituída por diversas outras camadas) é maioritariamente responsável por tornar a deteção de caraterísticas independente da presença de pequenas alterações ou ruído nos dados e por melhorar o processo de convergência durante os treinos. Por último, a camada *output* (também composta por diversas outras camadas) encarrega-se do processo de classificação da imagem. [10]

Os conceitos de *Deep Learning*, nomeadamente o funcionamento e a constituição das Redes Neuronais Convolucionais, apresentam uma elevada importância tendo em conta que a solução desenvolvida implicou a utilização e modificação de uma CNN com vista a detetar emoções.

2.2.2 *Lexicon-based*

Uma das outras abordagens principais de análise de sentimentos é a abordagem baseado em léxico - denominada de *lexicon-based*. Esta calcula o sentimento a partir da orientação semântica de palavras ou frases que ocorrem num texto (*bag-of-words*) com base nas palavras presentes num dicionário (cada palavra deste dicionário tem uma orientação semântica a que corresponde, para além de ter outros fatores como a polaridade, subjetividade, entre outros).

Esta orientação semântica pode ser positiva, negativa ou neutra, à qual estão associados os valores numéricos de 1, -1, e 0, respetivamente. O conceito de polaridade sucede quando uma palavra/frase demonstra um dos sentimentos descritos acima, de forma muito clara, sem deixar quaisquer margens para dúvida. Caso a palavra/frase apresente múltiplos sentimentos então pertencerá ao grupo das palavras não-polares. O dicionário pode ser criado tanto manualmente como de forma automática.

Posto isto, o algoritmo procurará palavras-chave (antónimos e sinónimos) no texto de forma a tentar encaixar as mesmas nos léxicos conhecidos e tirar conclusões da orientação semântica; a abordagem *Lexicon-based* prefere o uso de adjetivos e advérbios para descobrir a orientação semântica.

Esta abordagem é particularmente utilizada para o processamento de linguagem natural em texto e voz, o que não está incluído no âmbito deste projeto. No entanto, no futuro, caso exista uma evolução da solução com a adição de reconhecimento de emoções através de texto ou voz, esta técnica seria uma boa abordagem a adotar.

2.2.3 *Transfer Learning*

Transfer Learning é o nome que se dá à reutilização de um modelo pré-treinado num novo problema. É atualmente bastante popular em *deep learning* porque permite treinar uma rede neuronal com relativamente poucos dados. Isto é bastante importante já que existem vários problemas reais que não têm milhões de dados categorizados para treinar modelos complexos (Figura 3).

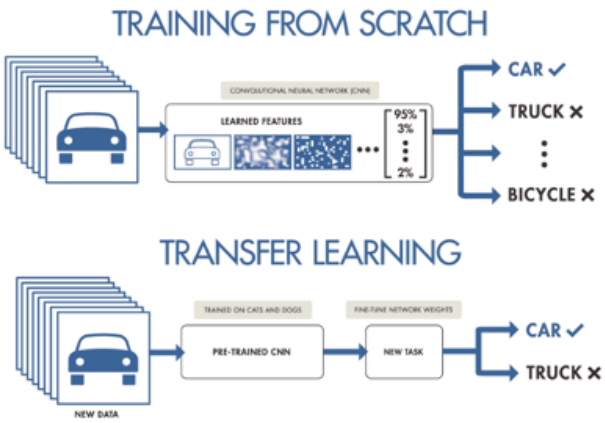


Figura 3: Diferenças entre um treino de raiz e um treino usando *Transfer Learning*, imagem retirada de [1].

Nesta abordagem, o conhecimento que já existe num modelo treinado é aplicado a um problema diferente, mas relacionado. Em suma, tentamos explorar aquilo que já foi aprendido numa tarefa e tentamos generalizar para que se encaixe noutra. Transferimos os pesos que foram treinados num modelo para a "tarefa A"para uma nova "tarefa B".

Uma das técnicas pertencentes ao *Transfer Learning* é chamada *Fine Adjust*. Esta congela as camadas de transformação de características e treina as camadas de classificação (Figura 4). O uso desta técnica torna possível o ajuste de classes de saída, ou por outras palavras, o ajuste ao problema pretendido; a compreensão desta técnica é relevante para o desenvolvimento da solução proposta neste documento [10] [65].

Em suma, o fundamento da abordagem anteriormente referida, *Machine Learning* e *Transfer Learning* são conceitos de maior importância para a descrição da solução apresentada no presente relatório. Estes são aplicados para a deteção e personalização de emoções através da utilização de redes neuronais convolucionais e de *Fine Adjust* (doravante designado simplesmente de "*Transfer Learning*") para a reutilização de um modelo já existente.

2.3 FERRAMENTAS EXISTENTES NO MERCADO

Existem diversas ferramentas no mercado destinadas à deteção de emoções sendo que existem algumas que apenas se focam em análise baseada em léxico processando linguagem natural, nomeadamente: HubSpot Service Hub [32], TalkWalker's Quick

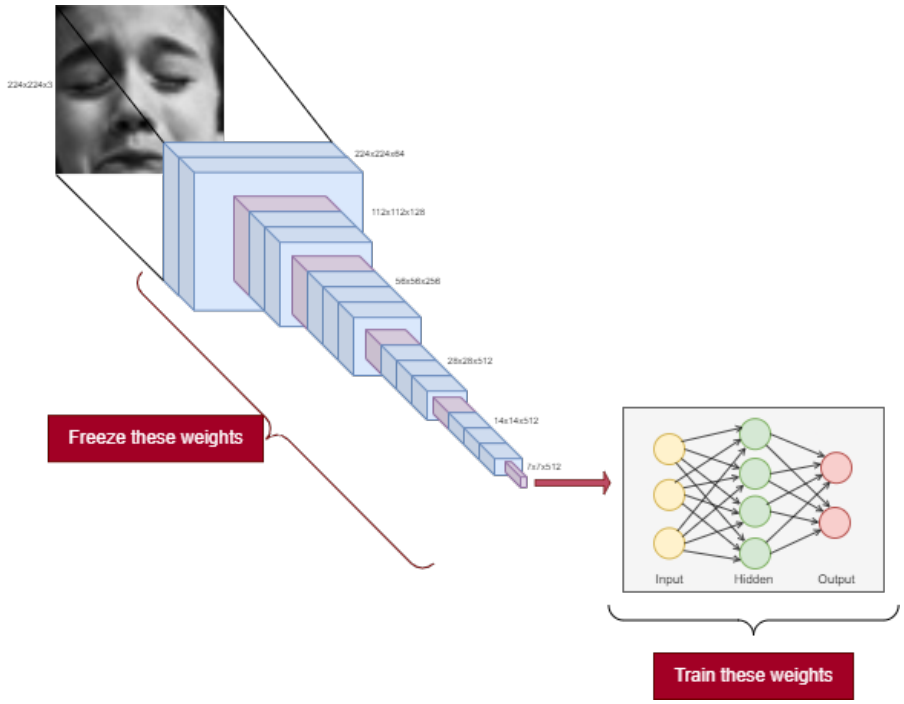


Figura 4: Congelamento de camadas em *Transfer Learning*.

Search [43], Lexalytics [8], MeaningCloud [61], RapidMiner [62], Twiggle [66], IMB Watson [6], BasicAI [15], BrandWatch [54] e MonkeyLearn [70].

Existem também outras ferramentas que são relevantes para o nosso trabalho uma vez que usam visão computacional para a deteção de emoções nomeadamente:

- **Affectiva** - [5]
- **Kairos** - [68]
- **CrowdEmotion** - [7]
- **nViso** - [31]
- **EmoVu** - [20]
- **FaceReader™** - [25]
- **API Cloud Vision** - [17]
- **Amazon Rekognition Image API** - [18]
- **Azure Media Face Detector** - [45]

A presente secção inclui a baixo uma análise pormenorizada a estas ferramentas.

2.3.1 *Affectiva*

Affectiva [5] é uma empresa de *software* que desenvolve diversos serviços na área de Inteligência Artificial. O sistema Affectiva Automotive AI é o seu principal produto e o seu objetivo consiste não só na análise de estados emocionais e físicos dos ocupantes dos veículos (como por exemplo a deteção de sinais de fatiga ou distração no condutor, como se pode constatar na Figura 5) mas permite também perceber quais as atividades que os ocupantes estão a realizar e até quais os objetos com que interagem dentro do veículo, tudo isto com o propósito de aumentar a segurança rodoviária. De forma a conceber este sistema a empresa optou pela utilização de técnicas de *Machine Learning*, aquisição e processamento de dados.

Figura 5: Análise dos estados mentais e emocionais de um condutor, imagem retirada de [5].

2.3.2 *Kairos*

Esta empresa de Inteligência Artificial desenvolveu um *software*, Kairos [68], especializado no reconhecimento facial; no entanto, disponibiliza ainda um vasto conjunto de funcionalidades, nomeadamente a verificação de identidade, deteção de emoções, deteção das coordenadas faciais, e ainda providencia deteção da faixa etária e do género da pessoa, baseando-se na análise de fotos, vídeos ou imagens que leva à criação de um objeto Json. O seu objetivo no mercado consiste na transformação de como os negócios e outras empresas interagem com as pessoas, permitindo a estes compreender melhor os seus clientes.

2.3.3 *CrowdEmotion*

CrowdEmotion [7] é uma empresa que fornece diversos serviços, sendo que a sua API baseada em REST, denominada CrowdEmotion, permite analisar os estados

emocionais presentes em vídeos e apresenta os resultados através de gráficos ou código; sendo facilmente integrada com outros sistemas, algo que se pode observar com a análise da documentação que possui exemplos de implementação para *client-side* em diversas linguagens distintas, como Java, JavaScript, Python, Php.

2.3.4 *nViso*

A empresa nViso, líder mundial em Inteligência Artificial de comportamento humano, desenvolveu um software [31], através da utilização de tecnologia de imagem facial 3D e Inteligência Artificial, com capacidades de captura e interpretação dos movimentos oculares e micro expressões faciais com a finalidade de deteção de emoções. Esta empresa foca-se no desenvolvimento de produtos que visam a transformação de cuidados a pacientes.

2.3.5 *EmoVu*

Desenvolvido pela empresa Eyeris, EmoVu [20] consiste num *software* que também faz uso da análise das micro expressões faciais com o objetivo de não só reconhecer as emoções de uma pessoa como inferir o ambiente que a rodeia. Este *software* utiliza também técnicas de *Machine Learning* que permitem medir de forma precisa o envolvimento emocional e a eficácia de um determinado conteúdo para o público-alvo.

2.3.6 *FaceReader*^{TM}

Noldus é uma empresa que se tem esforçado para avançar na pesquisa comportamental. Fornece sistemas integrados como o FaceReader [25], um *software* que permite efetuar a leitura de expressões faciais e distingui-se através da sua abordagem diferenciada tendo em conta que avalia cada músculo da face e associa uma emoção consoante a expressão apresentada pelo respetivo músculo. FaceReader é um *software* bastante conhecido e usado para estudos de usabilidade. Este disponibiliza outras funcionalidades bastante interessantes que o tornam flexível, objetivo e fácil de usar.

2.3.7 *API Cloud Vision*

Desenvolvida pela Google, API Cloud Vision [17] permite efetuar a deteção de rostos presentes em imagens através da identificação de pontos cruciais da face humana (nomeadamente os olhos, sobrancelhas, boca, nariz, etc.) Estes pontos são definidos como *landmarks* e correspondem aos pontos marcados a verde no rosto presente na imagem da Figura 6.

Figura 6: *Landmarks* de um rosto, referenciado em [22].

O processamento das imagens na API retornam diversas informações, que constam de um objeto Json, sendo as mais relevantes as posições de cada *landmark*, a confiança associada à deteção das *landmarks*, a probabilidade de cada emoção estar presente na foto, a confiança na deteção de um rosto humano e, por último, informações relativas ao uso de objetos na cabeça.

2.3.8 *Amazon Rekognition Image API*

A empresa Amazon desenvolveu uma API [18] que permite detetar rostos, *landmarks* e emoções presentes em fotos ou vídeos. Esta API permite ainda realizar verificação facial. O processamento da imagem retorna a percentagem relativa à confiança da deteção tanto do rosto como das *landmarks* e o carregamento das imagens é realizado através do sistema de ficheiros ou da *Amazon S3 Bucket*.

2.3.9 *Azure Media Face Detector*

Azure Media Face Detector [45] é uma ferramenta desenvolvida pela Microsoft que pode ser utilizada de diversas formas, nomeadamente através do Azure Portal, de uma API ou da ferramenta Azure Media Services Explorer. Esta ferramenta permite detetar múltiplos rostos em vídeos, através da associação de um ID a cada rosto identificado. No entanto, a ferramenta não permite memorizar o rosto de uma pessoa, o que implica se a pessoa sair de quadro no vídeo e voltar a aparecer, será

atribuído um novo ID ao seu rosto. Uma das outras funcionalidade disponibilizadas consiste na análise das *landmarks* do rosto de maneira a detetar a emoção presente no *frame* do vídeo, sendo as *landmarks* descritas na Figura 7.

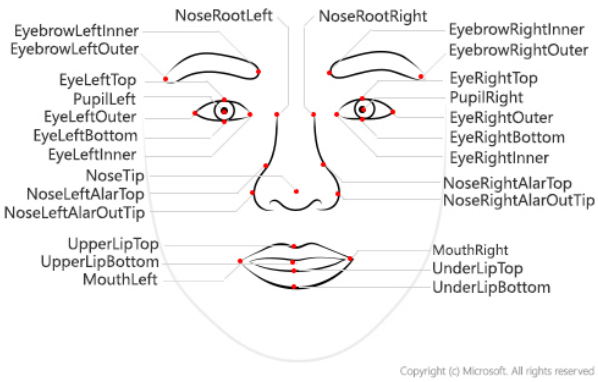


Figura 7: *Landmarks* devolvidas após análise de rosto pela ferramenta Azure Media Face Detector referenciado em [37].

2.3.10 *Discussão*

Nesta secção será sintetizada e filtrada a informação retida na análise às ferramentas existentes no mercado.

De modo geral, todas estas ferramentas têm o mesmo objetivo, deteção de emoções por meio de visão computacional. No entanto, nenhuma permite a deteção da emoções com a granularidade pretendida, nomeadamente detetar a emoção de dor. Devido a serem ferramentas comercializáveis não foi possível ter acesso ao código ou a qualquer mecanismo para usar a abordagem de *Transfer Learning* abordada previamente em 2.2.3.

A API Cloud Vision da Google, o Azure Media Face Detector da Microsoft, Amazon Rekognition Image API e outras estão apenas disponíveis por meio de uma API o que não permite a análise em tempo real pois é preciso fazer pedidos singulares para cada *frame*; e teriam um custo por pedido, não sendo viáveis para o projeto.

Em suma, não existem ferramentas comercializáveis que vão de encontro ao desejado e, desse modo, a solução terá de passar pela implementação própria de um sistema que permita cumprir os objetivos definidos.

2.4 *DATASETS* PARA RECONHECIMENTO DE EXPRESSÕES FACIAIS

O *dataset* é um ponto crucial para o treino de modelos, não existe treino sem um. Desse modo, um *dataset* completo e adequado para o nosso problema terá de conter imagens de expressões faciais que representem predominantemente emoções negativas, como por exemplo: raiva, tristeza, dor, desgosto, ou mesmo surpresa. Na teoria o ideal seria um modelo detetar o máximo de emoções possível; porém, na prática, quanto maior for o número de classes mais difícil será manter uma taxa de erro baixa. Na Tabela 2 encontra-se a comparação de *datasets* bem conhecidos onde a coluna "Nome"indica o nome do mesmo, a coluna "N. Classes"refere-se ao número de emoções (ou seja, o número de classes da caraterística objetivo), sendo que a coluna "Tamanho"corresponde ao número de imagens contidas. A coluna "Acesso cedido"permite indicar se as imagens presentes no *dataset* puderam ser acedidas e, por último, a coluna "Privado"tem como objetivo indicar se o acesso a este é privado ou público.

Nome	N. Classes	Tamanho	Acesso cedido	Privado
FER-2013 [26]	7	35904	Sim	Não
CK+ [41]	7	981	Sim	Não
Multi-PIE [29][3]	-	+750000	Não	Sim
NBC-McMaster Shoulder Pain Expression Archive Database [38]	1	+48000	Sim	Sim
JAFFE [42]	1	+48000	Sim	Sim
LFW [35]	-	+13000	Não	Não
YouTube Faces DB [71]	-	+600000	Não	Não

Tabela 2: Comparação de *datasets* estado da arte para deteção de expressões faciais.

Numa análise aos *datasets* anteriormente referidos conclui-se que a maior dificuldade de um problema de classificação de emoções personalizadas é a qualidade e a quantidade de dados. Todos eles têm características diferentes, seja o tamanho da imagem, a escala de cores, o ambiente onde foram capturadas as imagens, vídeo ou foto, luminosidade ou formato do próprio (csv, mp4, png, etc) *dataset*.

Os *datasets* FER-2013, CK+ e JAFFE são os considerados mais promissores para o nosso caso de estudo. No entanto, o CK+ tem poucas imagens o que poderia causar problemas na precisão do modelo, o JAFFE apenas contém imagens de

pessoas do sexo feminino. Já o *dataset* FER-2013 contém uma quantidade razoável de imagens; no entanto, estão imagens de animais e desenhos animados contidas neste *dataset*, considerado ruído para a solução deste problema.

O único *dataset* que contém imagens representativas de dor ao qual tivemos acesso foi o *dataset* privado NBC-McMaster Shoulder Pain Expression Archive Database [38]; este pertence à universidade do Norte da Colômbia Britânica, é constituído por *frames* retirados de vídeos capturados a pessoas que apresentam dores musculares no ombro. Devido à quantidade de falsos positivos que este *dataset* revelou, tornou-se inútil para treino e para os propósitos pretendidos.

O *dataset* que terá continuidade no nosso projeto será o FER-2013 visto que este contém as melhores características para o desenvolvimento de um modelo que atinja os nossos objetivos. Estas caraterísticas são o elevado número de imagens, a deteção das sete emoções básicas e o facto de este *dataset* ser público e acedível.

2.5 DATA AUGMENTATION

Data Augmentation consiste numa técnica de *data analysis*, exemplificada pela Figura 8, que permite aumentar não só a quantidade mas também a diversidade dos dados presentes num *dataset* através da modificação ligeira de elementos já existentes; que no caso aplicacional das imagens implica criar novas através de transformações aplicadas a antigas, nomeadamente, rotação, recorte, ampliação, alteração de níveis de saturação, brilho, entre outras.

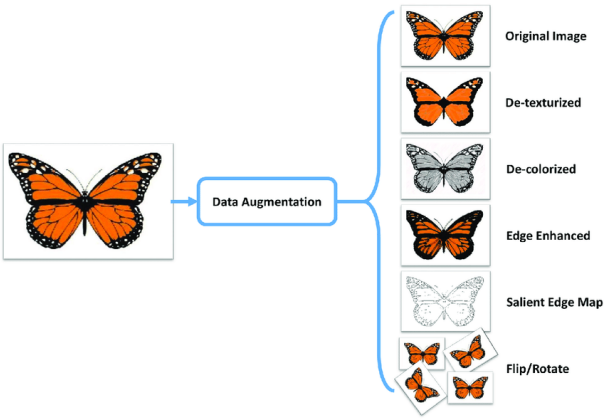


Figura 8: *Data Augmentation*, imagem retirada de [16].

Data Augmentation tem uma relevância bastante elevada para o desenvolvimento da nossa solução, pois é com esta técnica que será criada uma maior variedade e

quantidade de dados antes do treino do modelo concebido para responder ao nosso problema.

2.6 HUMAN LABELLING

Human Labelling é uma técnica para o aumento do *dataset* usando ações humanas para catalogar imagens nas devidas classes. Um *dataset* bem classificado trará resultados bastante promissores na fase de previsão. [47]

Esta técnica pode seguir dois métodos opostos, *Online Learning* usado para referir um modelo que recebe um fluxo contínuo ou sequencial de dados durante o treino, o oposto seria *Offline Learning* (ou *batch learning*), onde o modelo é pré-treinado num *dataset* estático.

Continual Learning, ou aprendizagem contínua (também chamado de aprendizagem incremental ou vitalícia) refere-se a uma parte de *Machine Learning* que trabalha no contexto de *Online Learning* onde os modelos são desenhados para aprender novas tarefas enquanto mantêm o desempenho em tarefas anteriores. Pode ser aplicado em vários paradigmas, como *Class-incremental learning*, onde cada nova tarefa representa uma nova categoria com vista a expandir o número de classes da caraterística objetivo.

Esta técnica irá permitir treinar modelos que são ajustados a cada idoso, onde o cuidador irá influenciar o crescimento do *dataset* usando a técnica de *Human Labelling* em articulação com *Transfer Learning* abordagem apresentada em 2.2.3.

2.7 PRÉ-PROCESSAMENTO DE IMAGENS

O pré-processamento de imagens consiste numa fase que ocorre antes das imagens serem fornecidas ao modelo - quer no processo de treino, quer no de testes. O pré-processamento de imagens permite melhorar os dados presentes na imagem tendo em conta que permite eliminar as partes desnecessárias da imagem, e.g apagar a parte da imagem que apresente objetos no caso do modelo apenas precisar de rostos. O pré-processamento também permite ajustar o tamanho da imagem àquele utilizado pelo modelo (Figura 9), centrar, modificar a escala de cores entre outros (Figura 10).

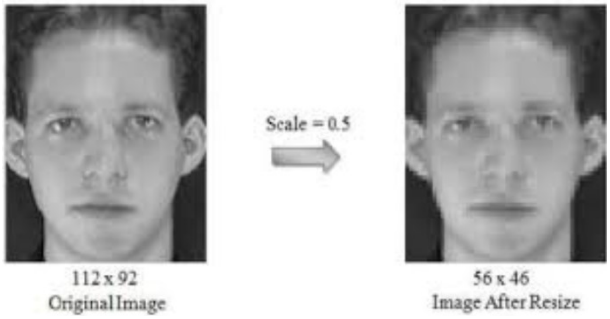


Figura 9: Ajustar tamanho da imagem, imagem retirada de [34].

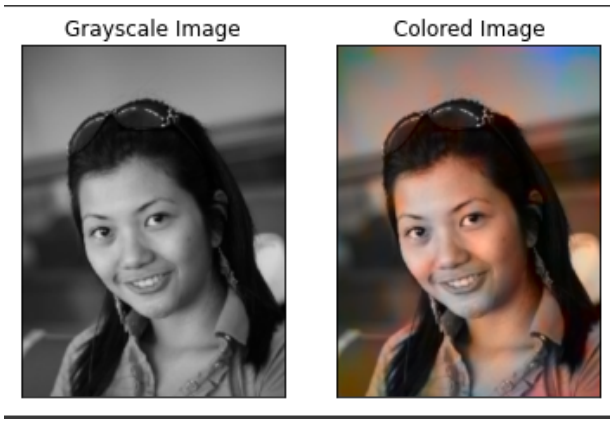


Figura 10: Ajustar escala de cores da imagem, imagem retirada de [46].

3

ANÁLISE DE MODELOS ESTADO DA ARTE

Serve a presente secção para mostrar os resultados de treinos e avaliações de 5 modelos estado da arte, *VGG-16* [14], *FaceNet* [48], *OpenFace* [9], *DeepFace* [52] e *VGG-Face* [53]. Todos estes modelos foram retreinados, e posteriormente avaliados com dados diferentes daqueles dos de treino. Com o objetivo de adquirir um leque de conhecimentos relativos a estes modelos que permita suportar a decisão sobre qual o melhor modelo a usar para a solução iremos relatar cada teste.

Estes modelos foram retreinados várias de forma a encontrar a melhor combinação dos parâmetros: função de ativação, função de perda, métricas de avaliação, número de épocas e número de *batch size* - visando fazer uma previsão de emoções concreta, que nenhum destes modelos foi treinado para fazer. Estes 5 modelos foram então retreinados usando várias técnicas de processamento como *Transfer Learning*, *data augmentation* e *early stopping*. A análise de resultados irá ajudar-nos a criar uma solução para classificação de emoções positivas e negativas, um problema de classificação binária.

As Tabelas 3 e 4, descrevem as caraterísticas das máquinas onde os diversos testes foram realizados, visto que determinadas caraterísticas dos resultados dos testes, como o tempo despendido, são dependentes das máquinas onde são realizados.

Processador	Intel 4 Core i5-8265U CPU @ 1.60GHz 1.80 GHz
Placa Gráfica	NVIDIA GeForce MX130
Memória RAM	8 GB

Tabela 3: Caraterísticas da máquina A utilizada nos testes.

Processador	Intel Core i5 de núcleo duplo
Placa Gráfica	Intel Iris Plus Graphics 640 1536 MB
Memória RAM	8 GB

Tabela 4: Caraterísticas da máquina B utilizada nos testes.

A arquitetura modelada para a análise de modelos estado da arte é representada pela Figura 11. Para a realização dos testes aos modelos referidos usámos a linguagem Python com o auxílio da biblioteca *TensorFlow* [59] devido ao facto de ser uma plataforma *open-source*, escalável que permite visualização dos dados e que ainda possibilita paralelismo [58]. Para além disto, ainda foi utilizado o código base da biblioteca *DeepFace* [52] tendo em conta que o código fonte público permitiu realizar de forma simples e eficaz o processamento de imagens e reconhecimento facial principalmente relembrando que o foco nesta fase do projeto consistia em testar a performance dos modelos. Com isto, para treinarmos um novo modelo precisámos de fazer *download* dos pesos, mudar a estrutura da rede neuronal convolucional e aplicar a técnica de *Transfer Learning* para congelar as camadas de extração de características. Depois, apenas precisámos de treinar as camadas de classificação para que a saída das previsões do modelo sejam ajustadas ao nosso problema.

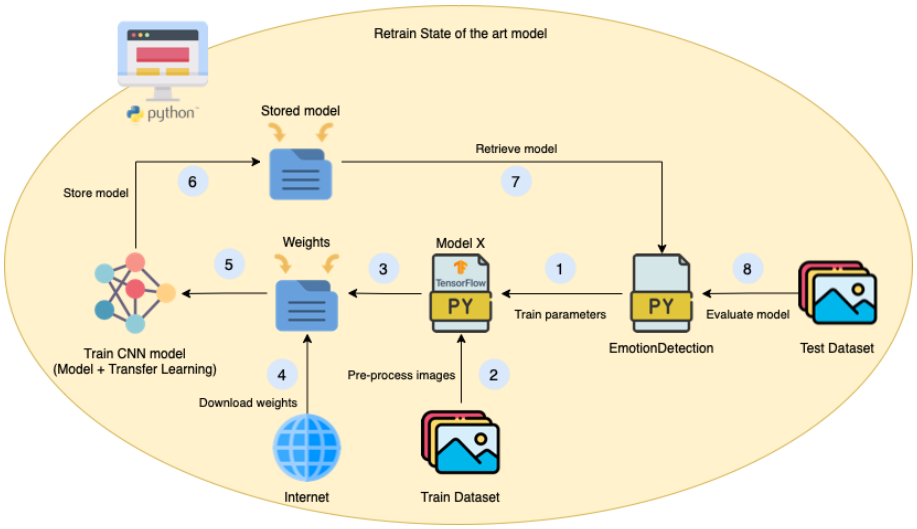


Figura 11: Esquema relativo ao retreino de modelo estado da arte.

Como já referido neste documento existem *datasets* de expressões faciais bem conhecidos como: FER-2013 [26], CK+ [41] e JAFFE [42]. Nestes *datasets* estão presentes sete emoções distintas, mas em nenhum está contida a emoção de dor. Consideramos a deteção da emoção de dor o fator mais importante na resolução do nosso problema.

A emoção de dor está contida no espectro negativo de emoções; e com base nesse espectro, criámos dois *datasets* baseados nos *datasets* FER-2013 e CK+ (respetivamente), onde cada um destes novos *datasets* está dividido em três classes, emoções positivas, neutras e negativas como representado na Figura 12 e na Tabela 5, sendo que durante os testes aos modelos estado da arte foram utilizados ambos os *datasets* criados a partir do FER-2013 e CK+.

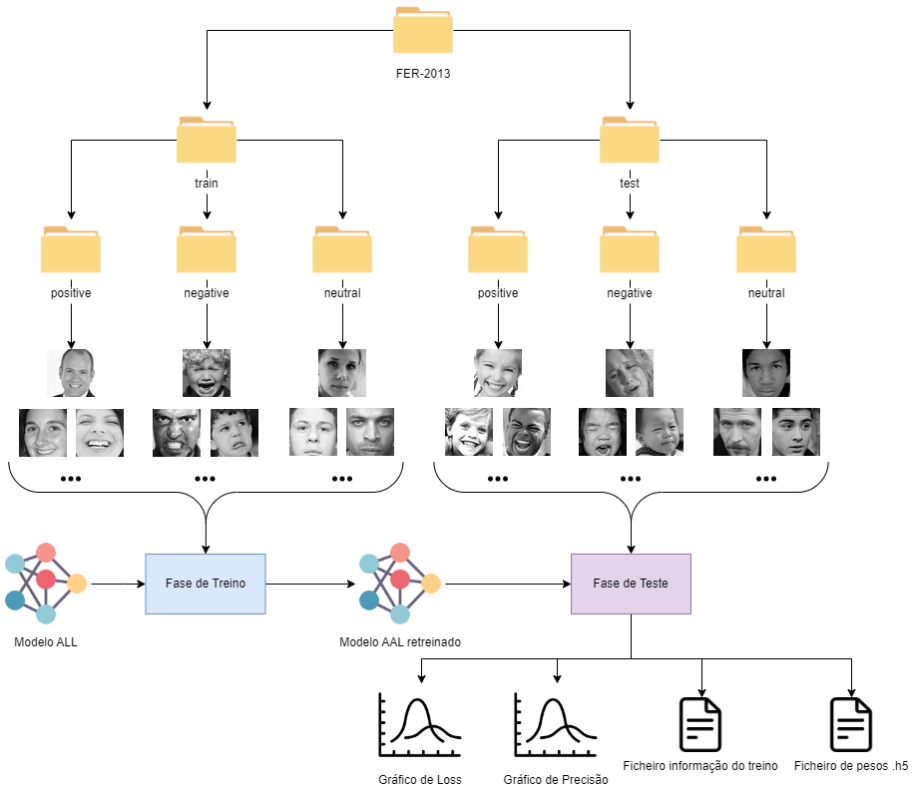


Figura 12: Esquema relativo ao *dataset* de treino dos modelos estado da arte.

A distribuição de imagens por classe é feita da seguinte forma:

Classe Nova	Classes Anteriores	N Imagens
Positive	Happy	8989
Neutral	Neutral	6198
Negative	Angry, Surprise, Sad, Fear, Disgust	20717

Tabela 5: Distribuição de imagens por classe para treino de modelos estado da arte.

De modo a garantir um maior variedade e quantidade dos dados usámos a técnica de *Data Augmentation*, previamente introduzida neste relatório, que implicou a

parametrização de valores. Os parâmetros (representados também no Código 1) foram:

```
roatation_range = 30,
width_shift_range = 0.1,
height_shit_range = 0.1,
shear_range = 0.1,
zoom_range = 0.2,
horizontal_flip = True,
brightness_range = [0.4,1.5]
```

Código 1: Parâmetros utilizados para a *Data Augmentation*.

O treino de um modelo pode levar bastante tempo: desde horas, até dias ou semanas. Porém existem algumas técnicas que permitem reduzir este tempo de treino, como a técnica *early stopping*. Esta técnica permite parar o treino quando este já não está a ser produtivo durante X épocas guardando sempre o melhor resultado [57].

Os modelos estado da arte precisam de ser parametrizados, nomeadamente o número de épocas, número de *batches*, o tipo de *optimizer*, função de *loss*, a função de ativação e as métricas sendo estes parâmetros o foco dos testes demonstrados ao longo desta secção.

Época em Inteligência artificial corresponde a um ciclo da fase de treino onde todos os dados de treino são apresentados ao modelo [19]. Desta forma, o número de épocas irá definir o número de ciclos que o modelo dispõe para a fase de treino. O valor para o número de épocas é normalmente ajustado com base em tentativa-erro, no entanto uma boa prática consiste em utilizar um valor alto para as épocas, (como 256 ou até 500) e conjuntamente utilizar a técnica de *early stopping* descrita anteriormente, sendo que esta técnica encontrará o melhor número de épocas para o treino.

O conceito de número de *batches* corresponde ao número de amostras do *dataset* que serão processadas antes do modelo ser atualizado; os seus valores variam entre 1 e o número de amostras do *dataset*. Desta forma, o *dataset* poderá ser dividido em um ou mais *batches*. Por exemplo, quando uma *batch* termina são calculados os erros relativos à diferença entre os valores previstos e os valores reais. Este erro é usado para atualizar o modelo [19].

A função de *loss* visa a calcular o valor da diferença entre os valores previstos pelo modelo e os valores reais o que indica a qualidade do modelo (ou seja, o quão bem o modelo se está a comportar relativamente ao *dataset*) sendo que o valor obtido por esta função é denominado de *loss* ou perda [30]. Tendo em conta que o

problema de deteção de emoções pertence à classe de problemas de classificação - onde o atributo objetivo é um valor nominal - foram utilizadas as seguintes funções de *loss*: *categorical_crossentropy* (problemas de classificação de múltiplas classes) e *binary_crossentropy* (problemas de classificação binária) [67].

O *optimizer* é um algoritmo matemático que permite alterar os pesos do modelo com vista a minimizar o valor de perda do modelo, ou seja aumentar a *performance* do modelo, isto tendo em conta que permite mudar os pesos e o *learning rate* do modelo. Ao longo dos testes foi utilizado *optimizer* denominado de adam (Adaptive Moment Estimation) que se carateriza por ser o mais popular sendo que apresenta algumas vantagens, nomeadamente ser de fácil implementação, computacionalmente eficiente e necessitar de pouco espaço de memória. [40]

As funções de ativação utilizadas que visam a gerar os valores de *output* da última camada da rede neuronal a partir dos valores da soma ponderada dos seus *inputs*. No caso dos modelos em questão, foram selecionadas as funções *softmax* e *sigmoid* sendo muito utilizadas para problemas de classificação. [39]

A função *softmax* (representada na Figura 13) apresenta o eixo do x como os *inputs* e o eixo do y como *outputs*. Esta função permite normalizar os valores de *output* através da conversão da soma dos pesos dos valores de *input* para um valor de probabilidade sendo, desta forma, utilizado em problemas onde a previsão de cada classe deve ser um valor de probabilidade, como sucede com o problema em questão - onde numa determinada imagem diversas emoções poderão estar representadas.

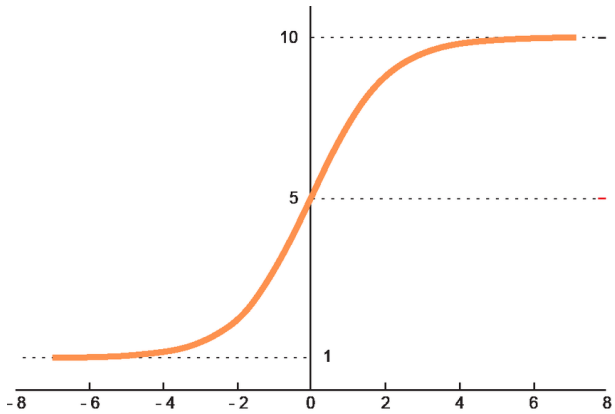


Figura 13: Função de ativação *softmax*.

Relativamente à função de ativação *sigmoid* (representada na Figura 14) onde o eixo dos x corresponde aos valores de *input* e o dos y aos valores de *output*. Sendo uma função para problemas de classificação binária onde a soma dos valores de *output* não precisa de ser 1. Esta função produz valores no intervalo de 0 a 1, como destaca a Figura 14.

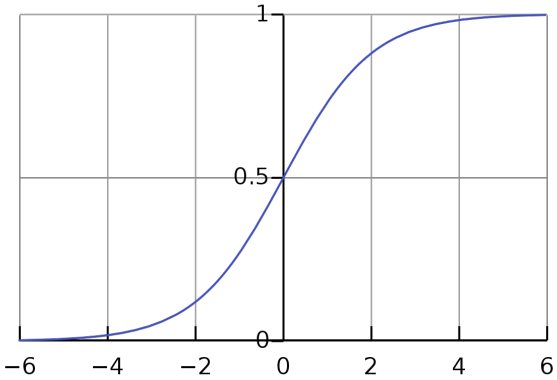


Figura 14: Função de ativação *sigmoid*.

O último parâmetro a abordar consiste nas métricas de avaliação dos modelos, de nome *metrics*, que permitem averiguar a qualidade do modelo [21]. Durante os testes optámos pela utilização das métricas *accuracy* e *binary_accuracy*, cuja única diferença consiste no tipo de problema onde são utilizadas: *accuracy* para um problema de múltiplas classes e *binary_accuracy* para um problema binário. O conceito de *accuracy* corresponde à percentagem de amostras que o modelo classificou corretamente. As métricas são dependentes da função de *loss*, descrita anteriormente, o que implica que para a função de perda *categorical_crossentropy* é utilizada a métrica de *accuracy* e à função *binary_crossentropy* está associada a métrica *binary_accuracy*.

Cada teste realizado aos respetivos modelos resulta num ficheiro de pesos, extensão h5, cujo padrão de nome consiste em: <nome do modelo>_v<número do teste>_<modo>_<número de épocas>_<número de *batches*>.h5. Resulta ainda num gráfico de perda de informação, num de precisão e num ficheiro com o output de treino do modelo.

O ficheiro dos pesos tem como finalidade ser usado futuramente para fazer previsões de dados sem ter de-se voltar a treinar o modelo exaustivamente. O gráfico de perda corresponde ao percurso de aprendizagem do modelo; isto quer dizer que este pode "saber"demasiado (*Overfitting*), "saber"de menos (*Underfitting*) e "saber"o suficiente, o que é o ideal [12]. O gráfico de precisão demonstra o quão preciso o modelo foi durante o treino.

3.1 DEEPFACE

Desenvolvido pela empresa Meta Platforms, Inc. (previamente designada por Facebook), este modelo bastante versátil permite não só a deteção de emoções como também de outras caraterísticas, nomeadamente deteção do género, idade e raça das pessoas presentes nas imagens. Este modelo, durante o seu desenvolvimento foi treinado com diversos *datasets* (mencionados anteriormente no documento 2.4) sendo eles:

- *Labeled Faces in the Wild Dataset* [35]
- *Youtube Faces Database* [71]
- *Social Face Classification Dataset* [56]

Na Tabela 6 estão presentes todos os testes realizados ao modelo referenciado, onde cada teste implica a variação dos valores dos parâmetros - conFigurações do modelo, desde o número de épocas, número de batches, tipo de *optimizer*, função de *loss* e métricas.

Nº Teste	Nº Épocas	Nº *Batches*	*Optimizer*	Função de *Loss*	Função de Ativação	Métricas	*Dataset*
1	10	128	adam	categorical_crossentropy	softmax	accuracy	FER-2013
2	10	128	adam	categorical_crossentropy	softmax	accuracy	CK+
3	10	128	adam	binary_crossentropy	softmax	binary_accuracy	CK+
4	10	128	adam	binary_crossentropy	softmax	binary_accuracy	FER-2013
5	100	256	adam	binary_crossentropy	softmax	binary_accuracy	FER-2013
6	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013

Tabela 6: Testes ao modelo Deepface realizados na máquina B.

Nº Teste	Precisão Mínima	Precisão Máxima	Perda Mínima	Perda Máxima	Tempo (Segundos)
1	62%	63%	0.88	0.96	360
2	30%	45%	0.70	0.80	30
3	41%	48%	0.70	0.74	30
4	62%	75%	0.52	0.72	691
5	68%	75%	0.52	0.70	-
6	52%	75%	0.52	0.68	4718

Tabela 7: Resultados obtidos dos testes ao modelo DeepFace realizados na máquina B.

Com base nos resultados dos testes ao modelo DeepFace representados pela Tabela 7 conclui-se que este modelo teve um desempenho bastante bom, onde a precisão chegou a 75% em vários testes (Figura 15). Este modelo tem uma complexidade mais baixa relativamente aos restantes e por essa razão teve um tempo de treino bastante

mais reduzido em todos os testes; esta característica foi crucial para escolhermos o modelo a usar na nossa solução.

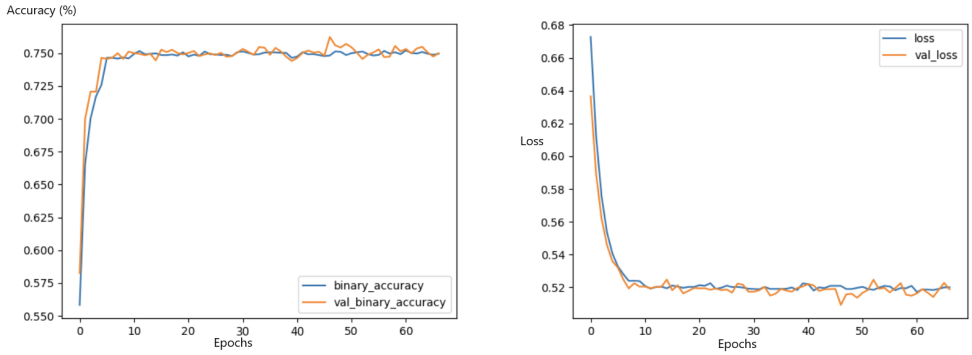


Figura 15: Gráficos de perda e precisão do modelo DeepFace relativos ao teste 6.

Uma vez que o modelo DeepFace ia de encontro com os requisitos para solução do problema referenciado por este documento foi avaliado o modelo criado pelo sexto teste com as imagens do *dataset* CK+, estas imagens não são conhecidas pelo modelo. Os resultados são apresentados na Figura 16.

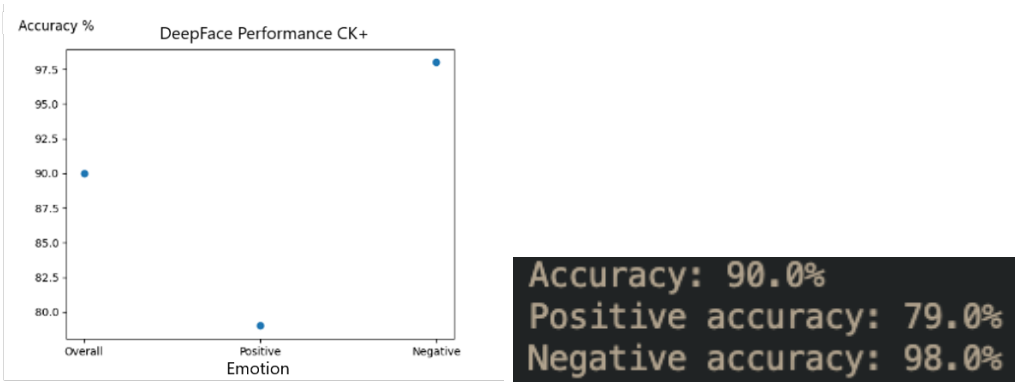


Figura 16: Resultados obtidos da avaliação do modelo DeepFace com o *dataset* CK+ usando o modelo criado no teste 6.

Na previsão feita ao *dataset* CK+ (Figura 16) obtivemos uma precisão média de 90%, o que é impressionante dados os 75% de precisão que o modelo mostrou no seu treino. A precisão nas imagens positivas foi de 79%, o que continua a ser bastante elevado apesar de ser 10% abaixo da média; este valor poderia tornar-se mais elevado se o *dataset* de treino contivesse mais imagens correspondentes a emoções positivas (o modelo foi treinado com 7216 imagens de emoções positivas). Já para as emoções negativas o *dataset* teve uma prestação incrível, com 98% de precisão. Esta precisão elevada em emoções negativas deve-se ao número de imagens negativas ao qual o modelo foi exposto: 16529 imagens. Esta diferença de precisão

deve-se à diferença do número de imagens entre classes. O modelo não conseguiu prever a emoção neutra pois esta não está presente no *dataset* CK+.

Estes resultados são ideais para o objetivo do projeto, a precisão na deteção de emoções negativas é um ponto crítico para nós. Sublinhe-se que uma vez que com 98% de precisão num teste com imagens não conhecidas pelo modelo é surpreendente este resultado.

O modelo foi ainda avaliado com o *dataset* NBC-McMaster Shoulder Pain Expression Archive Database [38]. Este como referido 2.4, não nos foi útil para treinar qualquer modelo; porém será usado como método de avaliação para o modelo DeepFace. Este *dataset* contém bastante ruído, pelo que é de esperar alguma dificuldade no acerto da previsão.

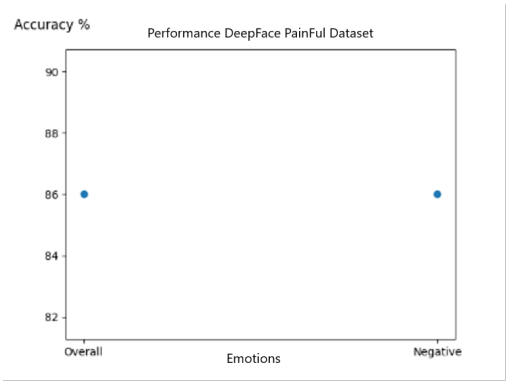


Figura 17: Esquema relativo ao *dataset* de treino dos modelos estado da arte.

Podemos observar na Figura 17 percentagens de 86% de precisão nesta avaliação, o que é bastante bom e coerente com a avaliação anterior. Apesar destes 86% serem inferiores aos resultados da avaliação com o *dataset* CK+, temos de ter em conta que das 48398 imagens que este *dataset* tem em 3085 o algoritmo não detetou o rosto devido à rotação da cara e outros fatores (Figura 18). Tivemos 3750 imagens onde foi detetada uma emoção positiva (Figura 19).

Em suma, e como já referido, este modelo será usado na solução proposta uma vez que teve, de maneira geral, as melhores características, como a alta precisão, o baixo tempo de treino e a baixa complexidade.

3.2 VGG-16

VGG-16 é um modelo que usa uma rede neuronal convolucional e que foi criado por Karen Simonyan e Andrew Zisserman [11]. Este modelo conseguiu uma precisão

Figura 18: Imagem do *dataset* PAINFUL Database com rotação de rosto.

Figura 19: Imagem do *dataset* PAINFUL Database onde foi detetada emoção positiva.

de 92.7% com o *dataset* ImageNet (o *dataset* mais completo relativamente a imagens) [33]. Note-se que o VGG-16 foi treinado durante semanas utilizando placas gráficas NVIDIA Titan Black. Este modelo é muito usado para problemas de deteção e classificação de objetos e é muito fácil de se-lhe aplicar *Transfer Learning*.

Na Tabela seguinte estão presentes todos os testes realizados ao modelo referenciado, onde cada teste implicou a variação dos valores dos parâmetros - configurações do modelo, desde o número de épocas, número de batches, tipo de *optimizer*, função de *loss* e métricas. Dado que este modelo é muito mais complexo e exige muito mais capacidade de processamento os testes irão começar com valores de parâmetros mais baixos.

Nº Teste	Nº Épocas	Nº *Batches*	*Optimizer*	Função de *Loss*	Função de Ativação	Métricas	*Dataset*
1	10	32	adam	binary_crossentropy	sigmoid	binary_accuracy	CK+
2	10	16	adam	binary_crossentropy	sigmoid	binary_accuracy	CK+
3	10	64	adam	binary_crossentropy	sigmoid	binary_accuracy	CK+
4	20	32	adam	binary_crossentropy	sigmoid	binary_accuracy	CK+
5	20	64	adam	binary_crossentropy	sigmoid	binary_accuracy	CK+
6	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013
7	500	256	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013
8	500	64	adam	categorical_crossentropy	sigmoid	accuracy	FER-2013
9	500	128	adam	categorical_crossentropy	sigmoid	accuracy	FER-2013
10	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013

Tabela 8: Testes ao modelo VGG16 realizados na máquina A.

Nos primeiros seis testes a este modelo foi usada como ferramenta de treino o Google Colaboratory [28], uma plataforma que permite executar código Python. Esta ferramenta estava a ser usada na versão gratuita o que levava a problemas de desempenho e lacunas no *output* sendo que estão representados na Tabela 8 os cinco testes realizados na plataforma. A utilização do Google Colaboratory [28] implicou utilizar valores pequenos para os parâmetros (como se pode observar na diferença

de valores para *epochs* e *batches* entre os testes 5 e 6 e.g). Qualquer tentativa para aumentar os valores dos parâmetros resultava no excesso do limite de memória RAM da conta o que levou ao abandono da ferramenta.

Nº Teste	Precisão Mínima	Precisão Máxima	Perda Mínima	Perda Máxima	Tempo (Segundos)
1	40%	75%	5	35	-
2	40%	80%	6	18	-
3	40%	80%	5	30	-
4	40%	85%	4	25	-
5	45%	75%	2.5	20	-
6	65%	75%	0.5	2.5	-
7	65%	75%	0.5	2.25	26497
8	49%	60%	1	4	9215
9	46%	60%	1	4	12717
10	62%	73%	0.6	2	11252

Tabela 9: Resultados obtidos dos testes ao modelo VGG-16 realizados na máquina A.

Com base dos testes apresentados na Tabela 8 e nos resultados obtidos representados pela Tabela 9 conclui-se que a complexidade do modelo leva a tempos muito altos de treino, apesar de termos obtido valores elevados de precisão.

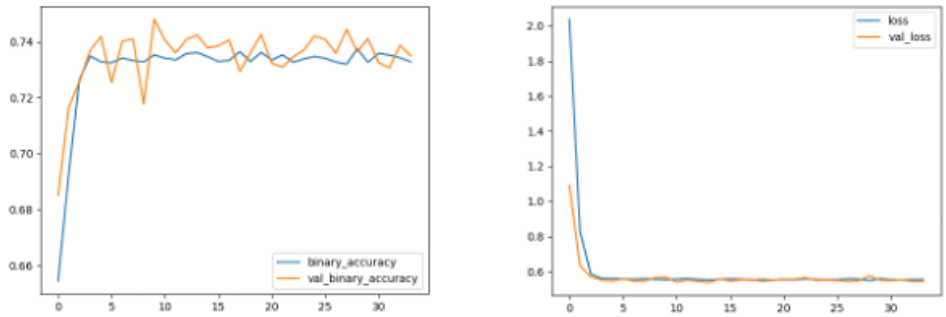


Figura 20: Gráficos de perda e precisão do modelo VGG-16 relativos ao teste 10.

O teste que revelou ser mais promissor, tendo em conta o *dataset* de treino e os resultados apresentados, é o décimo teste, cujos resultados se podem observar na Figura 20, sendo que foi ainda realizada uma avaliação deste modelo perante o *dataset* CK+. Os resultados obtidos nesta avaliação foram os seguintes: precisão média de 57%, onde classificou todas as imagens como negativas (100% 'negative' *accuracy* e 0% 'positive' *accuracy*) como consta da Figura 21. Perante esta inconsistência na avaliação com dados novos e complexidade do retreino do modelo conclui-se que o VGG-16 não será usado para a nossa solução.

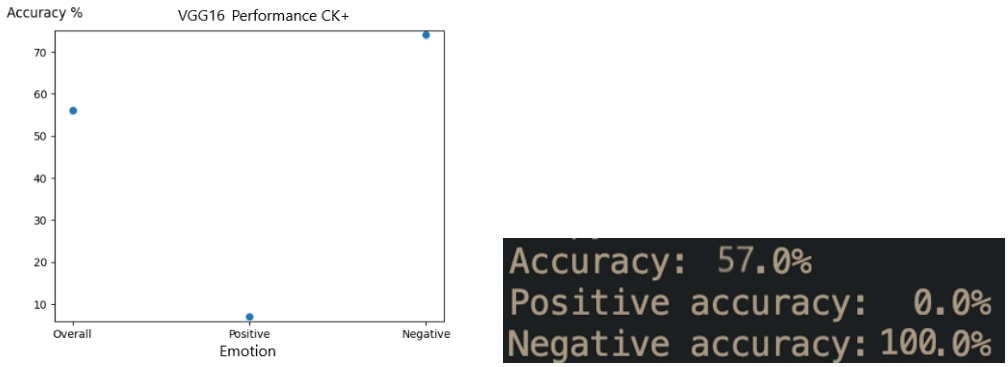


Figura 21: Resultados obtidos da avaliação do modelo VGG16 com o *dataset* CK+ usando o modelo criado anteriormente.

3.3 FACENET

Desenvolvido pela Goggle, FaceNet [49] é um sistema de reconhecimento de rostos, sendo que foi treinado inicialmente utilizando os *datasets Labeled Faces in the Wild Dataset* [35] e *Youtube Faces Dataset* [71].

Nº Teste	Nº Épocas	Nº *Batches*	*Optimizer*	Função de *Loss*	Função de Ativação	Métricas	*Dataset*
1	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013

Tabela 10: Teste ao modelo FaceNet realizados na máquina B.

Nº Teste	Precisão	Perda	Tempo (Segundos)
1	73%	0.55	$\sim$ 54000

Tabela 11: Resultados obtidos dos testes ao modelo FaceNet realizados na máquina B.

Relativamente a este modelo estado da arte apenas foi realizado um teste, que consta da análise da Tabela 10, tendo em conta o tempo elevado de duração do treino: aproximadamente 15 horas (54000 segundos) para 56 épocas (o parâmetro foi configurado inicialmente para 500, mas utilizando o *early stopping* o modelo treinou menos épocas).

Através da análise da Figura 22 é visível que durante as primeiras 10 épocas a precisão aumentou significativamente e até ao final do treino não existiu mais nenhum aumento de realçar, apenas oscilações. O valor de perda, também observável na Figura 22, apresentou uma descida brusca durante as 10 primeira épocas estagnando de seguida.

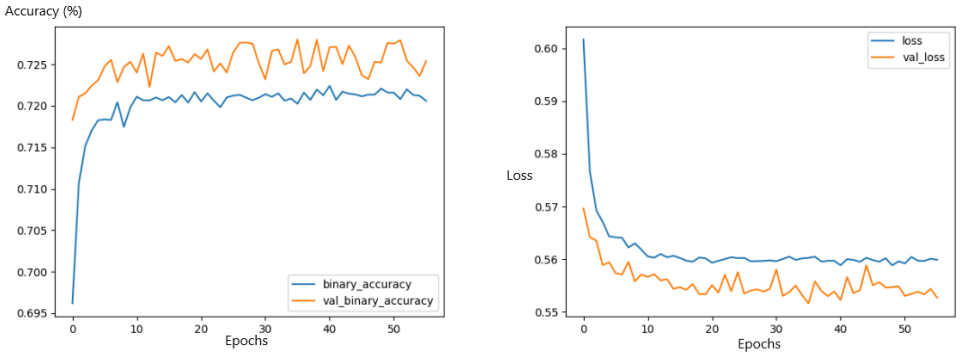


Figura 22: Gráficos de perda e precisão do modelo FaceNet.

Relativamente à precisão obtida, 73% que se pode observar na Tabela 10, o valor foi ligeiramente pior aos modelos anteriores, o que permitiu concluir que este modelo demorou muito mais tempo que os outros modelos principalmente quando comparado por exemplo com o sexto teste do DeepFace, que constou das mesmas configurações de parâmetros, e demorou apenas 1 Horas e 20 minutos (aproximadamente).

O modelo FaceNet foi testado com o *dataset* CK+ para averiguar a sua performance. Através da análise da Figura 23 é conclusivo que a precisão geral foi péssima, apenas de 30%, e que para a classe *Positive* atingiu valor de 21%, e para *Negative* um valor ligeiramente superior, de 37%.

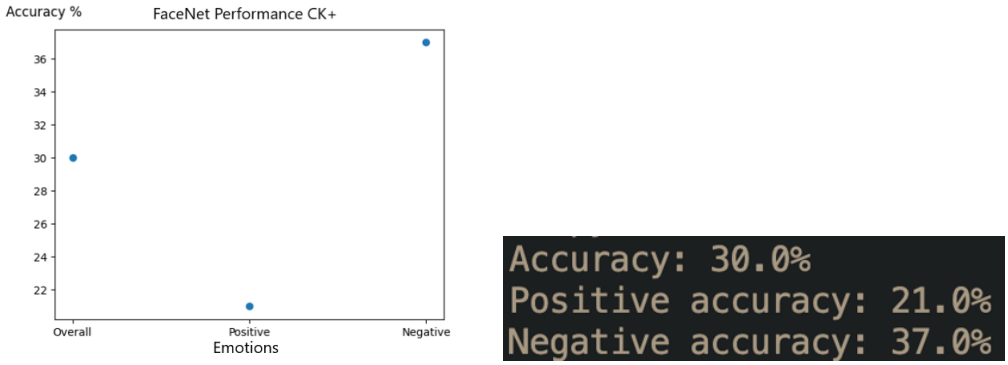


Figura 23: Resultados obtidos da avaliação do modelo FaceNet com o *dataset* CK+ usando o modelo criado anteriormente.

Tendo em conta a conjunção de diversos fatores abordados previamente, nomeadamente o tempo elevado de treino, *accuracy* de treino ligeiramente inferiores a outros modelos e *performance* de teste péssima, conclui-se que para o projeto em causa este modelo torna-se inviável.

3.4 OPENFACE

OpenFace [9] é uma ferramenta *open-source* para reconhecimento de rostos, deteção de *landmarks* do rosto, estimar a pose da cabeça, reconhecimento de *FACS* - abordadas previamente na secção 2.1 - e estimativa de olhar.

Os parâmetros de teste a este modelo (Tabela 12) foram baseados nos melhores resultados dos testes a modelos anteriores. Assim é expectável que os resultados se assemelhem aos demais como é visível na Tabela 13.

Nº Teste	Nº Épocas	Nº *Batches*	*Optimizer*	Função de *Loss*	Função de Ativação	Métricas	*Dataset*
1	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013

Tabela 12: Teste ao modelo OpenFace realizados na máquina A.

Nº Teste	Precisão	Perda	Tempo (Segundos)
1	72%	0.55	$\sim$ 16920

Tabela 13: Resultados obtidos dos testes ao modelo OpenFace realizados na máquina A

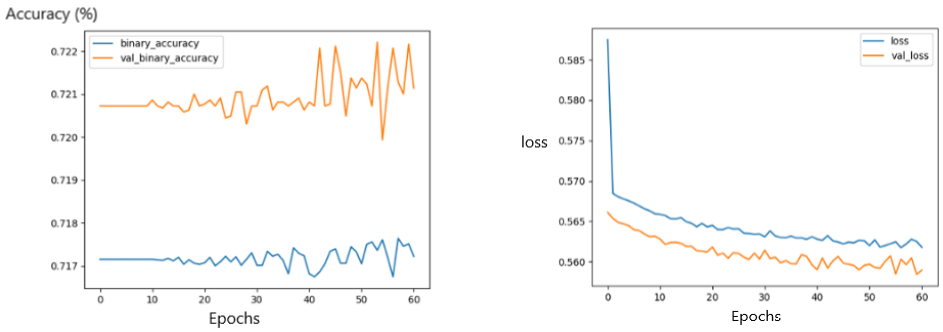


Figura 24: Gráficos de perda e precisão do modelo OpenFace.

Este teste teve uma peculiaridade comparado com os testes dos outros modelos: as curvas de ambos os gráficos - representados na Figura 24 - nunca se cruzam. Algo interessante é que os valores de *loss* iniciais são muito mais pequenos quando comparados com outros modelos (e.g VGG16 que começa com valores mais elevados). Os valores da precisão são, no entanto, idênticos aos valores obtidos através dos outros modelos. O tempo de retreino foi também elevado, cerca de 4 horas e 50 minutos, principalmente quando comparado com o DeepFace, cujo retreino apresentou tempos muito menores para valores de *accuracy* melhores também.

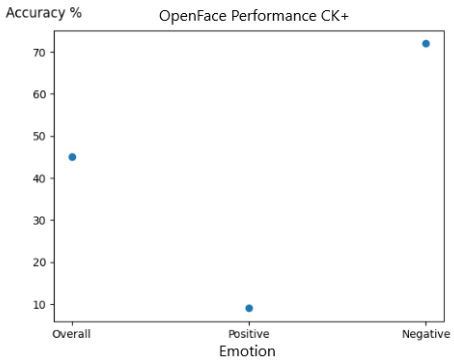


Figura 25: Resultados obtidos da avaliação do modelo OpenFace com o *dataset* CK+ usando o modelo criado anteriormente.

A Figura 25 apresenta os resultados ao teste realizado envolvendo este modelo e o *dataset* CK+, sendo que a sua precisão geral foi de 45%, onde classificou as imagens com emoção *Negative* com uma precisão de 72% e as de *Positive* com uma precisão muito baixa (de apenas 9%). Tendo em conta este fator, a precisão das imagens negativas pode ser considerada alta, mas a precisão das imagens positivas foi demasiado baixa e tendo em conta também o tempo de retreino elevado, a utilização deste modelo não traria vantagens para a solução.

3.5 VGGFACE

A rede neuronal convolucional VGG-Face é codificada com base na arquitetura da rede VGG-Very-Deep-16 e avaliada com rostos em ambientes sem restrições e rostos de vídeos do Youtube. Devido a testes anteriores temos conhecimento que os modelos com base em arquiteturas VGG são modelos complexos e, por isso, demorados na sua fase de treino/retreino. Os testes a este modelo tiveram como base os parâmetros que demonstraram melhores resultados de precisão em testes anteriores (Tabela 14). Espera-se deste teste uma precisão alta, porém um treino altamente demorado.

Nº Teste	Nº Épocas	Nº *Batches*	*Optimizer*	Função de *Loss*	Função de Ativação	Métricas	*Dataset*
1	500	128	adam	binary_crossentropy	sigmoid	binary_accuracy	FER-2013

Tabela 14: Teste ao modelo VGGFace realizados na máquina B.

O único teste ao modelo VGG-Face envolveu 500 épocas, 128 *batch size*, *Optimizer* adam e métricas e precisão em modo binário. Durante a sua execução notámos que este modelo estava a demorar bastante a passar de época (duas horas e meia a

fazer a primeira época). Concretizando a sexta época optámos por parar o treino deste modelo pois já ultrapassava as quatorze horas. A sexta época obteve uma precisão de 72%, o que é bastante semelhante ao caminho que os restantes modelos obtiveram, após poucas épocas demonstram um grande crescimento de precisão. Assim acreditamos que os valores finais fossem bastante semelhantes aos restantes modelos.

3.6 DISCUSSÃO GERAL DE MODELOS ANALISADOS

Nesta secção serão discutidos aspetos relevantes sobre a análise de modelos estado da arte.

Durante a análise a estes modelos houve alguns aspetos chave que levaram à escolha do modelo a usar para a solução do problema de deteção de emoções de idosos. Nomeadamente o tempo de retreino dos modelos, onde o DeepFace foi o modelo que demorou menos tempo, com uma média de 60 segundos por época; todos os outros modelos são muito mais demorados, e se ordenarmos os modelos por ordem decrescente do seu tempo médio para completar uma época obtemos a seguinte ordem: VGGFace (8200 s), FaceNet (900 s), VGG16 (350 s), o OpenFace (300 s) e DeepFace (60 s).

Uma breve comparação do tempo de treino destes modelos pode ser expressada pela seguinte forma:

- **DeepFace**: 67 épocas, 1 hora e 20 minutos.
- **VGG16**: 34 épocas, 3 horas e 8 minutos.
- **VGGFace**: 7 épocas, 14 horas e 14 minutos (não foi terminado este teste).
- **FaceNet**: 56 épocas, 14 horas e 51 minutos.
- **OpenFace**: 61 épocas, 4 horas e 45 minutos.

Outro aspeto relevante observado durante os testes aos modelos, em exceção ao treino do modelo DeepFace: as máquinas ficaram muito lentas, a própria máquina deixava de responder, o que significa que o treino destes modelos exige muitos recursos computacionais. Conclui-se que os modelos (exceto o DeepFace) não são indicados para equipamentos *low-cost* onde não existe tanta capacidade de processamento.

Para concluir, e de modo a dar uma visão geral e comparativa entre os modelos, seguem-se algumas Tabelas relevantes e uma breve discussão de cada uma.

A Tabela 15 representa os resultados obtidos dos modelos treinados face à sua avaliação com o *dataset* CK+.

Modelo	Precisão Média	Precisão da classe 'Positive'	Precisão da classe 'Negative'
DeepFace	90%	79%	98%
VGG16	57%	0%	100%
FaceNet	30%	21%	37%
OpenFace	45%	9%	72%

Tabela 15: Métricas de comparação entre avaliação de modelos com o *dataset* CK+.

Nas Tabelas 16, 17, e 18 estão representadas os resultados e métricas dos melhores testes realizados a cada modelo.

Modelo	Nº Épocas	*Batches*	Maior Precisão no treino (%)	Menor perda
DeepFace	500	128	75	0.52
VGG16	500	128	73	0.6
FaceNet	500	128	73	0.55
OpenFace	500	128	72	0.55
VGGFace	500	128	Não concluído	Não concluído

Tabela 16: Métricas dos parâmetros dos melhores testes de cada modelo para o *dataset* FER-2013.

Modelo	Nº Épocas	Tempo total de retreino (segundos)	Média de temp por época	Ranking de tempo (segundos)	Épocas por Hora
DeepFace	67	4718	70	+0	51
VGG16	34	11252	331	+261	11
FaceNet	56	53432	954	+884	4
OpenFace	61	17105	280	+210	13
VGGFace	6	51223	8537	+8467	0

Tabela 17: Métricas dos resultados dos melhores testes de cada modelo.

Na Tabela 19 representa a comparação entre os *datasets* de avaliação de modelos.

Em suma, foi atingido o objetivo deste capítulo que consistiu em comparar os diversos modelos estado de arte de forma a averiguar qual o melhor modelo para usar na nossa solução. O modelo selecionado para constituir a solução foi o DeepFace,

Modelo	*Dataset* de Teste	*Overall Accuracy* (%)	*Positive* (%)	*Negative* %
DeepFace	CK+	90	79	98
DeepFace	PAINFUL Database *dataset*	86	-	86
VGG16	CK+	57	0	100
FaceNet	CK+	30	21	37
OpenFace	CK+	45	10	70
VGGFace	-	-	-	-

Tabela 18: Métricas dos resultados de cada modelo com *datasets* CK+ e PAINFUL Database.

Dataset	nº Imagens	nº classes	nº imagens *positive* - Treino	nº imagens *negative* - Treino	nº imagens *neutral* - Treino	nº imagens *positive* - Teste	nº imagens *negative* - Teste	nº imagens *neutral* - Teste
FER-2013	35886	3	7216	16529	4966	1775	4172	1234
CK+	180	2	56	66	-	21	34	-
PAINFUL Database	48398	1	-	48398	-	-	-	-

Tabela 19: Métricas dos *datasets* de testes.

tendo em conta a sua precisão alta tantos nos treinos como nos testes, mas também o facto de possuir o menor tempo de treino. Relativamente aos restantes modelos, algo dececionante a relatar foi o elevado tempo de retreino e a precisão baixa do teste ao *dataset* CK+ quando comparando com a precisão do retreino onde os modelos apenas apresentam $\sim 2\% de diferença para com a precisão do retreino DeepFace$.

4

DESIGN E ARQUITETURA DA SOLUÇÃO

Serve o presente capítulo para apresentar a arquitetura e *design* geral da solução proposta. Esta será inicialmente descrita em conjunto com as suas funcionalidades e requisitos com o apoio visual da Figura 26. Com base nos requisitos a solução proposta é dividida em sistema Elder Home e plataforma SmartEmotion, estes descritos a nível arquitetural e algorítmico com o auxílio de diagramas funcionais.

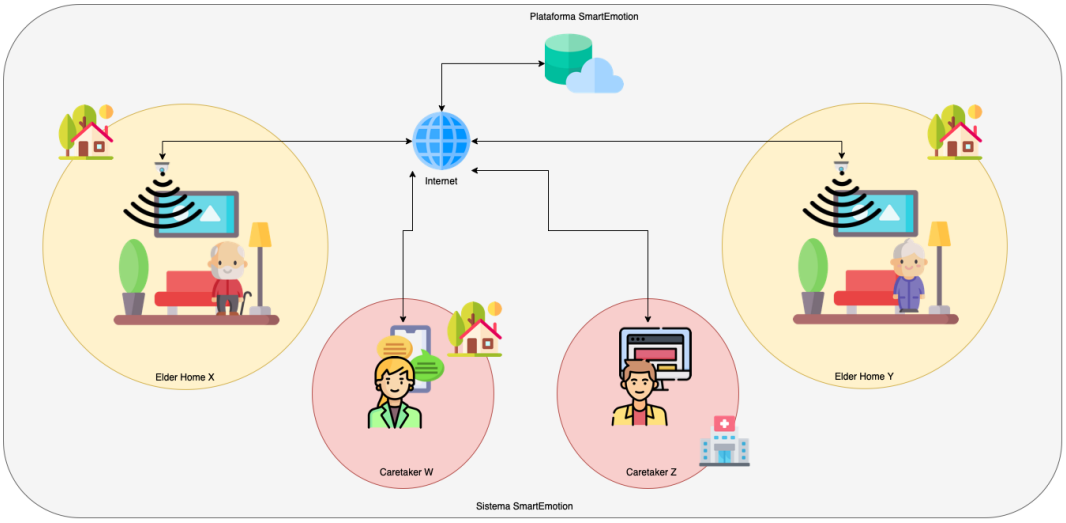


Figura 26: Protótipo de deteção das regiões de interesse e *landmarks*.

A solução para a deteção de emoções e reconhecimento facial em idosos num contexto de *Ambient Assisted Living* passa pela criação de um sistema que permita o acompanhamento do estado emocional das pessoas idosas e pela definição de emoções que despoletem alertas para os respetivos cuidadores (Figura 26). A partir das funcionalidades descritas acima são extraídos os requisitos funcionais do sistema:

- Aquisição de Imagens
- Reconhecimento Facial
- Deteção de Emoções
- Envio de Notificações e alertas
- Registo de *Logs*
- Treino do modelo

- *Human Labelling*
- Parametrização das Notificações
- Administração e Gestão do Serviço

Tendo em conta os requisitos mencionados anteriormente conclui-se que o sistema será constituído por 2 módulos: a plataforma SmartEmotion e o Sistema *Elder Home*. A plataforma SmartEmotion, alojada na Web, visa a interagir com o cuidador da pessoa idosa de forma a que este consiga acompanhar os estados emocionais da pessoa ao seu cuidado, personalizar o sistema e efetuar o *Human Labelling*, outra entidade poderá também usufruir da plataforma SmartEmotion, nomeadamente os administradores que visam a gerir o serviço e a realizar tarefas de administração. Já o Sistema *Elder Home* será integrado nas casas das pessoas idosas sendo que irá ser responsável pela aquisição e processamento das imagens, reconhecimento facial, deteção de emoções, envio de notificações e *logs* de sistema e também pelo retreino do modelo.

As tecnologias são sistemas de comunicação e informação que permitem a recolha, propagação e disponibilização de dados. Sendo que neste tópico irão ser abordadas quais as tecnologias usadas na construção de cada módulo do sistema SmartEmotion.

4.1 SISTEMA ELDER HOME

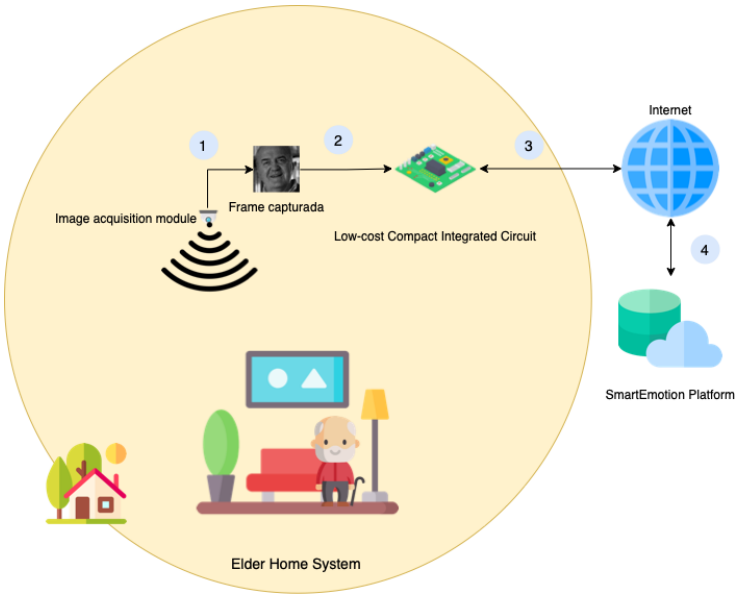


Figura 27: Arquitetura do Sistema Elder Home com hardware especificado.

Este sistema, esquematizado na Figura 27, é constituído pelos equipamentos instalados na residência da pessoa idosa, um dispositivo compacto de circuito integrado barato (doravante designado simplesmente de "microcontrolador"por questões de simplicidade) e uma câmara. A câmara tem como objetivo a captura de *frames*, sendo o microcontrolador responsável por três processos, previsão de emoções, catalogação de *frames* e, treino de modelos (Figura 28). Onde o primeiro processo destina-se ao recebimento, pré-processamento e análise dos *frames*, e a realização de pedidos à plataforma SmartEmotion, quer sejam para autenticação, envio de *frames*, *logs* de sistema e notificações, ou até recebimento de informações nomeadamente dados sobre a pessoa idosa e a lista de emoções que os seus cuidadores pretendem receber notificações. O segundo processo tem como objetivo receber e catalogar os *frames* classificados por humanos (Human Labelling) vindas da plataforma em pastas de modo a serem usadas no treino de um novo modelo no terceiro processo, este modelo irá ser usado para fazer uma melhor análise de *frames* no primeiro processo.

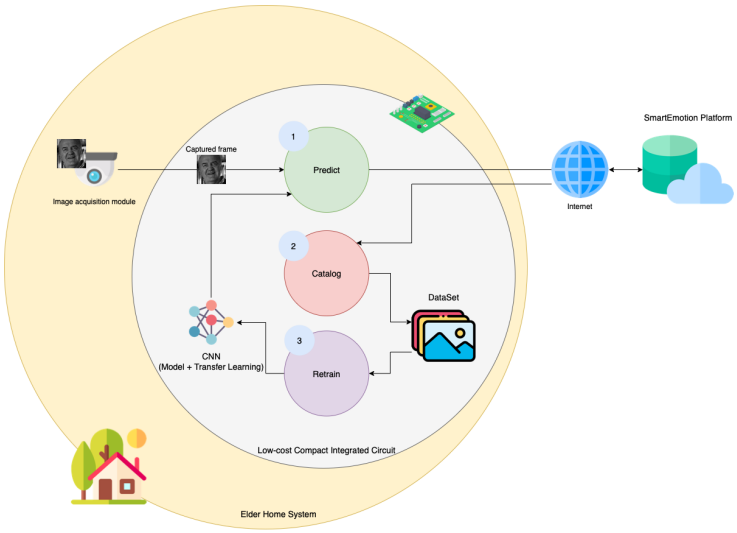


Figura 28: Arquitetura do Sistema Elder Home.

4.1.1 *Processo de Previsão de frames*

A previsão é um processo presente no microcontrolador (Figura 28 - este processo é responsável por fazer o pré-processamento e análise de *frames* capturados pela câmara de modo a enviar os *frames* para a plataforma SmartEmotion em forma de iteração ou de notificação.

O envio de *frames* não pode ser feito em tempo real pois iria causar uma quantidade grande de pedidos, ao invés será feito o envio de X em X tempo. A este envio de *frames* cíclico é denominado por iteração, e inclui os melhores *frames* de cada classe durante um dado intervalo de tempo e que são enviados após este intervalo temporal ter terminado. Esta analogia pode ser observada na Figura 29. Com isto, o número de iterações realizadas por ciclo é igual ao número de classes ao qual o modelo está treinado.

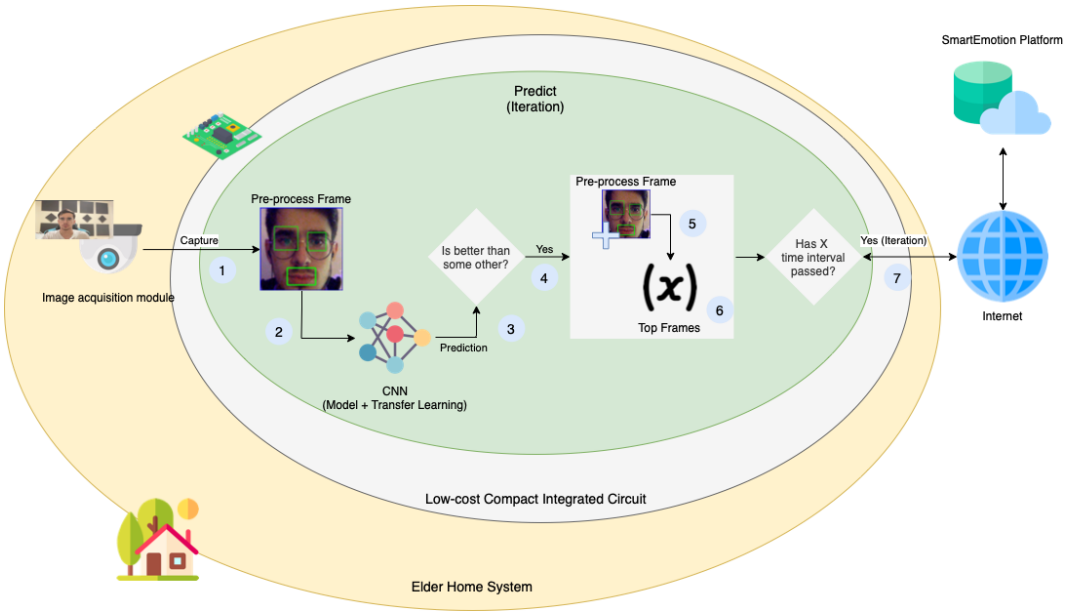


Figura 29: Algoritmo do processo de previsão resultante numa iteração de *frames*.

As notificações, geridas pelo microcontrolador (Figura 30), são relativas a situações onde o modelo detetou/previu sinais de uma emoção, presente na lista de emoções a notificar o cuidador, onde o valor detetado encontra-se acima de um determinado limite durante um período de tempo (ambas variáveis definidas pelo utilizador). Tendo em conta isto, quando uma notificação é despoletada a pessoa responsável pelo idoso precisa de receber uma notificação assíncrona na plataforma a que tem acesso de forma a ficar a par do ocorrido como seu familiar, desta forma o microcontrolador efetua o envio de notificações, não obstante este ainda envia as notificações para o servidor de forma a que sejam persistidas sendo que o servidor é responsável pelo envio de notificações para o cuidador por *email* e por mensagem.

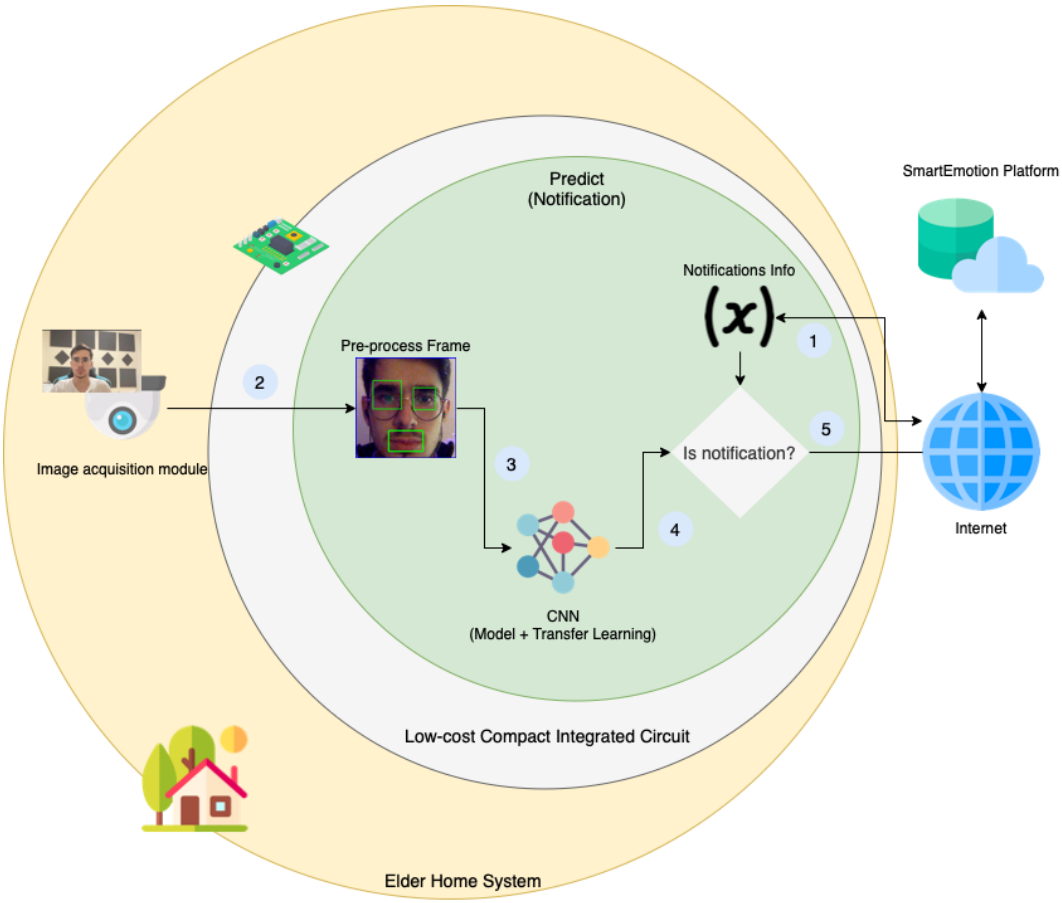


Figura 30: Algoritmo do processamento dedos até despoletar a notificação no microcontrolador.

4.1.2 *Processo de catalogação de frames*

O processo de catalogação de *frames* presente no microcontrolador (Figura 28) tem como objetivo receber os *frames* provenientes de *Human Labelling* da plataforma SmartEmotion e catalogá-los em pastas (Figura 31). Esta catalogação irá facilitar o processo de treino de modelos referido na próxima secção.

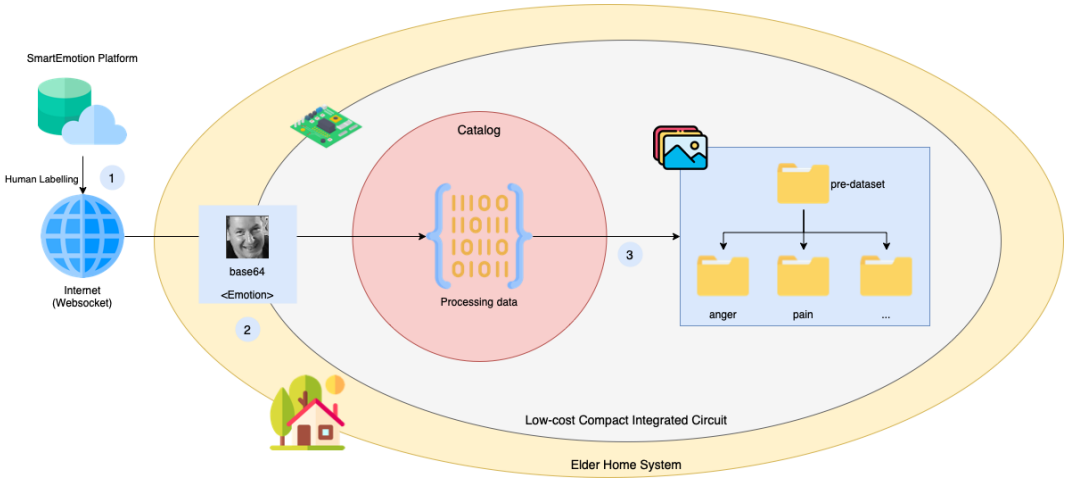


Figura 31: Algoritmo da catalogação de *frames* no microcontrolador.

4.1.3 *Processo de treino do modelo*

O treino do modelo é um dos processos contidos no microcontrolador (Figura 28); este processo é responsável por treinar o modelo com os novos dados vindos do processo de catalogação de *frames*, como demonstrado na Figura 32. O modelo resultante será posteriormente usado pelo processo de previsão para analisar os *frames* capturados pela câmara. De modo a satisfazer estes requisitos o modelo será treinado apenas com classes que contenham mais que Y *frames* e aplicando a abordagem de *Transfer Learning* visível na Figura 33.

4.2 PLATAFORMA SMARTEMOTION

Este módulo foi desenvolvido com o propósito de interação com o cuidador da pessoa idosa, principalmente tendo em conta todos os aspetos personalizáveis do Sistema Elder Home que precisam de informação fornecida pelo cuidador.

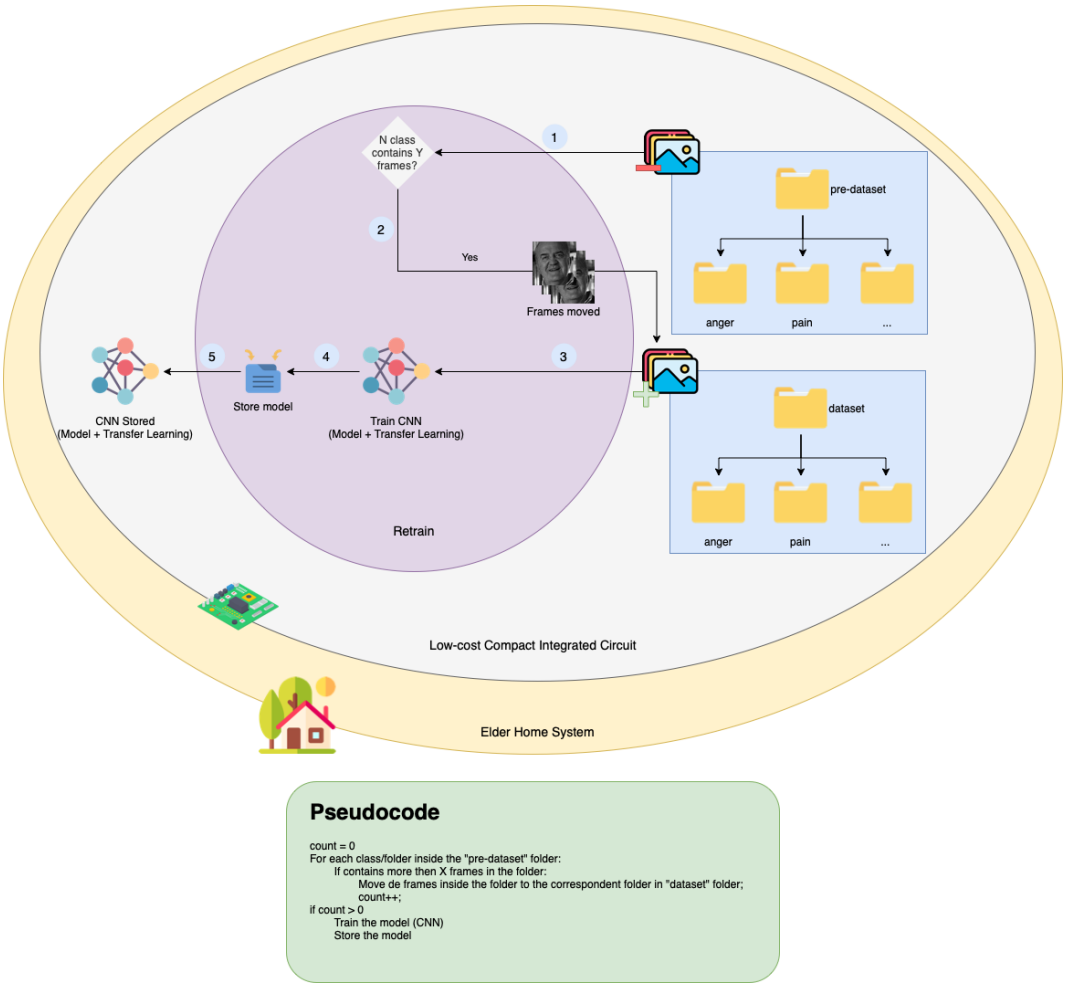


Figura 32: Algoritmo do processo de treino do modelo no microcontrolador.

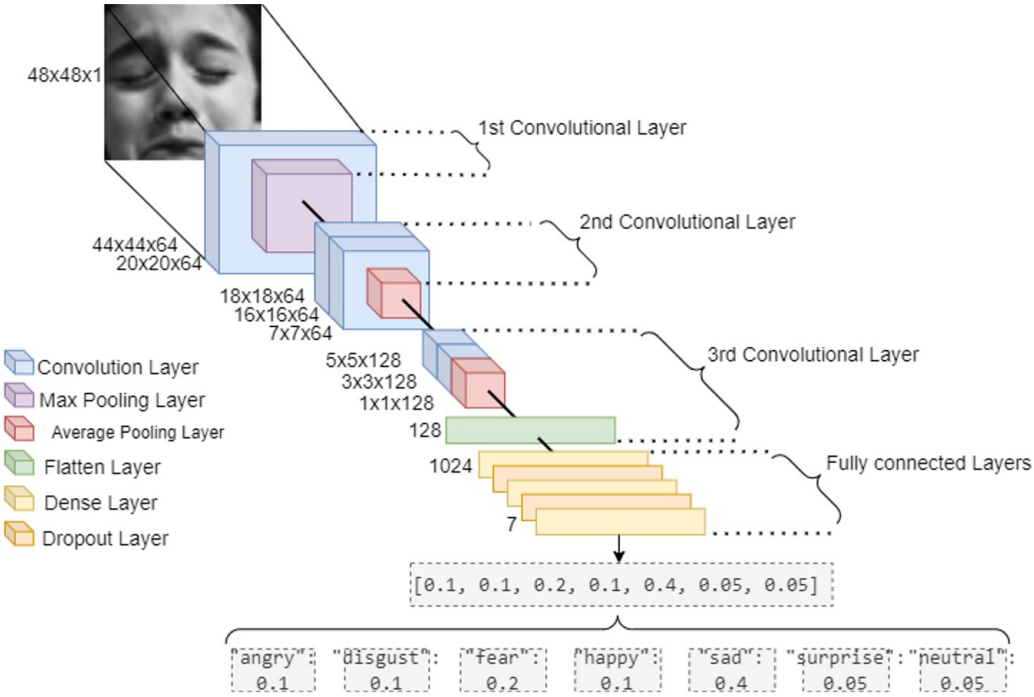


Figura 33: Arquitetura de camadas do treino do modelo no microcontrolador.

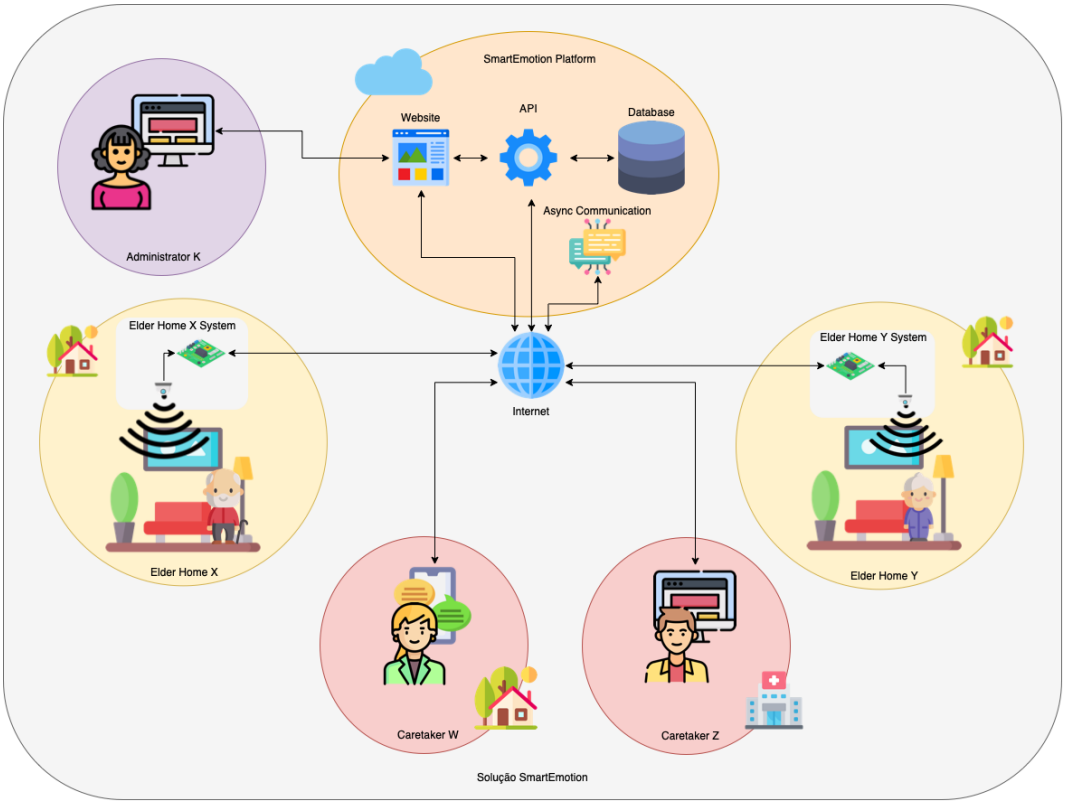


Figura 34: Arquitetura da Plataforma SmartEmotion.

Como é visível na figura Figura 34 a plataforma SmartEmotion é constituída por: um *website*; uma interface de programação de aplicações (API); uma base de dados; e um protocolo de comunicação assíncrona.

O *website* permite aos cuidadores e administradores da plataforma, por meio de uma interface gráfica, acederem à informação pretendida a qualquer hora e em qualquer lugar desde que haja acesso à *internet*. Este acesso é feito por interveniente da API que é responsável pelo processamento, persistência e divulgação dos dados recebidos da base de dados.

Tal como acontecia com o sistema Elder Home, aqui existe também a necessidade de um protocolo de comunicação assíncrono de forma a receber as notificações despoletadas pelo microcontrolador em tempo-real e também enviar um sinal quando o utilizador realizar *Human-Labelling*, tópico abordado ao longo da secção 2.6 do presente documento.

Em suma, através da análise da Figura 34 conclui-se que a API recebe e processa todos os pedidos provenientes do Sistema Elder Home e da plataforma Web. O protocolo de comunicação assíncrona como foi previamente mencionado é responsável pela partilha de notificações entre o sistema Elder Home e o Website.

5

SMARTEMOTION - UMA SOLUÇÃO PARA RECONHECIMENTO DE EMOÇÕES NUM CONTEXTO AMBIENT ASSISTED LIVING

O presente capítulo visa explicar o processo de implementação resultante da solução proposta no Capítulo 4. A solução SmartEmotion tem como principal contribuição permitir o acompanhamento emocional dos idosos por parte dos seus cuidadores em qualquer lugar e a qualquer momento. Deste modo, existem dois sistemas que são fundamentais para o funcionamento desta solução: Sistema Elder Home e Plataforma SmartEmotion, como está representado na Figura 35. Neste capítulo começar-se-á por apresentar a metodologia de desenvolvimento e a calendarização seguida no processo de desenvolvimento, em seguida detalhar-se-á a implementação dos dois sistemas, com o auxílio de diagramas de fluxo, figuras esquemáticas e demonstrativas.

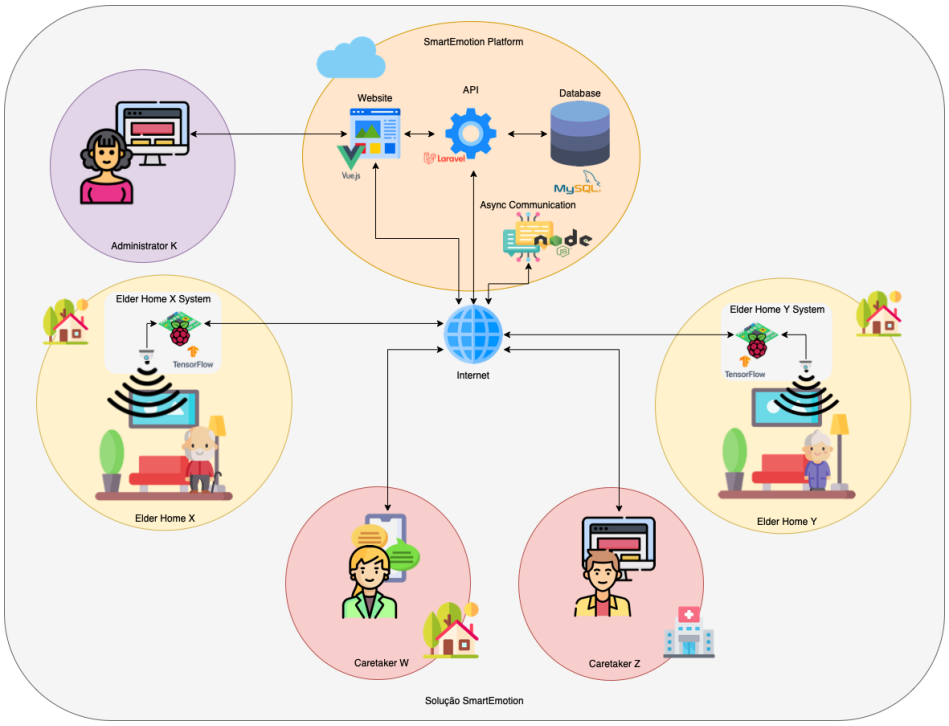


Figura 35: Proposta de implementação.

5.1 METODOLOGIAS E PROCESSO DE DESENVOLVIMENTO

A presente secção tem como objetivo mostrar todo o processo de desenvolvimento da solução. Este processo encontra-se representado através de um diagrama de GANTT (Figura 36). Será ainda descrito o processo de aprendizagem até à solução final.

Como mostra o diagrama de GANTT, Figura 36, a fase inicial foi dedicada à investigação, onde foram abordados conceitos base já referidos nos Capítulos 2 e 3, posteriormente iniciou-se o desenvolvimento da solução, processo descrito ao longo dos Capítulos 4 e 5. Por último, o diagrama aborda os testes em ambiente semi-controlado da solução no Capítulo 6.

O desenrolar deste processo deu origem a conceitos que acabaram por não ser abordados neste relatório, nomeadamente alguns protótipos iniciais que foram cruciais para chegarmos à solução do problema, particularmente:

- **Protótipo de deteção das regiões de interesse e *landmarks*** - Figura 37 - Este protótipo usava modelos bastante pesados para fazer a deteção de regiões de interesse.
- **Protótipo de deteção de emoções usando *Machine Learning*** - Figura 38 - O pré-processamento de imagens deste protótipo consistia na extração de distâncias entre pontos, este calculava a distância entre pontos na região da boca permitindo a classificação da emoção de felicidade. Este protótipo seria adequado se não tivéssemos acesso a modelos já treinados como o DeepFace para reaproveitar.

Assim, estes protótipos estão integrados no nosso processo de aprendizagem e através do seu desenvolvimento tivemos a oportunidade de conhecer mais sobre outras abordagens que poderão ser usadas futuramente neste projeto ou noutro da mesma área.

5.2 SISTEMA ELDER HOME

Previamente ao desenvolvimento do sistema Elder Home tiveram de ser analisados os diversos equipamentos de *hardware* que irão corresponder ao microcontrolador

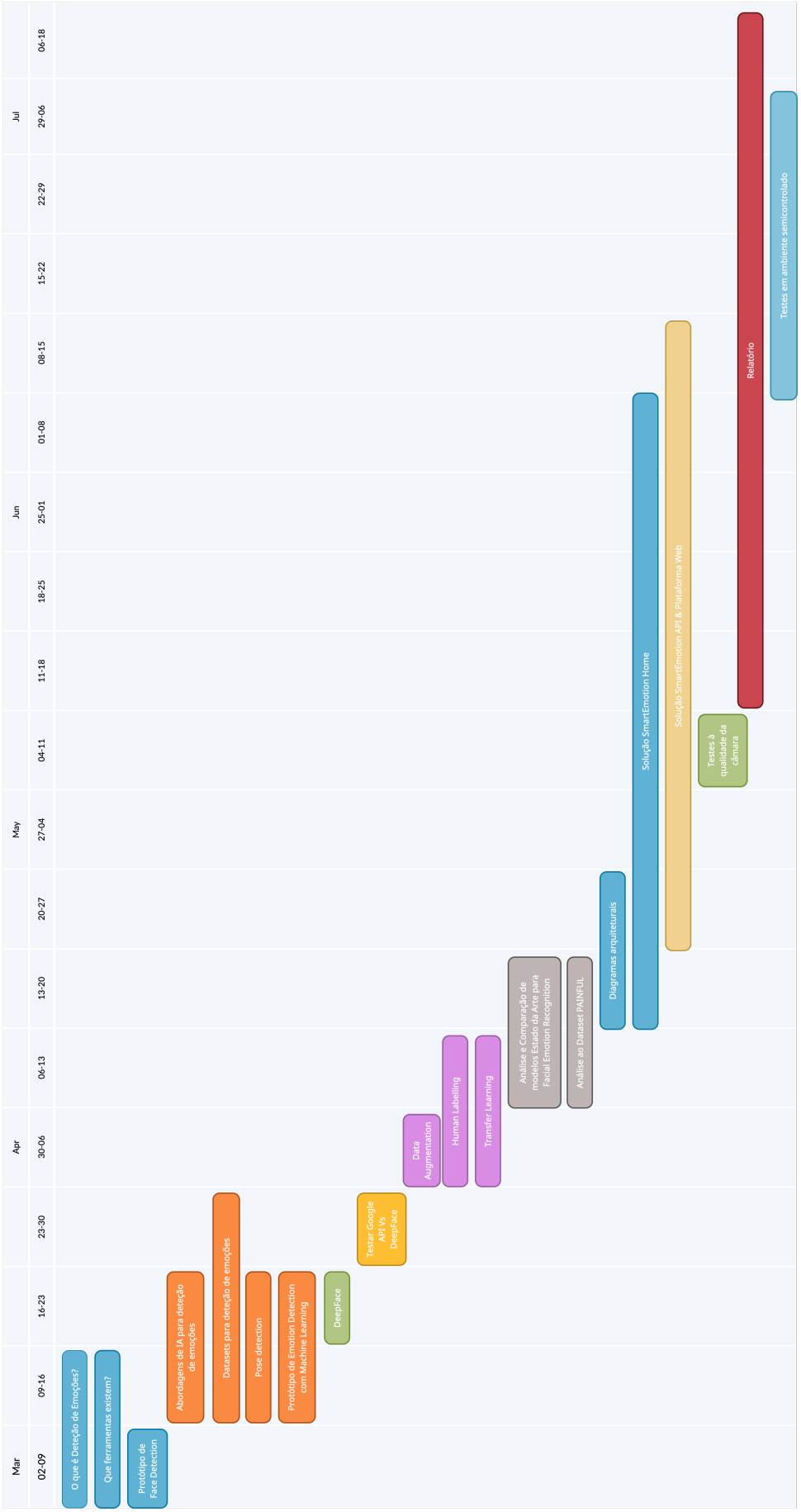


Figura 36: Diagrama de GANTT representativo do processo de desenvolvimento e investigação da solução.

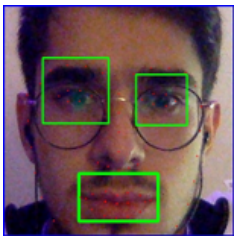


Figura 37: Protótipo de deteção das regiões de interesse e *landmarks*.

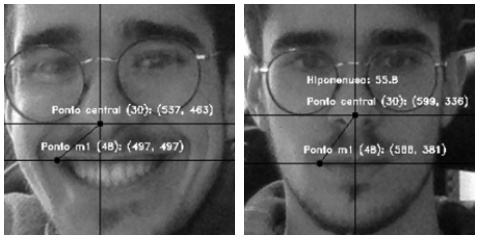


Figura 38: Protótipo de deteção de emoções usando *Machine Learning*.

visível na Figura 27 com vista a decidir qual o melhor equipamento usar. A câmara usada para este sistema foi uma Xiaomi Imilab Full HD. [1]

Na escolha do hardware/equipamento a utilizar tiveram de ser considerados os seguintes aspetos:

- **Preço do equipamento** - A criação de um sistema *low-cost* era um dos objetivos do sistema.
- **Capacidade de processamento** - Este é um dos pontos cruciais sendo que o equipamento precisa de ter capacidade para usar modelos de arquitetura *Deep Learning*
- **Sistema operativo** - Existe necessidade do equipamento conter um sistema operativo para fazer a instalação do dependências e recursos necessários.
- **Capacidade de armazenamento** - Devido à necessidade de guardar imagens no equipamento é obrigatório que o equipamento tenha pelo menos 16GB de memória.
- **Discreto** - O equipamento precisa de ser discreto, não é pretendido um equipamento que seja vistoso e que perturbe a normalidade do dia-a-dia da pessoa idosa.
- **Câmara de vídeo** - Devido à necessidade de capturar o rosto da pessoa idosa em tempo real é mandatório que o equipamento suporte câmara de vídeo.

1 Consulte o link para informações relativas à câmara Xiaomi Imilab Full HD https://www.kontrolsat.com/pt/gaming/perifericos/webcams/webcam-imilab-full-hd?utm_source=kuantokusta&utm_medium=cpc&utm_campaign=catalogo

Existem duas marcas bem conhecidas de microcontroladores que podem ser consideradas possíveis candidatas a ser utilizadas, Arduino e Raspberry Pi (Figura 39). Ambas têm as suas vantagens: o Arduino é mais discreto e está disponível por preços baixos, mas não tem sistema operativo e também não tem capacidade de processamento nem armazenamento; o Raspberry Pi tem um preço mais elevado, porém já é um microcontrolador que satisfaz todas as necessidades. Com base nestes argumentos, o equipamento escolhido foi o Raspberry Pi CM4 com Board IO (Figura 40), com o sistema operativo *Raspberry Pi OS Lite (64-bit)*. [36][44][55]

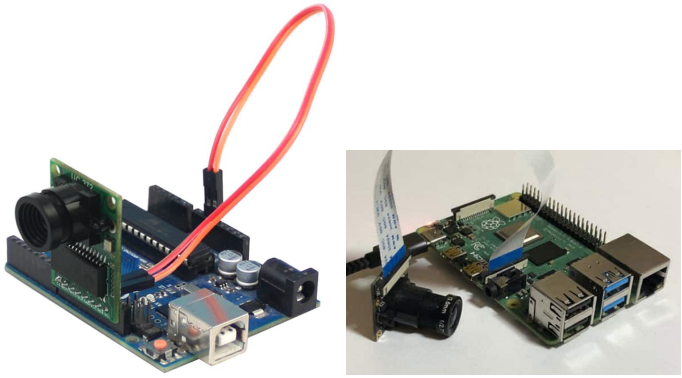


Figura 39: Microcontroladores Arduino e Raspberry Pi.

Figura 40: Arquitetura do Módulo do sistema IoT.

O sistema Elder Home (Figura 28), como referenciado no capítulo 4, permite efetuar a captura e processamento de imagens que posteriormente serão passadas pelo modelo que foi construído através da aplicação da técnica *Transfer Learning* (Secção 2.2.3) ao modelo das emoções do DeepFace (Secção 3.1), e no caso de existirem as condições adequadas será despoletada uma notificação que será enviada para a plataforma SmartEmotion. Em qualquer caso de erro será enviado um *log* que conterá os detalhes relativos ao problema, e no término de uma iteração será enviado um *log* contendo a temperatura do equipamento. Quando o número de *frames* adquiridos com Human Labelling para qualquer das emoções personalizadas for superior ao limite definido pelo utilizador o retreino do modelo acontecerá.

O sistema Elder Home é constituído por três processos:

- ***AAL.py*** - processo responsável pela captura de imagens, deteção de emoções, envio de notificações e envio de *logs*
- ***Retrain.py*** - processo encarregado da verificar periodicamente se o retreino poderá ocorrer e realização do mesmo
- ***Websocket.py*** - processo responsável por receber e catalogar os *frames* classificados através de *Human Labelling* provenientes da plataforma SmartEmotion

Um dos pontos diferenciadores da nossa solução consiste na personalização de todo o sistema. Relativamente ao sistema Elder Home o utilizador poderá modificar os parâmetros consoante as suas preferências através da alteração de valores presentes no ficheiro .env (Código 2). Apenas alguns dos parâmetros deste ficheiro podem ser alterados pelo utilizador de forma a manter o correto funcionamento do sistema Elder Home, nomeadamente:

- ***CLIENT_EMAIL*** - Email da conta do utilizador na plataforma SmartEmotion
- ***CLIENT_PASSWORD*** - Password da conta do utilizador na plataforma SmartEmotion, de momento está legível no entanto para implementações futuras será encriptada
- ***RETRAIN_TIME*** - Intervalo de tempo, em segundos, para verificar se existem condições para realizar o retreino de modelo
- ***RETRAIN_NUMBER_FILES*** - Número de imagens provenientes de *Human Labelling* que a pasta de qualquer das emoções personalizáveis terá de conter para que o retreino do modelo aconteça
- ***NOTIFICATION_INTERVAL*** - Intervalo de tempo, em segundos, entre o envio de notificações

Relativamente aos parâmetros restantes:

- ***API_URL*** - URL da API que integra a plataforma SmartEmotion
- ***WEBSOCKET_URL*** - URL do WebSocket que integra a plataforma SmartEmotion
- ***TOP_FRAMES_PATH*** - Nome da pasta onde os 10 melhores *frames* serão armazenados
- ***PRE_DATASET_PATH*** - Nome da pasta onde os *frames* resultantes do *Human Labelling* irão ser catalogados por sub-pastas com os nomes das emoções personalizáveis

- ***DATASET_PATH*** - Nome da pasta onde se encontra o *dataset* utilizado para os treinos e retreinos

```
API_URL = http://aalemotion.dei.estg.ipleiria.pt/api
WEBSOCKET_URL = http://aalemotion.dei.estg.ipleiria.pt/
TOP_FRAMES_PATH = top10Frames
PRE_DATASET_PATH = pre_dataset
DATASET_PATH = FER-2013
CLIENT_EMAIL=2191170@my.ipleiria.pt
CLIENT_PASSWORD=1234567
RETRAIN_TIME = 86400
RETRAIN_NUMBER_FILES = 10
NOTIFICATION_INTERVAL = 10

```

Código 2: Ficheiro .env do Sistema Elder Home.

A Figura 41 descreve a estrutura de dois modelos; à esquerda encontra-se o modelo inicial de emoções do DeepFace, sendo que à direita está o modelo do sistema atual, que implicou a utilização de *Transfer Learning* no modelo do DeepFace onde foram congeladas (escondidas) as camadas *fully connected* e foram adicionadas duas novas camadas onde se deixou de ter sete classes para apenas três, sendo este o valor inicial que aumentará com base no crescimento do *dataset* causado pela criação de emoções personalizáveis através de *Human Labelling*. Este modelo adaptado é utilizado nos processos AAL.py e Retrain.py e o uso deste modelo torna-se possível com a biblioteca TensorFlow [60].

5.2.1 *Processo AAL.py*

O modelo utilizado na fase inicial do sistema Elder Home corresponde ao melhor modelo obtido durante os testes realizados aos modelos estado da arte previamente abordados neste documento (Capítulo 3). No entanto o utilizador poderá forçar a criação de um novo modelo indicando na chamada ao processo AAL.py a *flag -f* ou *--force* e os valores dos parâmetros:

- ***-a*** - Função de ativação
- ***-l*** - Função de *loss*
- ***-m*** - Métricas
- ***-e*** - Nº de Épocas
- ***-b*** - Nº de *batches*

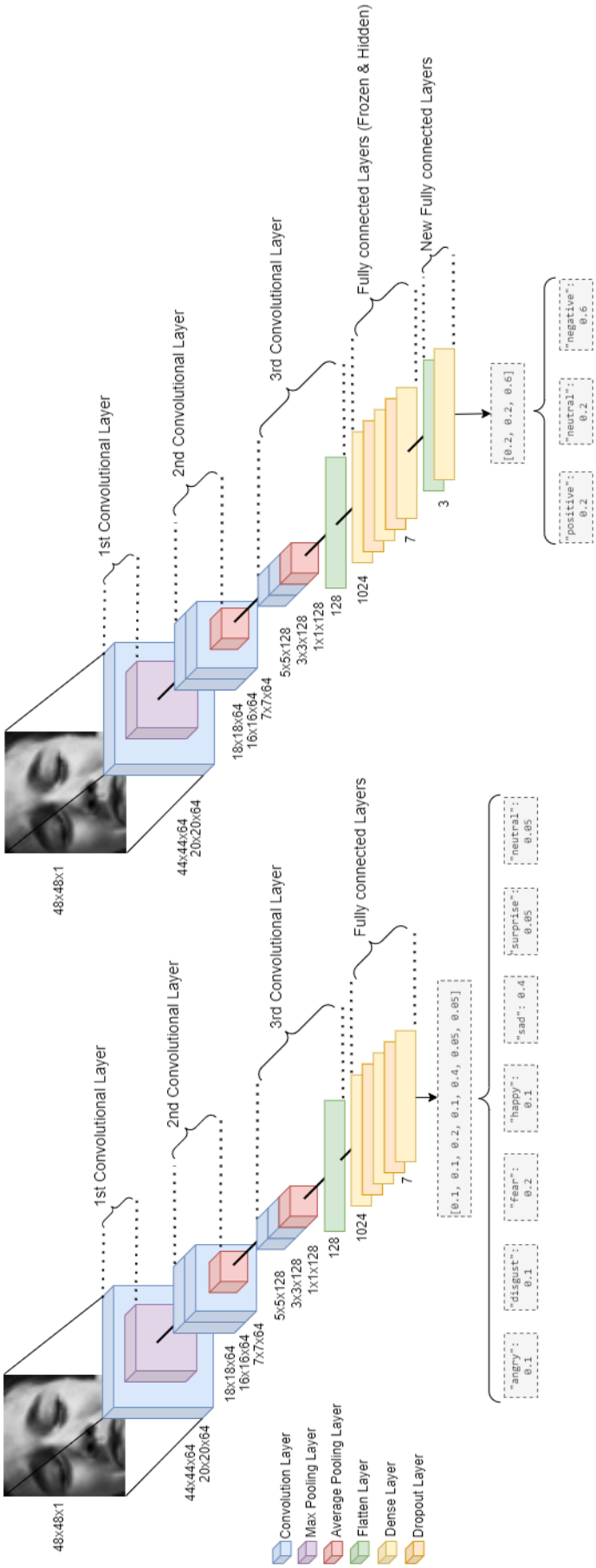


Figura 41: Modelo do DeepFace conjuntamente com *Transfer Learning*.

Na eventualidade de a *flag -f* ou *--force* não estar presente na chamada ao processo, serão utilizados os valores *default* do modelo (Código 3) mesmo que nenhum novo modelo seja criado - devido ao facto de estes parâmetros precisarem de estar sempre preenchidos.

```
        {
            default_model = {
            'model': "DeepFace",
            'epochs': 500,
            'batches': 128,
            'activation': "sigmoid",
            'loss': "binary_crossentropy",
            'metrics': "binary_accuracy"
        }

```

Código 3: Parâmetros *default* para o modelo.

De entre todos os parâmetros abordados acima, que poderão ser configurados na linha de comandos, destaca-se ainda um que não consta da criação de um novo modelo, nomeadamente o tempo entre cada iteração, definido com a *flag -t* ou *--time*, onde o valor precisa de ser indicado em segundos. A utilização da *flag -h* ou *--help* apresenta um guia para todos os parâmetros abordados acima. Após a recolha dos parâmetros é realizado um pedido de autenticação com os dados da conta do cliente, que estão contidos no ficheiro .env, e caso o pedido seja bem sucedido o *access_token* é guardado no ficheiro token.txt e inicia-se o ciclo da iteração onde ocorre a captura e processamento de *frames*, envio de *logs* e notificações. A autenticação é verificada e refeita ainda diversas vezes ao longo do processo.

O sistema Elder Home efetua de forma contínua a captura de *frames* sendo que cada *frame* é passado para o modelo. Posteriormente, o modelo retorna um objeto JSON (Código 4), onde *emotions* consiste num outro objeto JSON que contém todas as emoções que o modelo deteta. A cada emoção está associada o valor da confiança de estar contida no *frame*. O valor de *dominant_emotion* indica qual das emoções detetadas apresenta o maior valor de confiança (ou seja, a emoção que o modelo considera ser a mais predominante num determinado *frame*). Por último, *region* consiste num outro objeto JSON que indica as coordenadas x e y e o comprimento w e altura h onde a região de interesse do rosto se encontra (retângulo à volta da face).

As coordenadas x e y indicam as coordenadas 2D no canto superior esquerdo onde a região de interesse começa, sendo que o valor de w indica o comprimento do retângulo da região de interesse e o valor de h corresponde à altura do retângulo

da região de interesse. De seguida, é desenhada no *frame* a região de interesse que depois será usada para o corte do *frame* (de forma a que apenas seja mantida a região de interesse da imagem) que posteriormente sofre uma alteração na escala de cores para tons de cinza. A região de interesse, retângulo que constitui a face presente na imagem, constitui a única parte da imagem que é persistida no sistema, como demonstra a Código 4.

```
        {
            "emotion":{
            "negative":67.44962923611466,
            "neutral":17.521483402319756,
            "positive":15.02888736156559
            },
            "dominant emotion": "negative"
            "region":{
                "g":270,
                "y":227
                "w":154,
                "h":154
                1
            }
        }

```

Código 4: Objeto JSON resultante da previsão do modelo.

Figura 42: Região de interesse.

Após a extração da região de interesse, inicia-se o processo de averiguação dos dez melhores *frames* de cada emoção existente durante o intervalo de tempo definido no parâmetro *-t* ou *--time*, nomeadamente os dez *frames* que apresentam os maiores valores de confiança. O algoritmo é realizado para cada emoção dominante de cada *frame* e também sempre para a emoção de neutralidade (devido à escassez de imagens de neutralidade no *dataset* de treino, sendo que desta forma estamos a

forçar o aumento de nº de *frames* neutros). Estes dez *frames* são mantidos até ao fim da iteração numa pasta de nome *top10Frames* (definida no ficheiro .env). O algoritmo, que é executado para todas as emoções detetadas pelo sistema, é utilizado para garantir que apenas são guardados os dez *frames* com maior confiança, o que implica a atualização contínua do top dez de *frames*. O algoritmo desenvolvido para resolver este problema encontra-se demonstrado na Figura 43.

O algoritmo, no caso de existirem dois *frames* capturados no mesmo segundo, apenas guarda o *frame* que apresente um maior valor de confiança - que pode ser observado na estrutura de decisão marcados nos pontos 2 e 3 da Figura 43.

O algoritmo começa por analisar o nº de *frames* que pertencem ao conjunto dos 10 melhores *frames*. No caso de chegar um novo *frame* e o conjunto ainda não tiver 10 *frames*, apenas se adiciona esse *frame* novo ao conjunto. Caso contrário, torna-se necessário gerir todo o conjunto dos 10 melhores *frames* de maneira a que apenas sejam mantidos os 10 melhores *frames*.

Quando termina o tempo do ciclo da iteração são enviados, para a plataforma SmartEmotion, os 10 melhores *frames* de cada emoção. Desta forma, apenas são criadas iterações para as emoções que tenham *frames* associados. O envio de *frames* e criação de iterações estão destacados na secção 5.3. No caso de ocorrer um erro no envio da iteração o pedido será refeito no máximo três vezes através de um mecanismo de tentativas.

Tal como já foi mencionado, o processo AAL.py é responsável pelo envio de notificações, esquematizado na Figura 44, sendo que o intervalo de tempo para a ocorrência das notificações (parâmetro z na Figura 44) é definido pelo utilizador no ficheiro .env; sendo o valor atual de 10 segundos, o que implica que caso estejam reunidas condições para ocorrerem diversas notificações estas ocorrerão de 10 em 10 segundos – de forma a evitar um sentimento de saturação no cuidador. O controlo do envio das notificações ocorre de seguida ao processamento dos *frames*, ou seja, depois do algoritmo abordado acima.

O utilizador através da plataforma SmartEmotion poderá configurar quais as emoções para as quais pretende receber notificações, indicando o limite da confiança da previsão e também a duração em segundos, sendo este pedido realizado diversas vezes ao longo do processo AAL.py, nomeadamente depois de ser efetuada a autenticação do utilizador e também depois do envio das iterações de todas as emoções, ou seja, antes de ser iniciado outro ciclo de iterações. Desta forma, e tal como se observa nos pontos de decisão 1 e 2 representados na Figura 44, o ponto

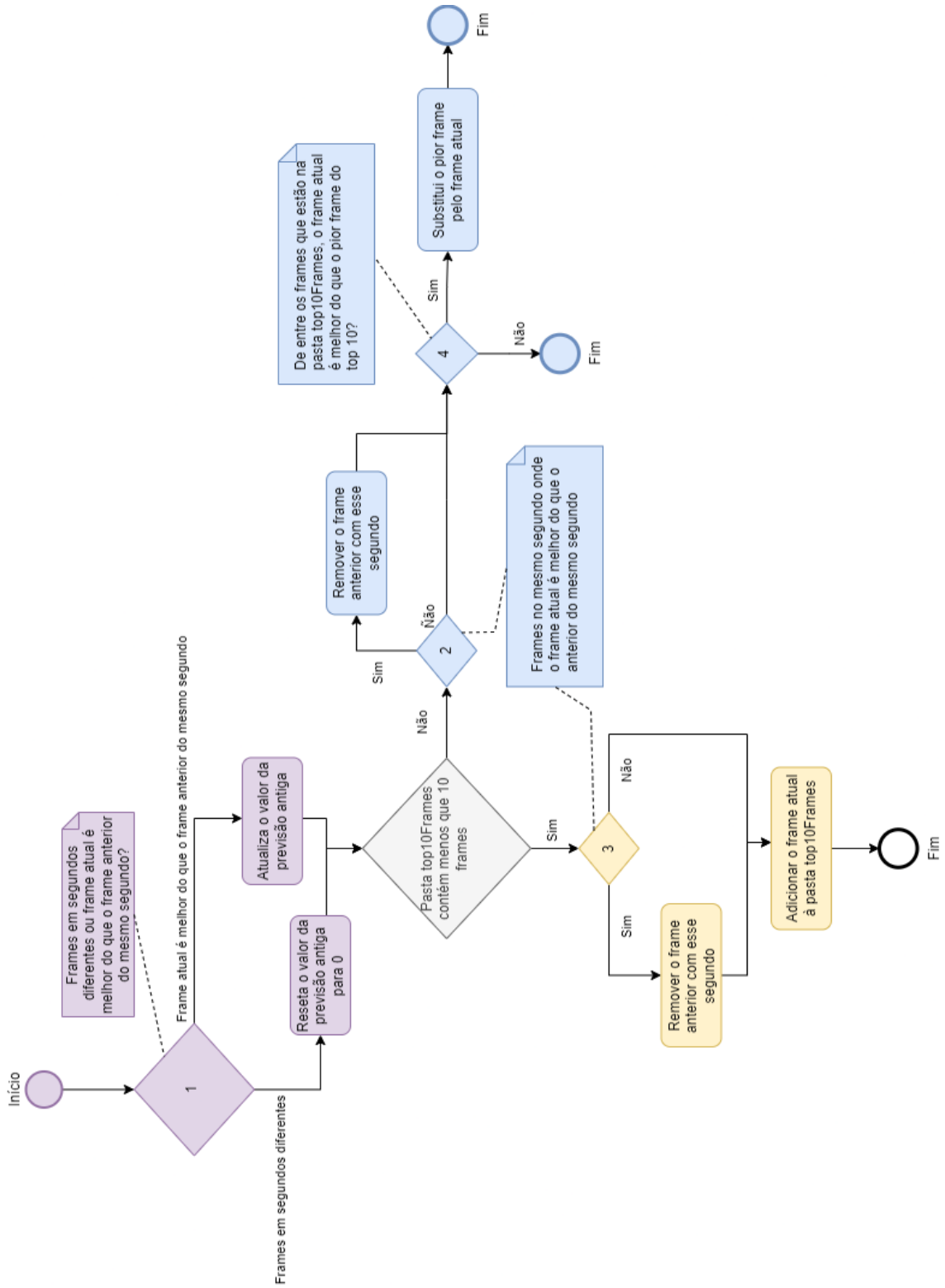


Figura 43: Algoritmo para o *top10Frames*.

crucial para o envio das notificações implica verificar se o limite de confiança da emoção dominante foi atingida durante um intervalo de tempo.

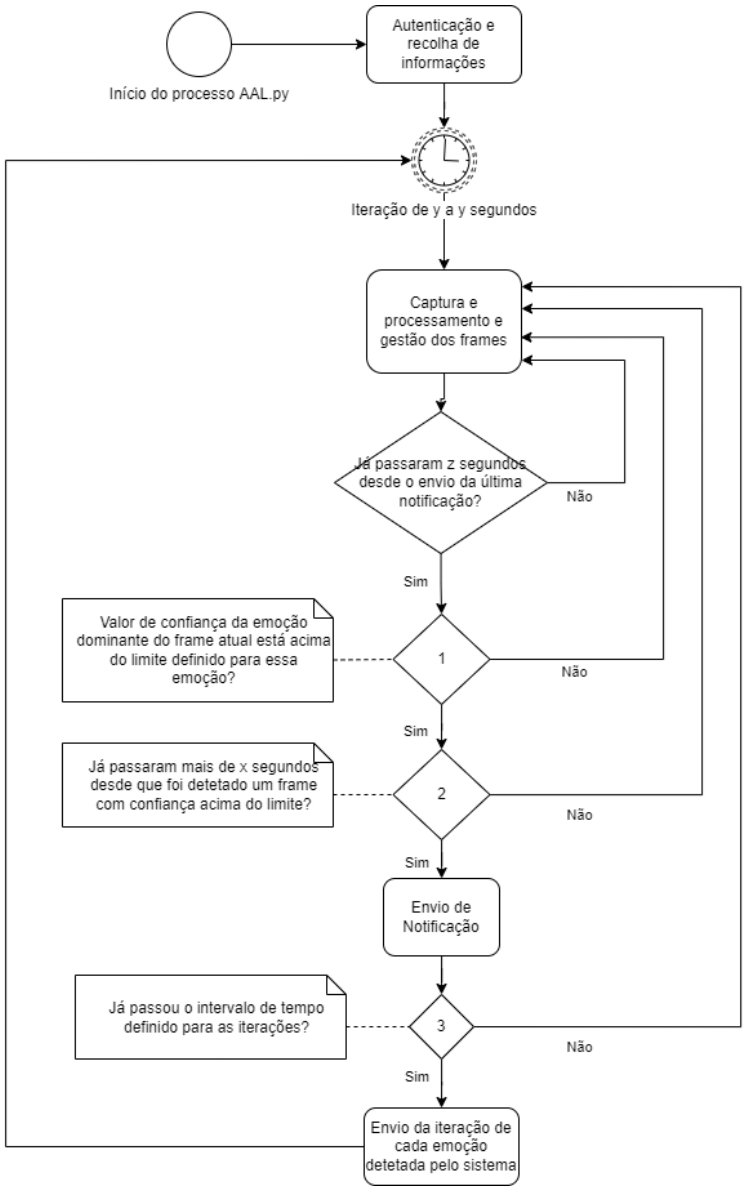


Figura 44: Esquema do envio das notificações.

De maneira a recolher todos e quaisquer dados relativos a erros de sistema e até a recolha de temperatura do Raspberry Pi 4 são enviados *logs* para a plataforma SmartEmotion *logs* de todas as situações mencionadas, sendo que os dados dos *logs* incluem o endereço MAC do equipamento, a descrição, a data de ocorrência e o nome do processo (tendo em conta que todos os processos do sistema Elder Home poderão despoletar logs). Algumas das situações que poderão levar ao envio de *logs* para a plataforma são a falhas nos pedidos das iterações, notificações e envio de *frames*, erros que envolvam o processamento e gestão dos *frames*, e por último

diversos *logs* são ainda enviados contendo a recolha de temperatura do equipamento. Finalmente, o processo AAL.py como mecanismo de segurança caso a plataforma esteja indisponível permite ainda registar no ficheiro local.logs.txt a informação de todos os erros e exceções que ocorram e também os valores de temperatura do Raspberry Pi.

5.2.2 *Processo Websocket.py*

Este processo visa a catalogar as imagens provenientes da plataforma SmartEmotion após o utilizador classificar os *frames* da iteração com a respetiva emoção personalizada - *Human Labelling*. Tendo em conta isto, este processo inicialmente conecta-se aos Websockets da plataforma SmartEmotion de maneira a receber comunicações de todos os *frames* classificados pelo utilizador. Sendo que, com a chegada de um novo *frame*, este processo irá guardar o respetivo *frame* na pasta da emoção personalizada que se encontra dentro da pasta *pre_dataset*.

5.2.3 *Processo Retrain.py*

O processo Retrain.py é responsável verificar periodicamente se o retreino poderá ocorrer e realizá-lo quando estão reunidas as devidas condições usando a biblioteca de Python TensorFlow [60], sendo que o funcionamento deste processo está explicado na Figura 45. Este processo inicia-se com a recolha dos parâmetros usados para treinar o modelo, tópico abordado no início desta secção.

Após a verificação da autenticação, realizada através da realização de um pedido à plataforma indicando o *token* contido no ficheiro token.txt, inicia-se um ciclo onde de x a x segundos (valor personalizável indicado no parâmetro *RETRAIN_TIME* do ficheiro .env, e.g 86400 segundos = 1 hora), se verifica se já passaram x segundos desde a última vez que foi verificado se o retreino poderia ou não ocorrer. Durante os x segundos desde a última verificação o processo fica à espera que esse tempo passe, e quando passarem mais de x segundos o processo verifica se existem *frames* catalogados, ou seja, se a pasta que recebe os *frames* classificados por *Humman Labelling* contém algum *frame* (estrutura de decisão 1 da Figura 45).

No caso de não existirem *frames* catalogados o retreino não acontece e o contador de tempo é atualizado, de seguida o processo retorna ao início do ciclo. Se existirem *frames* catalogados o processo percorre todas as sub-pastas (relativas às emoções

personalizadas) da pasta *pre_dataset* e averigua para cada uma dessas emoções se foi superado o limite de *frames* necessários para um treino ocorrer (parâmetro *RETRAIN_NUMBER_FILES* do ficheiro .env); esta avaliação corresponde ao ponto de decisão 2 da Figura 45. Se este facto se verificar é criada uma cópia do modelo atual e de seguida ocorre o retreino do novo modelo que utilizará a versão mais atualizada do *dataset*. Se para nenhuma das emoções personalizadas o limite foi ultrapassado o retreino não ocorre e apenas é atualizado o contador do tempo; o processo retorna para o início do ciclo, onde ficará à espera que passem mais x segundos.

5.2.4 *Instalação automática do sistema Elder Home e mecanismo de reboot automático*

A configuração do sistema Elder Home implica a instalação de diversas dependências, sendo que para um ambiente de utilização real é conveniente que a instalação seja rápida e que não ocorram erros. Tendo em conta este facto, a solução apresentada contém ainda um ficheiro startup.sh que realiza a instalação de todas as dependências necessárias ao funcionamento do sistema.

Num ambiente real a solução concebida seria configurada uma única vez. Sendo que, o comportamento ideal seria que, posteriormente a uma falha de eletricidade onde o Raspberry Pi se reiniciasse sozinho, os três processos descritos acima: (AAL.py, Retrain.py e Websocket.py) fossem inicializados no *reboot* do microcontrolador. Desta forma o sistema Elder Home poderia ser reiniciado sem intervenção humana, o que levou à criação do ficheiro AAL.bat responsável pela inicialização dos três processos. O mecanismo de alterar a sequência de ficheiros a serem chamados no processo de *reboot* podem ser implementados de diversas maneiras. A abordagem escolhida envolveu a modificação do conteúdo do ficheiro rc.local de acordo com as instruções descritas em [27], onde apenas foi necessário chamar o ficheiro AAL.bat referido anteriormente.

No entanto, esta abordagem não foi implementada com sucesso, sendo que após diversas tentativas de *reboot* os processos do Sistema Elder Home nunca foram inicializados de forma automática como era desejado. Não obstante, a reflexão e investigação de uma possível solução deste problema constitui uma mais-valia para a solução.

Ao longo do desenvolvimento da solução o microcontrolador Raspberry Pi 4 apresentou temperaturas elevadas e perigosas que podem causar danos no mesmo.

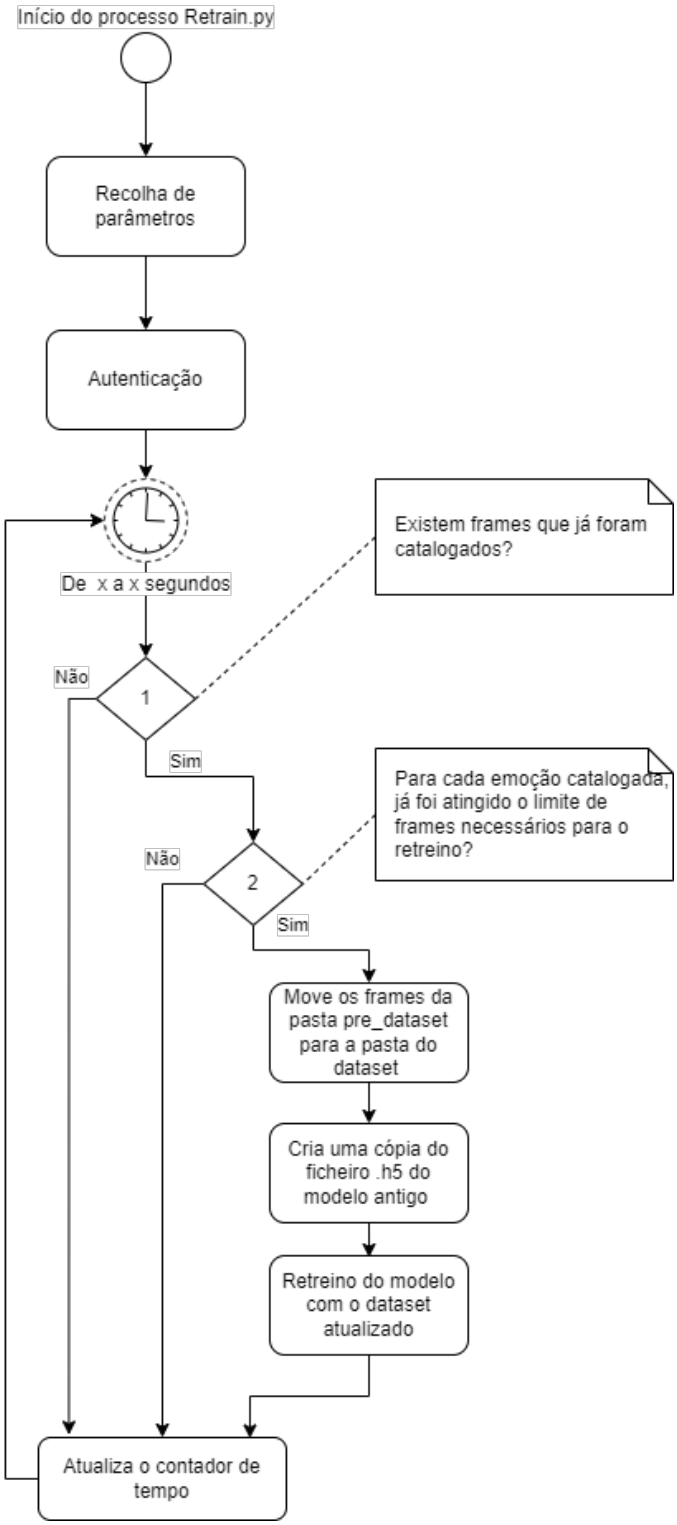


Figura 45: Esquema do processo Retrain.py.

Este facto foi obtido através do envio de múltiplos *logs* contendo a temperatura do equipamento ao longo do tempo, tal como foi previamente abordado. Sendo que durante muitos dias a temperatura apresentou valores entre 70ºC e 85ºC (e.g., Figura 46). Como medida de resolução para o problema, foram incorporadas ventoinhas para os equipamentos, que se observam na Figura 47. O correto funcionamento da ventoinha precisa da configuração de drivers e da intensidade da mesma, sendo que o ficheiro SetupToControlFan.rtf apresenta o tutorial necessário para o bom funcionamento da ventoinha.

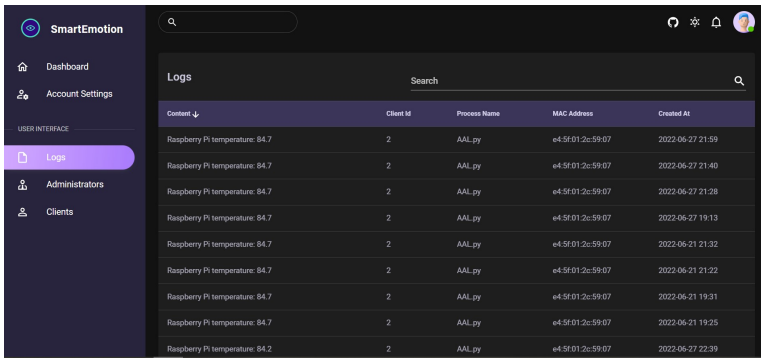


Figura 46: Raspberry Pi 4 integrado com ventoinha.

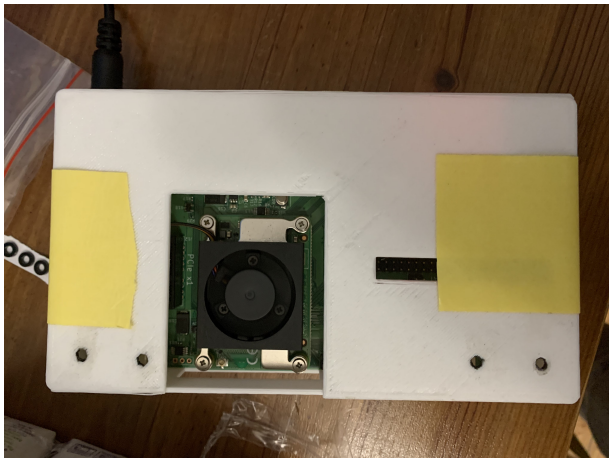


Figura 47: Raspberry Pi 4 integrado com ventoinha.

5.3 PLATAFORMA SMARTEMOTION

A plataforma SmartEmotion permite ao cuidador acompanhar o estado emocional do idoso e ainda receber notificações quando alguma alteração emocional acontecer. A plataforma permite ainda aos administradores do sistema gerirem a plataforma.

Esta secção irá permitir visualizar como foi implementada a solução apresentada no capítulo 4.

A plataforma de maneira a poder acompanhar o estado emocional da pessoa precisa de receber a informação enviada pelo microcontrolador. Esta informação vem em formato de autenticação, iteração (Secção 4.1.1), notificação e *logs*; a comunicação é feita por uma API desenvolvida em Laravel ou por um protocolo de comunicação assíncrono, Websockets, desenvolvido em Node.js, e é posteriormente persistida numa base de dados MySQL (Figura 48). De modo a estar disponível em qualquer dispositivo para o cuidador foi ainda desenvolvido uma plataforma cliente desenvolvida em Nuxt.js + Vuetify.

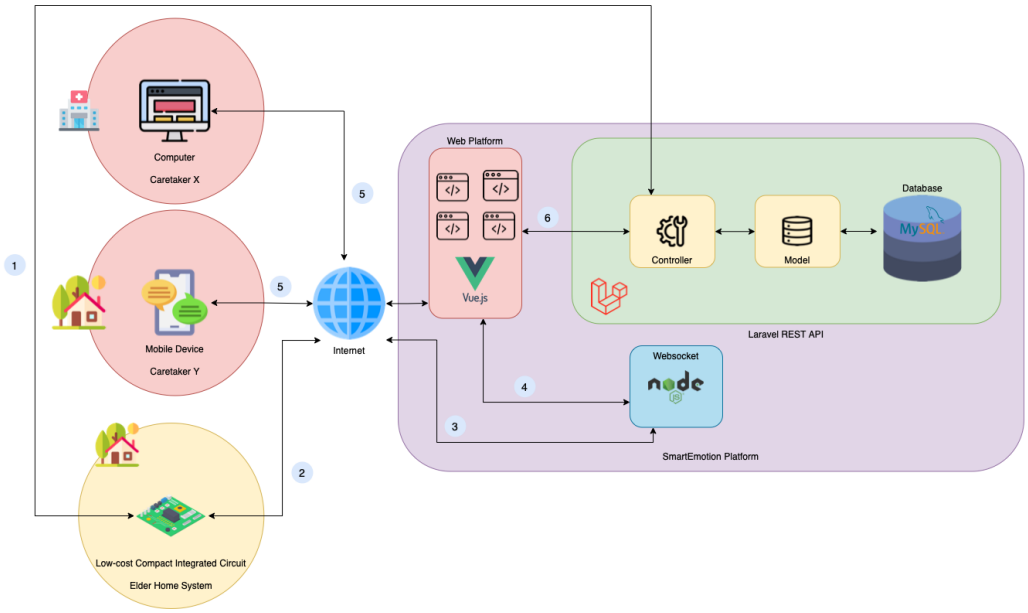


Figura 48: Tecnologias usadas na Plataforma SmartEmotion.

A acessibilidade e a disponibilidade que a plataforma SmartEmotion deve ter é uma preocupação, isto quer dizer que deve estar disponível de qualquer lugar para qualquer cuidador, administrador ou sistema Elder Home em qualquer momento. Assim, toda a plataforma foi publicada num domínio interno do Politécnico de Leiria usando uma máquina virtual *Ubuntu 20.04*. Está disponível com o seguinte endereço: `http://aalemotion.dei.estg.ipleiria.pt`.

A comunicação rápida é um dos pontos fundamentais para despertar um alerta, de modo a manter o cuidador e o administrador informados do estado dos idosos e da própria plataforma SmartEmotion respetivamente. Para tal é necessário que exista um protocolo de comunicação assíncrono, Websockets, entre todo o ecossistema SmartEmotion, esta comunicação abordará quatro fontes de conhecimento, repre-

sentadas na Figura 49: sistema de *logs*, envio de *frames*, notificações e informação de utilizador (Cuidador e Administrador).

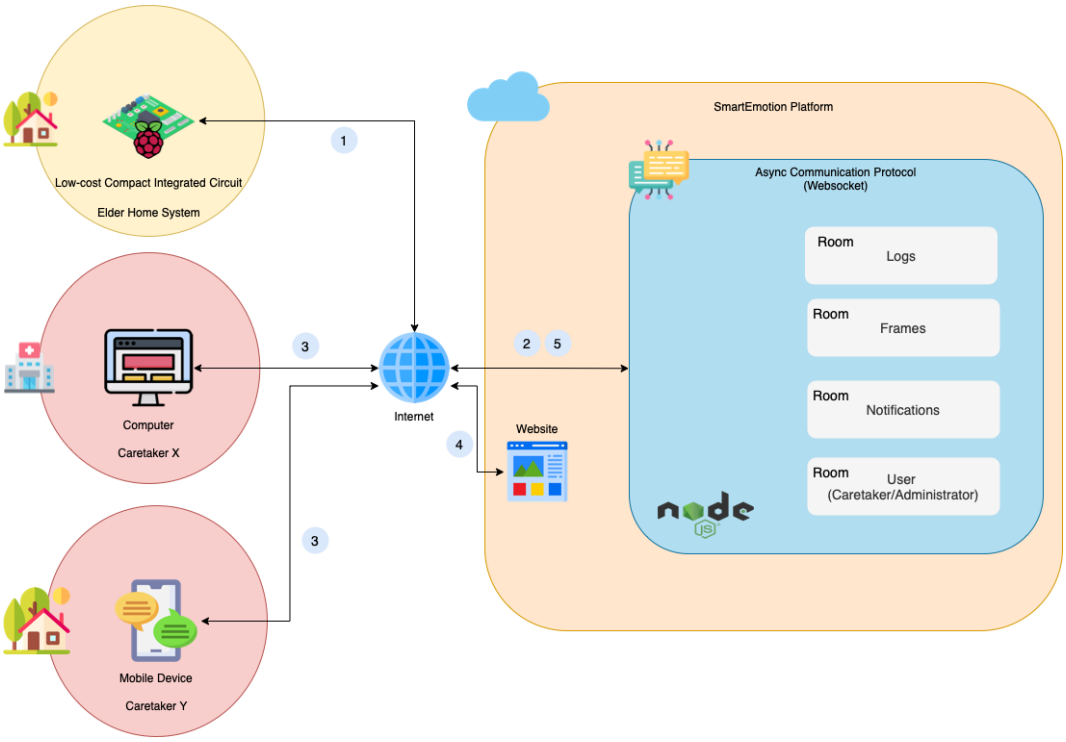


Figura 49: Protocolo de comunicação assíncrona - Websockets na Plataforma SmartEmotion.

A disponibilidade alta é um fator bastante relevante para uma plataforma que oferece serviços críticos, como o alerta de alterações emocionais. A persistência de informação torna a informação disponível para consulta a qualquer momento, a plataforma SmartEmotion persiste os seus dados numa base de dados relacional MySQL, cuja a estrutura e maior detalhe pode ser consultada no Apêndice A. É de salientar algumas relações importantes como: cada sistema Elder Home/cuidador tem as suas notificações e iterações, cada *frame* tem as suas classificações, e cada cuidador pode configurar notificações que pretende receber e o método que quer receber.

A autenticação é uma funcionalidade essencial quando é necessário restringir informação apenas a uma entidade. Na plataforma SmartEmotion cada cuidador apenas pode consultar os dados do idoso ao qual está encarregue e não de outros (Figura 50), assim como a administração da plataforma só pode ser acedida por quem está autorizado.

Assim que o utilizador tenha a sua autenticação bem sucedida terá acesso a uma *dashboard* onde terá uma visão geral dos últimos acontecimentos; caso este utilizador seja um cuidador terá acesso às ultimas notificações de alteração de estado do idoso

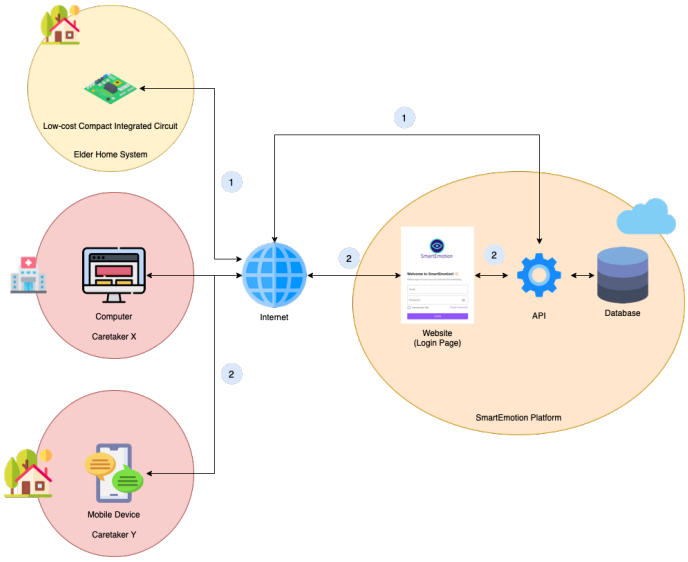


Figura 50: Autenticação à plataforma SmartEmotion.

e a uma visão geral do funcionamento do microcontrolador (Figura 51) – gráfico do número de iterações; gráfico do número de classificações; total de notificações despoletadas; emoção com mais notificações despoletadas; total de iterações; emoção com menor número de notificações despoletadas. Caso o utilizador seja um administrador este terá visível toda a informação correspondente aos microcontroladores – gráfico do número de iterações total; gráfico do número de classificações total; total de notificações despoletadas; emoção com mais notificações despoletadas; total de iterações; emoção com menor número de notificações despoletadas.

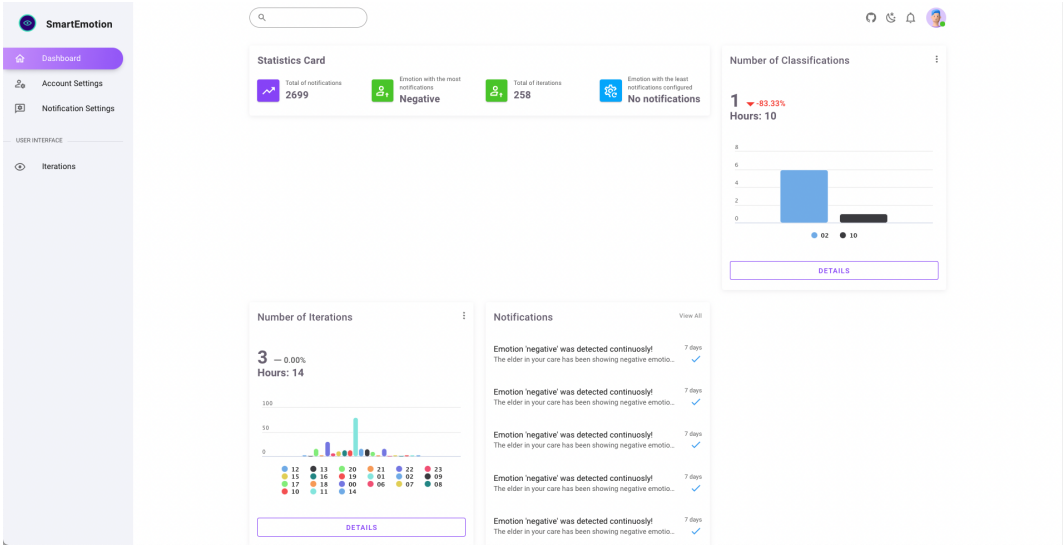


Figura 51: *Dashboard* da plataforma SmartEmotion.

A autenticação permite aos cuidadores acesso a outras funcionalidades como: acesso às iterações de *frames*, gestão e visualização de notificações, e gestão de conta,

e permite aos administradores de sistema gerirem os *logs* de sistema, os próprios administradores e os cuidadores e gestão da própria conta. Estas funcionalidades por meio de menus presentes ao longo da plataforma visíveis na Figura 52.

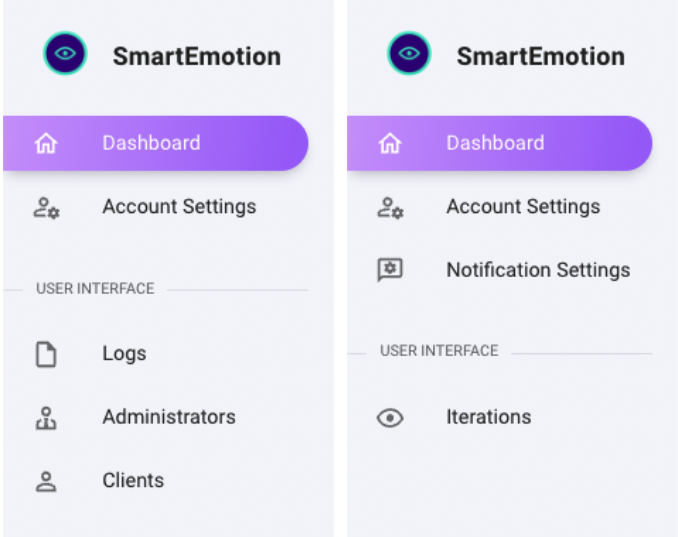


Figura 52: Menus laterais da plataforma SmartEmotion, Administrativo e Cuidador respetivamente.

Ao envio dos melhores *frames* classificados de X em X tempo dá-se o nome de iteração, estas chegam à plataforma por duas fases. Na primeira fase ocorre a inserção singular da iteração contendo o *mac address* do equipamento e a emoção classificada pelo microcontrolador; este pedido devolve uma chave única com um tempo de vida de 30 minutos, e esta chave permite que sejam inseridos *frames* na iteração durante esse tempo. Na segunda fase ocorre a inserção dos *frames* correspondentes com as suas confianças de classificação e datas de captura. Esta segunda fase é repetida tantas vezes quanto o número de melhores *frames* capturados usando a chave devolvida pela primeira fase. A Figura 53 expressa o mecanismo referido acima, este tem como objetivo assegurar que não são inseridos *frames* no contexto errado e por uma entidade não autorizada.

É pelo meio de iterações que é expressado um percurso emocional do idoso, este percurso tanto como as próprias iterações estão visíveis na plataforma como demonstra a Figura 54. Este percurso emocional permite ao cuidador acompanhar o idoso como percurso ou com mais granularidade mostrando o *frame* capturado. Pode ser observado na metade superior da figura a baixo um gráfico correspondente ao percurso emocional do idoso, a linha temporal no eixo das abcissas e a confiança num intervalo de [0 , 100] no eixo das ordenadas.

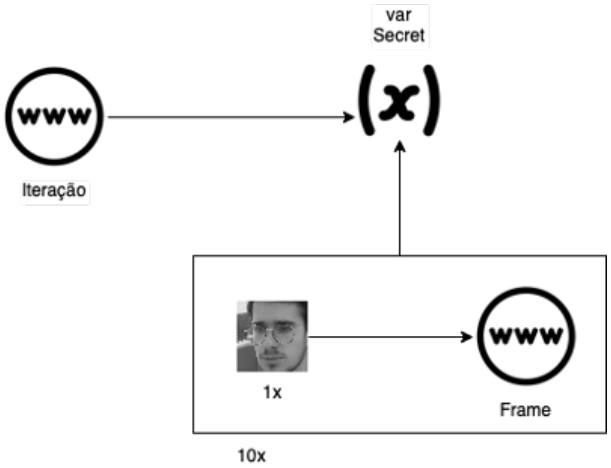


Figura 53: Método de implementação de iterações na plataforma SmartEmotion.

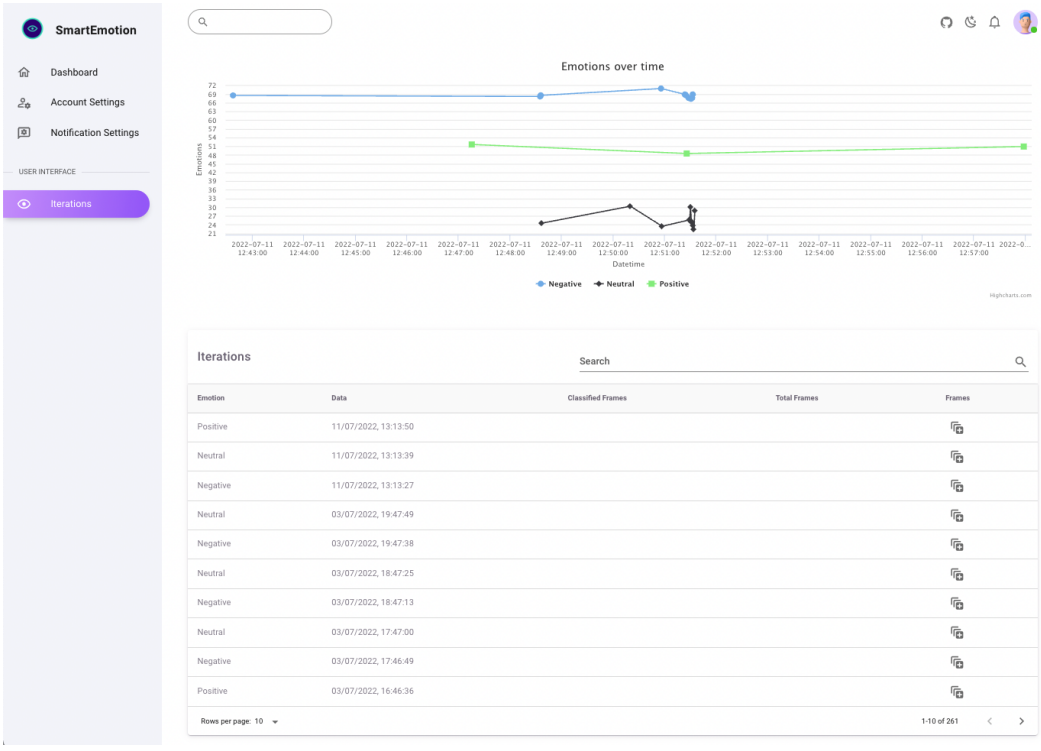


Figura 54: Página de Iterações na plataforma SmartEmotion.

Ao selecionar um ponto no percurso emocional correspondente a um *frame* informações do mesmo serão mostradas, as confianças da classificação e a funcionalidade de classificação humana (*Human Labelling*) como mostra a Figura 55.

As classificações visíveis podem ser vistas por grupo (positiva, negativa e neutra) ou por emoção (dor, felicidade, tristeza, etc).

A funcionalidade de classificação humana tem como objetivo possibilitar a criação um *dataset* que se adapte ao idoso e permita que o modelo no microcontrolador classifique outras emoções; na plataforma apenas é permitido escolher emoções pertencentes ao grupo classificado. Ao classificar um *frame* este irá ser reencaminhado para o microcontrolador para ser catalogado no *dataset* via *Websockets*.

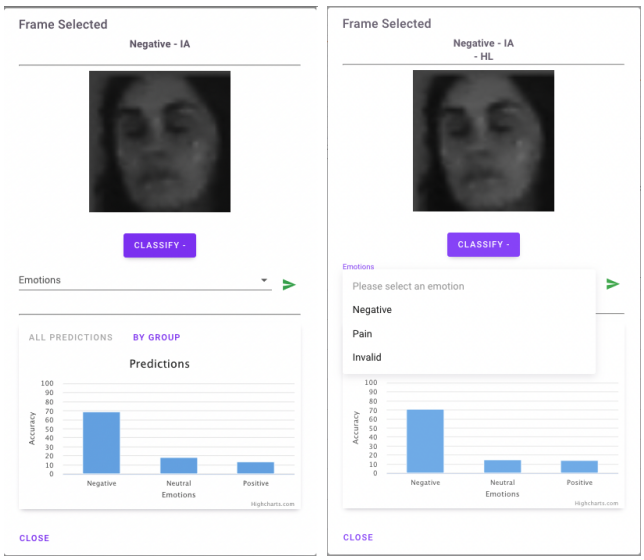


Figura 55: *Frame* na plataforma SmartEmotion.

As iterações, para além de formarem o percurso emocional do idoso podem ainda ser consultadas individualmente numa tabela, como demonstrado na parte inferior da Figura 54. Estas podem ser expandidas de maneira a serem visualizados todos os *frames* correspondentes da iteração, como demonstrado na Figura 56. Ainda nesta página ter-se-á a possibilidade de classificar as imagens visualizando-as todas ao mesmo tempo.

Os *frames* para além de funcionalidades relativas à visualização, iterações ou classificação humana também podem estar relacionados a notificações, que são enviadas singularmente por meio de *Websockets* e pedidos HTTP. A informação contida no seu envio é do tempo que a alteração emocional ocorreu, qual a emoção classificada, a precisão obtida e o próprio *frame*. Estas notificações têm três destinos, a plataforma, email do cuidador e SMS para o telemóvel do cuidador como mostra a Figura 57.

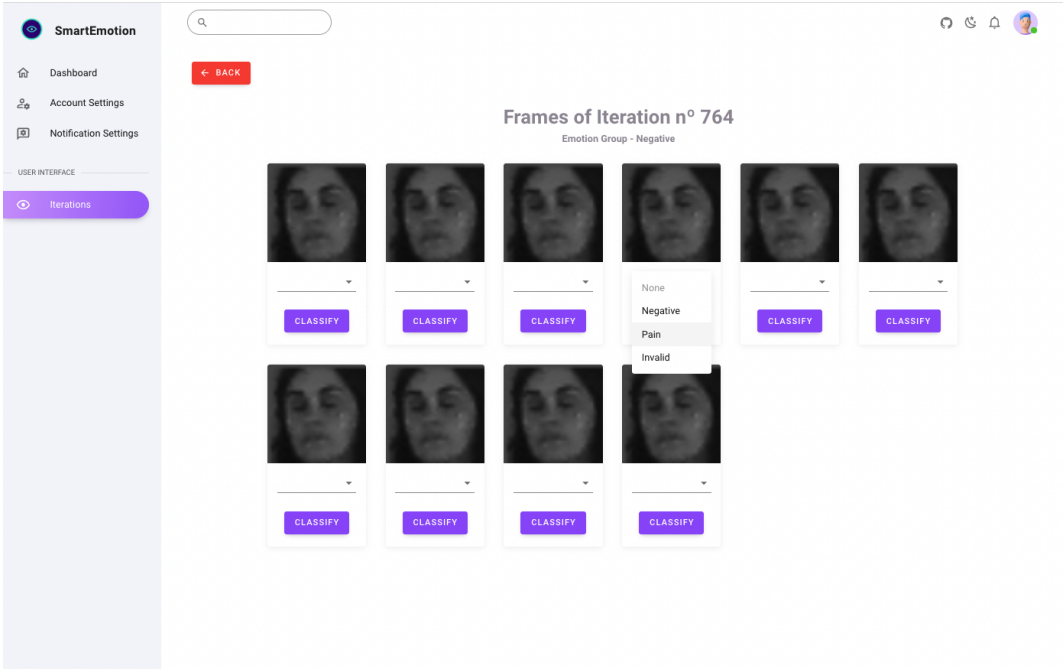


Figura 56: Página de Frames na plataforma SmartEmotion.

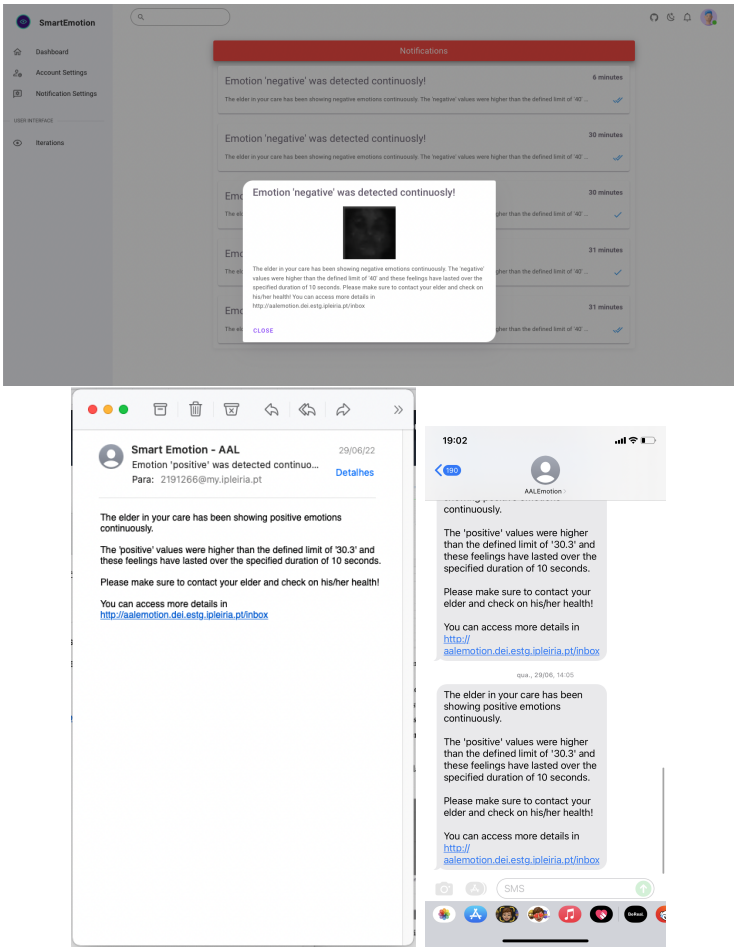


Figura 57: Destinos das notificações, plataforma SmartEmotion, Email e SMS.

As notificações podem ser configuradas como mostra a Figura 58, dando uma precisão/confiança mínima para que seja detetada a emoção e a qual terá de ser mantida durante um período te tempo definido.

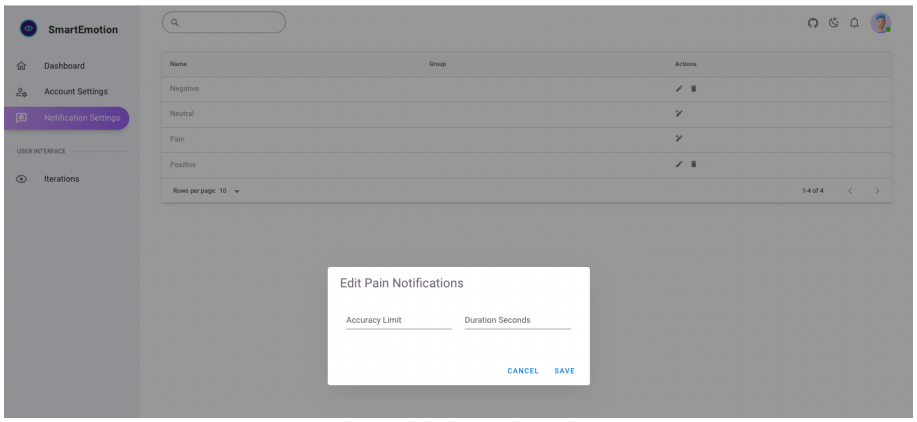


Figura 58: Configuração de notificações na plataforma SmartEmotion.

Relativamente às funcionalidades disponíveis apenas para os administradores da plataforma SmartEmotion, como referido anteriormente e visível no menu esquerdo da Figura 52 é-lhes permitido gerir todos os cuidadores e administradores (Figura 59) e ainda lhes é permitido visualizar todos os *logs* gerados pelos microcontroladores (Figura 60).

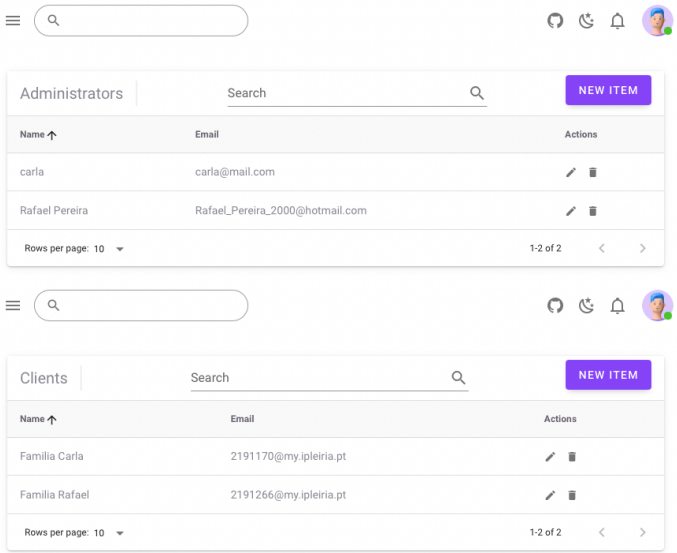
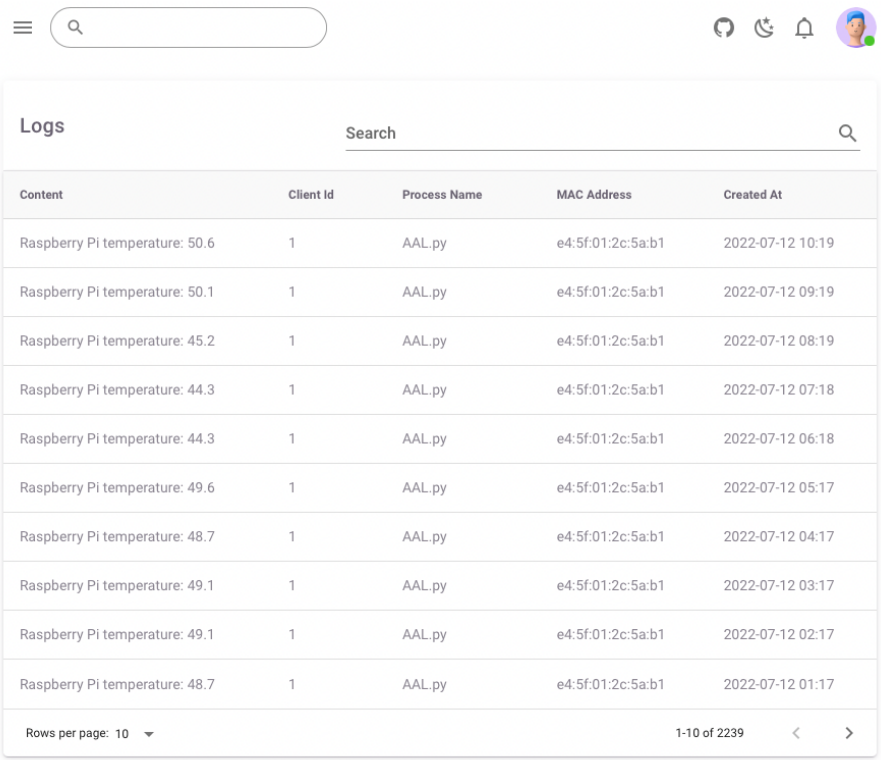


Figura 59: Páginas de Administradores e Cuidadores respetivamente na plataforma SmartEmotion.



Logs

Search

Content	Client Id	Process Name	MAC Address	Created At
Raspberry Pi temperature: 50.6	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 10:19
Raspberry Pi temperature: 50.1	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 09:19
Raspberry Pi temperature: 45.2	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 08:19
Raspberry Pi temperature: 44.3	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 07:18
Raspberry Pi temperature: 44.3	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 06:18
Raspberry Pi temperature: 49.6	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 05:17
Raspberry Pi temperature: 48.7	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 04:17
Raspberry Pi temperature: 49.1	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 03:17
Raspberry Pi temperature: 49.1	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 02:17
Raspberry Pi temperature: 48.7	1	AAL.py	e4:5f:01:2c:5a:b1	2022-07-12 01:17

Rows per page: 10

1-10 of 2239

Figura 60: Página de *logs* na plataforma SmartEmotion.

6

TESTES E RESULTADOS

Neste capítulo serão descritos dois tipos de testes realizados ao sistema SmartEmotion, o primeiro relativo à qualidade da câmara Xiaomi Imilab Full HD onde será testada a distância limite de captação de *frames*; o segundo teste consiste na idealização e aplicação de um cenário de testes em ambiente real e semi-controlado tendo como principal objetivo a perceção do estado de funcionamento do sistema SmartEmotion quando exposto a acontecimentos não expectáveis e fora do ambiente controlado. Finalmente, serão discutidos os resultados dos testes com auxílio a figuras capturadas durante os mesmos.

6.1 TESTES À QUALIDADE DA CÂMARA

A qualidade da câmara é um ponto crucial para a correta deteção de emoções. Se um *frame* estiver com má qualidade será bastante difícil haver uma deteção de emoções bem sucedida.

Nesta secção iremos mostrar alguns testes realizados à captura de *frames*, capturados em diferentes distâncias entre a câmara e o indivíduo.

Estes testes têm como objetivo procurar o limite da câmara de modo a perceber qual a distância que num contexto real esta pode estar dos idosos. Será apenas avaliado se o sistema consegue detetar um rosto ou uma emoção; caso não consiga consideramos falta de qualidade da imagem/câmara.

Para os testes foi usada uma câmara Xiaomi Imilab Full HD [69], esta tem características como: fácil de usar, para usar basta conectar, preço baixo ($\sim 25e.Relativamenteàluminosidade, sendoqueosistematerálugarnumacasadeumidosoospresentes$

A nossa principal preocupação foi de simular um ambiente que fosse de encontro ao quotidiano de um idoso de maneira a que houvesse uma fidelidade nos resultados dos testes. De modo a que seja simulado o ambiente onde as pessoas idosas passam o seu dia sentadas a ver televisão, colocámos a câmara ao nível de uma mesa (74 centímetros de altura) como demonstrado na Figura 61 capturando *frames*/imagens de um individuo sentado numa cadeira. A distância entre a cadeira e a câmara

Especificações	
Resolução	Full HD 1080p
Ângulo de visão	90º
Design	Alça dobrável
Correção de Luminosidade	Sim
Microfone Embutido	Sim
Cancelamento de Ruído	Sim
Megapixéis	2mp

Tabela 20: Especificações da webcam Xiaomi Imilab Full HD.

aumenta de teste em teste de modo a ser percebido o limite do nosso sistema. A distância da cadeira em cada teste aumenta 80 centímetros, começando numa distância de aproximadamente um metro.

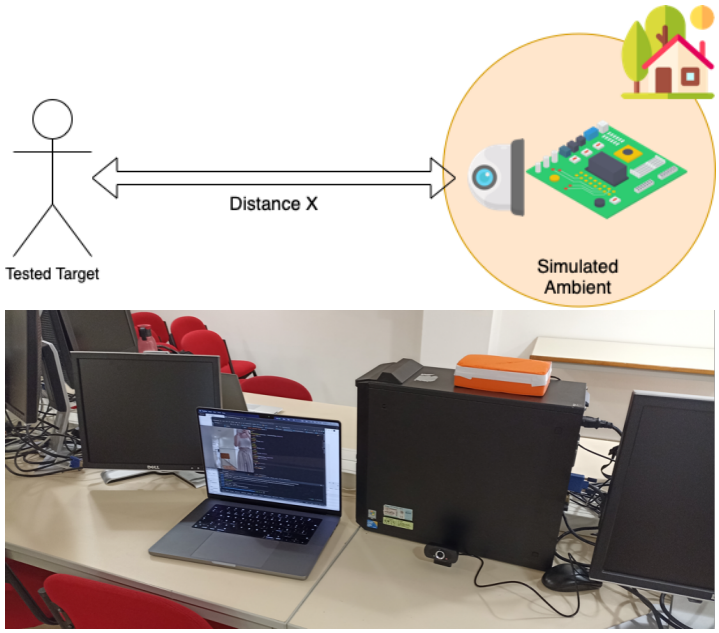


Figura 61: Disposição de equipamentos para teste de captura de *frames* em diferentes distâncias.

Segue-se uma tabela representativa de todos os testes realizados à qualidade da câmara de maneira a que seja percetível a variação entre teste e os seus resultados. Posteriormente será feita uma breve discussão sobre a mesma de modo a sintetizar os resultados obtidos no contexto do problema de deteção de emoções em ambiente semi-controlado. A Figura 62 representa o percurso dos resultados obtidos, onde ao aumentarmos a distância entre o idoso e a câmara percebemos que existe maior dificuldade na deteção de rosto e consequentemente emoções.

Nº Teste	Distância (Metros)	Nº Deteções em 30 segundos
1	1m	21
2	1,8m	24
3	2,6m	23
4	3,4m	27
5	4,2m	16
6	5m	1

Tabela 22: Resultados obtidos do teste à qualidade da câmara do sistema Elder Home.

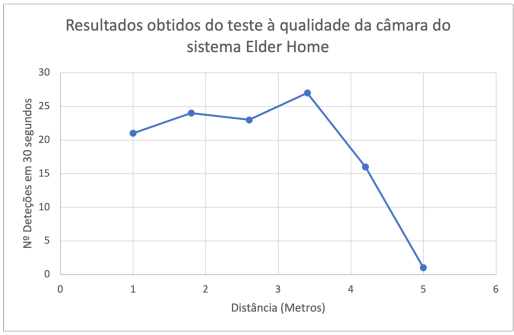


Figura 62: Percurso dos resultados obtidos do teste à qualidade da câmara do sistema Elder Home.

Numa análise à Figura 62 podemos observar como já referido um decréscimo no número de deteções de rosto/emoções à medida que nos vamos afastando da câmara, chegando a um limite de cinco metros onde onde apenas foi detetado 1 *frame*. É visível na Figura 63 a distância entre a cadeira e a câmara ao qual foram capturados os resultados visíveis na Figura 64, onde cada segundo, coordenada do eixo das abcissas, corresponde a uma deteção. Este limite será posteriormente considerado em testes em ambiente semi-controlado para o posicionamento dos sistemas Elder Home e do local de captação.

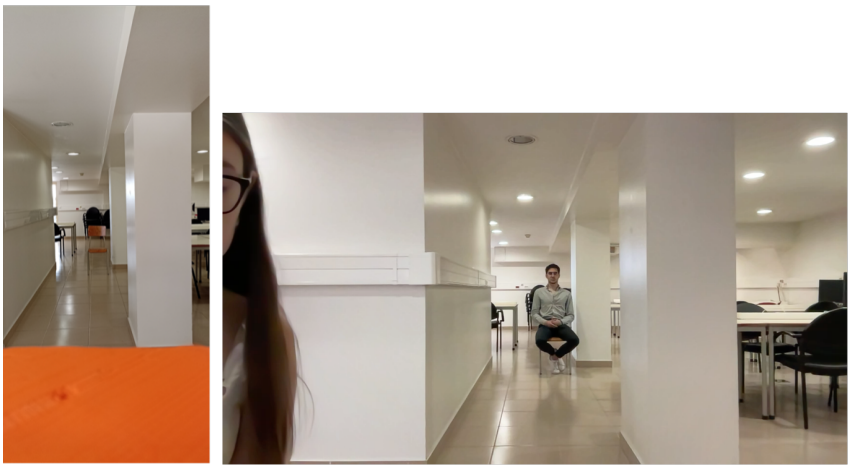


Figura 63: Distância entre a cadeira e o equipamento de teste à qualidade da câmara, sexto teste - 5 metros.

Em suma, podemos assumir que o sistema Elder Home tem um comportamento expectável até a uma distância de 5 metros, onde com base nos testes podemos assumir que qualquer distância inferior é ideal para testes em ambiente semi-controlado.

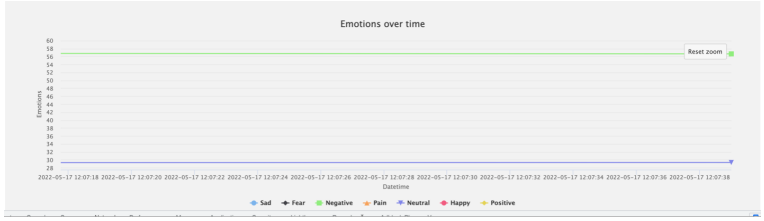


Figura 64: Resultado do teste à qualidade da câmara a 5 metros.

6.2 CENÁRIO DE TESTES

A presente secção visa descrever os requisitos que um cenário de teste semi-controlado deve ter para uma aproximação com a realidade para o bom funcionamento do sistema Elder Home. Serão abordados tópicos relativos ao hardware necessário e algumas características da infraestrutura destino onde o sistema Elder Home será instalado.

Os idosos passam a maior parte do seu tempo em suas casas, isolados a ver televisão e a descansar no sofá [50]. Este facto leva a que ambiente de habitação diário dos idosos seja um bom ambiente de cenário de testes. Para tal é preciso também perceber que infraestruturas é expectável encontrar de modo a preparar o direcionamento e distanciamento da câmara ao idoso.

De modo a adaptar o cenário de testes ao idoso é necessário perceber quais as divisões onde passa a maior parte do tempo – tipicamente são a sala, cozinha e quarto. Nestes testes semi-controlados será considerada a sala como ambiente de teste, tendo em conta que esta, para além de ser a divisão mais frequentada, é a mais adequada neste contexto semi-controlado para a realização destes testes, já que geralmente existe maior acesso à *internet*.

Quando é referida a sala de estar para a instalação do sistema Elder Home, torna-se relevante saber qual a distância entre o sofá, lugar expectável que o idoso esteja, e a televisão. Esta distância irá variar consoante o tamanho da divisão, da televisão e até pela preferência do idoso. Considerando que a distância do sofá e a televisão não ultrapassa os cinco metros [13] podemos concluir que o limite da câmara testado na secção anterior satisfaz os requisitos como demonstrado na Figura 65.

Os requisitos técnicos deste cenário de testes passam pela necessidade de acesso à *internet*, de modo a que o sistema Elder Home consiga realizar pedidos à plataforma SmartEmotion.

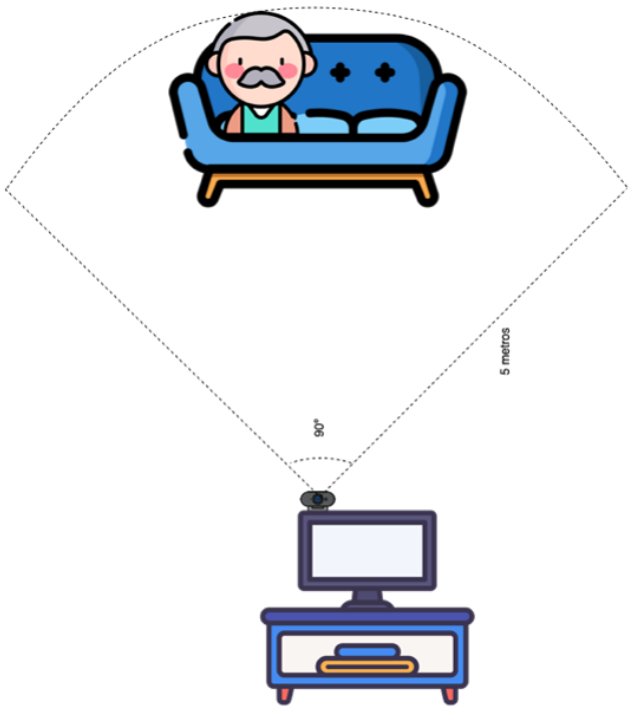


Figura 65: Cenário de testes em ambiente semi-controlado.

A execução dos testes terá lugar em duas residências em ambiente semi-controlado. Sendo que esta caraterização dos testes deve-se ao facto de estes serem realizados nas residências dos autores da solução (Figuras 66 e 67) e tendo em conta que os alvos dos testes são os respetivos familiares. Os objetivos com a realização destes testes são:

- Perceber qual o estado de funcionamento dos sistemas fora de um ambiente controlado e possíveis problemas;
- Perceber qual o impacto que a utilização e a presença do sistema SmartEmotion terá nos familiares envolvidos nos testes;
- Personalização de pelo menos uma emoção na deteção de emoções pelo sistema Elder Home.

Os testes, realizados durante os meses de junho e julho nas casas A e B, envolveram a captura de *frames* durante dias seguidos e tal como foi mencionado acima, a divisão escolhida para a localização do sistema Elder Home foi a sala de estar em ambas as residências. Os indivíduos de cada uma das famílias concordam no imediato com a utilização do sistema SmartEmotion assumindo simultaneamente os papéis de cuidador e cuidando. No entanto, demonstraram, como é de esperar, uma preocupação leve sobre a sua privacidade relativamente à captura e persistência das

Figura 66: Casa de Teste A.

Figura 67: Casa de Teste B.

suas imagens. Após uma breve explicação sobre o envio e persistência dos *frames* as suas preocupações deixaram de existir. Os sujeitos de teste disponibilizaram-se ainda para explorar a plataforma SmartEmotion e realizar o *Human Labelling*.

Durante a realização dos testes ocorreram algumas situações inesperadas, nomeadamente o reconhecimento errado de um objeto como uma face (Figura 68), falha no envio de alguns *frames* (Figura 69), sessão do utilizador no sistema Elder Home a expirar (o que levava a que o sistema Elder Home deixasse de funcionar) e ainda o sobreaquecimento do equipamento após poucas horas de funcionamento. Alguns destes tópicos já foram abordados no capítulo 5. No entanto, estes erros foram descobertos durante uma fase inicial dos testes o que levou à realização de diversas alterações em todo o sistema SmartEmotion, e.g alterações no envio dos *frames* para a plataforma SmartEmotion, implementação de um mecanismo de tentativas, mecanismo de *refresh_token* e utilização de ventoinhas incorporadas no Raspberry Pi. A ocorrência destes problemas em ambientes semi-controlados tornou-se uma mais valia para o melhoramento de todo o Sistema Elder Home. Visto que, os mesmos erros num ambiente não-controlado trariam consequências graves relativamente à opinião dos utilizadores e poderiam causar danos ao equipamento - devido ao sobreaquecimento mencionado anteriormente.

Durante os testes diversas iterações (Figuras 70 e 72) e notificações (Figuras 71 e 73) foram enviadas. O sistema Elder Home causou surpresas relativamente à deteção de rostos independentemente do nível de luminosidade do ambiente, orientação e posicionamento dos rostos.

A maioria das interações com a plataforma SmartEmotion na casa de testes A foi realizada pela pessoa que era fluente na língua Inglesa e apresentou mais tempo disponível (de entre todos os participantes nos testes da casa A). Desta forma, esta pessoa assumiu assim o papel de cuidador sendo que durante a utilização da plataforma SmartEmotion foi obtido algum *feeback* que permitirá a evolução do todo o sistema no futuro. Desta forma o sujeito de testes relatou que achou a página da *dashboard* confusa durante o tempo de utilização. Outra das dificuldades sentidas foi associar que a coluna - *frames* - da tabela das iterações (Figura **??**) levava à classificação dos *frames*.

Ao testar o tema *dark* da plataforma o utilizador notou que as letras tornavam-se difíceis de ler Para além disto, o utilizador perdeu-se após classificar diversos *frames* provenientes de iterações diferentes. Principalmente devido ao facto que ao retornar à página das iterações a tabela reaparece sempre na primeira página, ignorando a posição onde se encontra a última iteração visitada pelo utilizador. Um ponto

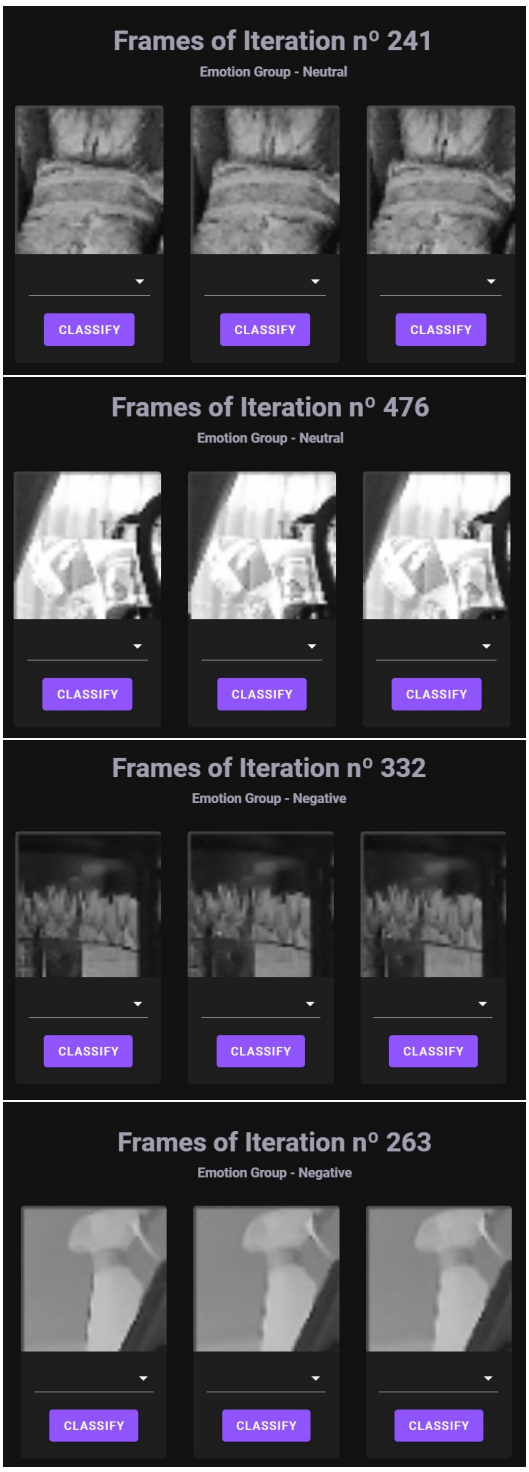


Figura 68: Deteção de objetos.

```
Notification was sent
Request succeded after 0 attempts
Request succeded after 0 attempts
2 iterations were performed successfully
0 iterations were performed successfully
0 iterations were performed successfully
Notification was sent
Request failed - Attempts: 1
Request failed - Attempts: 2
Request failed - Attempts: 3
Request negative failed - Attempts: 3
Request failed - Attempts: 1
Request failed - Attempts: 2
Request failed - Attempts: 3
Request neutral failed - Attempts: 3
2 iterations were unsucessfull
Notification was sent
Request failed - Attempts: 1
Request failed - Attempts: 2
Request failed - Attempts: 3
Request negative failed - Attempts: 3
Request failed - Attempts: 1
Request failed - Attempts: 2
Request failed - Attempts: 3
Request neutral failed - Attempts: 3
2 iterations were unsucessfull
0 iterations were performed successfully
0 iterations were performed successfully
0 iterations were performed successfully
0 iterations were performed successfully
0 iterations were performed successfully
0 iterations were performed successfully
```

Logs

Search

Content	Client Id	Process Name	MAC Address	Created At
Cliente Ligado	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:08
Cliente Ligado	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:17
An error occurred in the iteration log 2 of 3	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:20
An error occurred in the iteration log 2 of 3	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:21
An error occurred in the iteration log 0 of 2	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:22
An error occurred in the iteration log 0 of 2	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:24
An error occurred in the iteration log 0 of 2	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:33
Cliente Ligado	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:33
An error occurred in the iteration log 1 of 3	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:35
An error occurred in the iteration log 0 of 2	1	AAL.py	e4:5f:01:2c:59:08	2022-06-14 20:36

Figura 69: Erro no envio de *frames*.

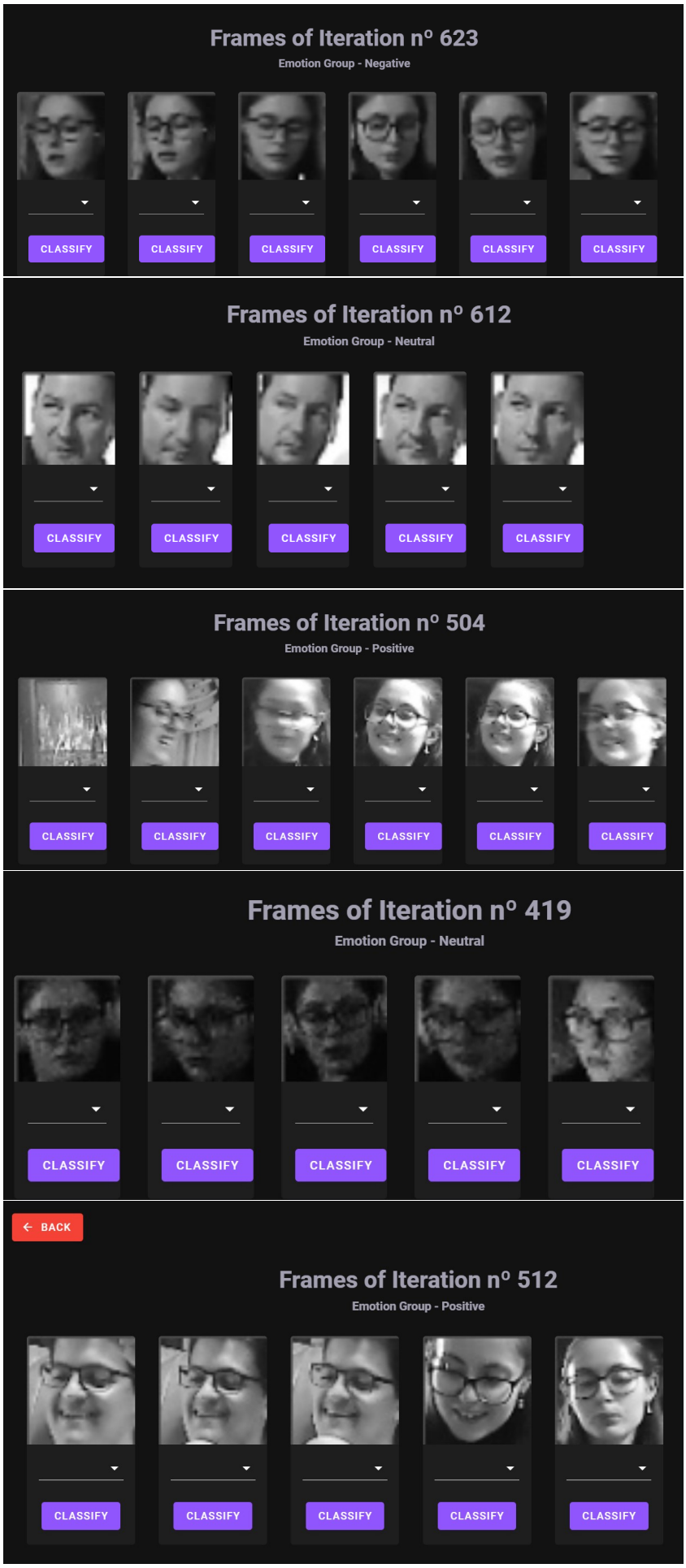


Figura 70: Exemplo de Iterações na habitação A.

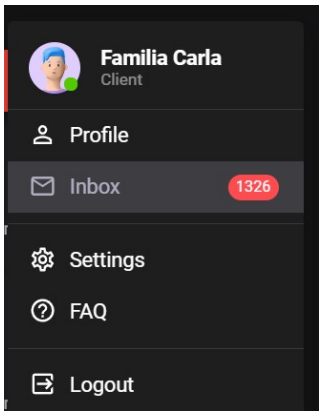


Figura 71: Número de Notificações na casa A.

que o participante destacou foi a possibilidade de evolução da qualidade de *frames* através da aquisição de uma câmara que apresente maior qualidade.

Em suma, os testes, cujo objetivo consistiu em avaliar o funcionamento do sistema Elder Home e a usabilidade da plataforma SmartEmotion, foram bastante positivos. A pessoa envolvida nos testes apresentou agilidade a utilizar a plataforma e apenas divulgou opiniões e ideias para melhorar facilitar a utilização da mesma. Posto isto, seria uma boa abordagem fornecer breves tutoriais que expliquem a utilização da plataforma de maneira a introduzir os cuidadores a todas as funcionalidades da plataforma.

Na casa de testes A, foram classificadas com a emoção de tristeza - através de *Human Labelling* - um total de 106 *frames* o que significa que o limite de 100 *frames* foi ultrapassado (valor original do limite que poderá ser modificado). Desta forma, o retreino do modelo inicial ocorreu sendo que foram utilizados os parâmetros originais e a única mudança foi o crescimento do nº de emoções presentes no *dataset*. O processo de retreino teve a duração de aproximadamente 4 horas onde ocorreram 66 épocas (Figura 74). O equipamento apresentou valores médios temperatura de 49.6ºC.

O teste ao retreino foi realizado com o objetivo de averiguar o correto funcionamento do retreino após o *Human Labelling* ter sido concluído e o limite de *frames* atingido. No entanto, não é possível ainda efetuar a deteção da emoção de tristeza devido à disparidade entre o nº de *frames* das emoções já existentes.

Em suma, os testes em ambiente semi-controlados obtiveram resultados técnicos e sociais bastante positivos sendo que, apesar de o sistema ter apresentado problemas iniciais, após a sua resolução o funcionamento foi imensamente positivo. Para além disso, os familiares demonstram sentimentos de confiança no sistema SmartEmotion

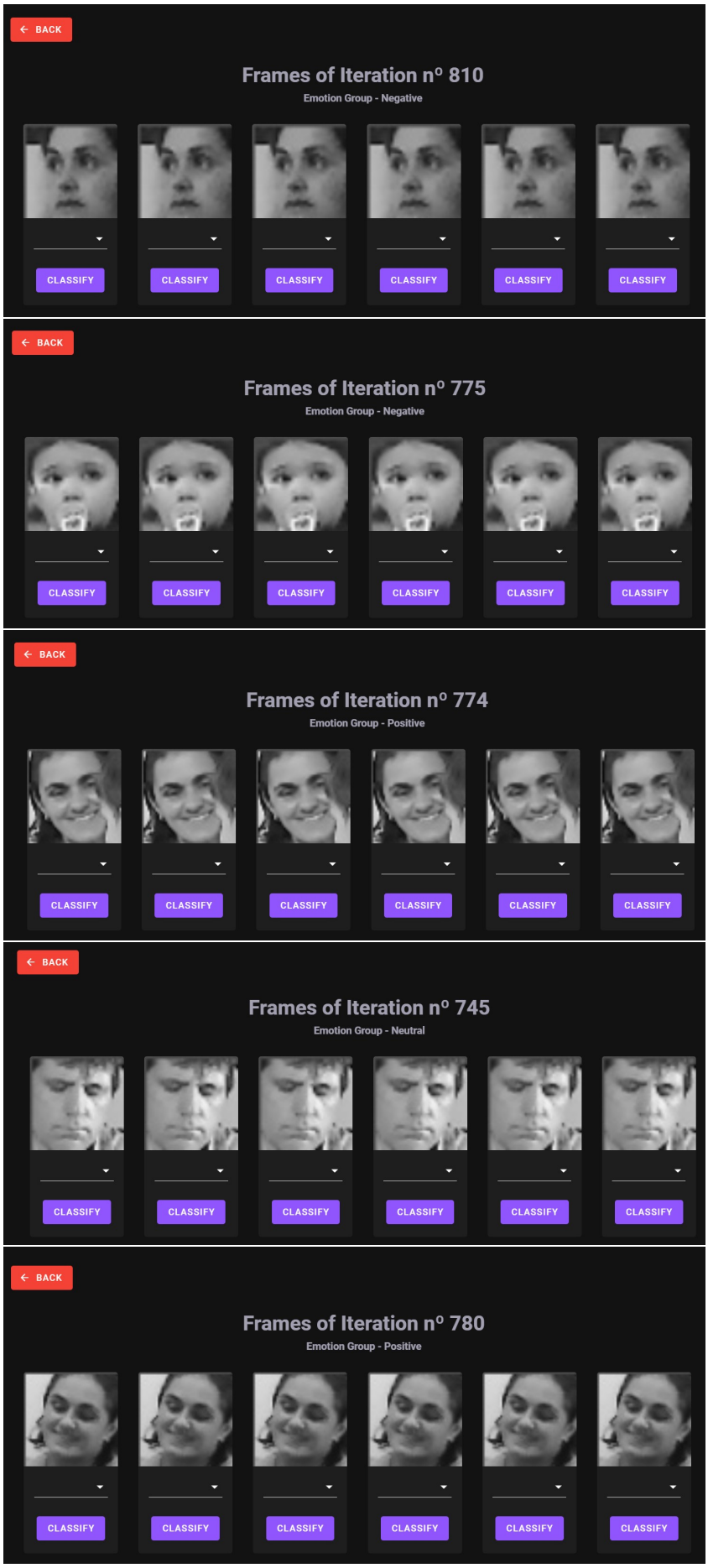


Figura 72: Exemplo de Iterações na habitação B.

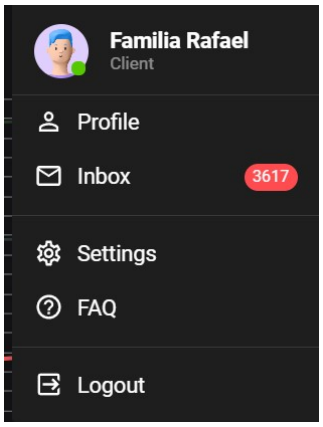


Figura 73: Número de Notificações na casa B.

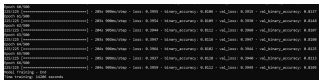


Figura 74: Retreino do modelo para uma emoção personalizada.

estando à vontade com a sua presença nas suas habitações e a sua utilização constituindo assim uma contribuição social de grande valor onde o nosso sistema fornece tanto aos cuidadores e cuidandos sentimentos de tranquilidade e conforto.

7

CONCLUSÕES

A solidão das pessoas idosas em suas casas é um problema atual, que se agravou mais ainda com a pandemia COVID-19. No entanto, a tecnologia poderá ser uma boa maneira de resolver este problema principalmente tendo em conta a ligação já existente entre a área da tecnologia e saúde. Desta forma foi proposta a criação de um sistema que permite realizar reconhecimento facial e deteção de emoções.

Inicialmente foram assim investigadas algumas das ferramentas já existentes no mercado, abordagens possíveis de utilizar e ainda *datasets*. Relativamente às ferramentas comercializáveis destaca-se a ferramenta *Affectiva* [5] que através de visão computacional permite a deteção de estados emocionais e físicos dos ocupantes dos veículos. De entre todas as abordagens exploradas destacaram-se *Deep Learning* e *Transfer Learning* que constituem as abordagens utilizadas na solução concebida.

Posteriormente foram analisados e testados os modelos estado de arte onde se concluiu que o *DeepFace* é o melhor modelo comparativamente com todos os outros, visto que apresentou o melhor valor de precisão e menor tempo de retreino.

A solução: sistema SmartEmotion, foi assim desenvolvida durante o período de um semestre partindo dos requisitos iniciais e utilizando a técnica de prototipagem evolutiva. O sistema SmartEmotion divide-se em sistema Elder Home e Plataforma SmartEmotion. O sistema Elder Home, presente na casa da pessoa idosa permite então adquirir imagens, efetuar reconhecimento facial e deteção de emoções, envio de notificações e *logs* de sistema, e realizar o treino do modelo. Já a plataforma SmartEmotion permite ao cuidador classificar as imagens da pessoa idosa ao seu encargo, parametrizar as notificações e o envio de alertas. A plataforma permite ainda aos administradores realizar a gestão do serviço.

Durante todo o semestre, imensuráveis foram as horas de investigação, criação e testagem de protótipos de maneira a obter a melhor solução possível. Após ter sido finalizado o protótipo final este foi testado em dois ambientes semi-controlados de maneira a testar o funcionamento do sistema SmartEmotion num contexto real. No geral, os resultados foram muito positivos: diversos rostos e emoções foram detetados de forma contínua, inúmeras notificações foram despoletadas corretamente, sendo toda a informação apresentada na plataforma. Durante os testes ocorreu ainda o

retreino do modelo devido a ter sido ultrapassado o limite de *frames* classificados para para uma emoção personalizada.

Tendo em conta os resultados dos testes podemos concluir que o sistema SmartEmotion apresenta diversas contribuições técnicas, científicas e sociais. A nível de contribuições sociais, permite o aumento do bem-estar e segurança das pessoas idosas simultaneamente aumentando o conforto e tranquilidade dos seus familiares na medida que permite que os idosos continuem a residir nas suas casa e ainda permite que cuidadores sejam notificados quando algo de estranho ocorre.

Ainda relativamente às contribuições sociais, o sistema permitirá diminuir a quantidade de pessoas institucionalizadas e os custos associados visto que os idosos terão melhores condições para continuar a residir em suas casas em segurança com a utilização do nosso sistema.

Relativamente às contribuições técnicas são de ressalvar o desenvolvimento de um sistema IoT personalizado que permite realizar a deteção e o acompanhamento do estado emocional das pessoas em tempo real e de forma contínua; e ainda o facto de o sistema apresentar um baixo custo para os seus utilizadores e ser constituído por poucos equipamentos – o que carateriza a solução como *low-cost*.

A nível de contribuições científicas, a solução desenvolvida permite a deteção e monitorização contínua de diversas emoções, inicialmente focando-se nas emoções gerais (positivo, negativo e neutral), com possibilidade para evoluir de maneira a detetar emoções específicas como tristeza, raiva, dor, etc. O sistema carateriza-se como auto-evolutivo, personalizável visto que o utilizador consegue personalizar diversos componentes do sistema, e.g a escolha das emoções a detetar, parametrização das notificações, ocorrência e parametrização do retreino, etc. O utilizador através da classificação das imagens recolhidas do sistema, instalado na residência da pessoa idosa ao seu cuidado, ao longo do tempo poderá assim escolher quais as emoções a detetar pelo modelo. Para além disso, o utilizador poderá também optar por realizar retreinos do modelo quando entender e com os parâmetros que preferir.

BIBLIOGRAFIA

[1] *0*xNjEPIZmPvKeqss6 (1000×694)*. URL: https://miro.medium.com/max/1400/0*xNjEPIZmPvKeqss6 (acedido em 05/07/2022).

[2] *1-s2.0-S095070512100397X-gr5_lrg.jpg (2917×2923)*. URL: https://ars.els-cdn.com/content/image/1-s2.0-S095070512100397X-gr5_lrg.jpg (acedido em 02/07/2022).

[3] *2009-056 - The CMU Multi-PIE Face Database | CMU Flintbox*. URL: https://cmu.flintbox.com/technologies/67027840-27d5-4570-86dd-ad4715ef3c09 (acedido em 06/07/2022).

[4] *A survey on sentiment analysis methods, applications, and challenges | SpringerLink*. URL: https://link.springer.com/article/10.1007/s10462-022-10144-1 (acedido em 21/03/2022).

[5] Affectiva. *Interior Sensing AI*. en. URL: https://go.affectiva.com/auto (acedido em 05/07/2022).

[6] *AI solutions on IBM Cloud*. en-us. Set. de 2021. URL: https://www.ibm.com/cloud/ai (acedido em 06/07/2022).

[7] *API - CrowdEmotion Developers*. URL: https://developers.crowdemotion.co.uk/api/ (acedido em 28/06/2022).

[8] *Applications - Lexalytics*. en-US. Mai. de 2022. URL: https://www.lexalytics.com/applications/ (acedido em 06/07/2022).

[9] Tadas Baltrusaitis. *OpenFace 2.2.0: a facial behavior analysis toolkit*. original-date: 2016-03-05T20:08:49Z. Jul. de 2022. URL: https://github.com/TadasBaltrusaitis/OpenFace (acedido em 07/07/2022).

[10] Marouane Birjali, Mohammed Kasri e Abderrahim Beni-Hssane. «A comprehensive survey on sentiment analysis: Approaches, challenges and trends». en. Em: *Knowledge-Based Systems* 226 (ago. de 2021), p. 107134. ISSN: 0950-7051. DOI: 10.1016/j.knosys.2021.107134. URL: https://www.sciencedirect.com/science/article/pii/S095070512100397X (acedido em 21/03/2022).

[11] Gaudenz Boesch. *VGG Very Deep Convolutional Networks (VGGNet) - What you need to know*. en-US. Out. de 2021. URL: https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/ (acedido em 13/07/2022).

[12] Jason Brownlee. *How to use Learning Curves to Diagnose Machine Learning Model Performance*. en-US. Fev. de 2019. URL: https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/ (acedido em 06/07/2022).

[13] Mónica Carvalho. *Como saber qual é a distância correta entre a TV e o sofá?* pt-PT. Section: Decoração e Bricolage. Out. de 2021. URL: https://www.e-konomista.pt/distancia-correta-entre-tv-sofa/ (acedido em 13/07/2022).

[14] Chris. *Tensorflow VGG16 and VGG19*. original-date: 2016-03-16T22:17:14Z. Jul. de 2022. URL: https://github.com/machrisaa/tensorflow-vgg/blob/dcc2449a07201fb4ac99a98dc7bb95915322306c/vgg16.py (acedido em 07/07/2022).

[15] *Data Labeling Platform for Computer Vision | BasicAI*. URL: https://www.basic.ai/tools/software/ (acedido em 06/07/2022).

[16] *Data-augmentation-using-semantic-preserving-transformation-for-SBIR.png (850×587)*. URL: https://www.researchgate.net/publication/319413978/figure/fig2/AS:533727585333249@1504261980375/Data-augmentation-using-semantic-preserving-transformation-for-SBIR.png (acedido em 04/07/2022).

[17] *Detecta rostos | API Cloud Vision*. pt-BR. URL: https://cloud.google.com/vision/docs/detecting-faces?hl=pt-br (acedido em 21/03/2022).

[18] *Detecting and analyzing faces - Amazon Rekognition*. URL: https://docs.aws.amazon.com/rekognition/latest/dg/faces.html (acedido em 21/03/2022).

[19] *Difference between batch size & epoch | Data Science and Machine Learning*. en. URL: https://www.kaggle.com/general/a (acedido em 06/07/2022).

[20] *EmoVu by Eyeris*. URL: http://emovu.com (acedido em 06/07/2022).

[21] *Evaluation Metrics for Classification problems | Data Science and Machine Learning*. en. URL: https://www.kaggle.com/getting-started/a (acedido em 07/07/2022).

[22] *faces.png (910×336)*. URL: https://cloud.google.com/static/vision/docs/images/faces.png (acedido em 05/07/2022).

[23] «Facial Action Coding System». Em: *The SAGE Encyclopedia of Communication Research Methods*. 2455 Teller Road, Thousand Oaks California 91320: SAGE Publications, Inc, 2017. DOI: 10.4135/9781483381411.n178. URL: https://sk.sagepub.com/reference/the-sage-encyclopedia-of-communication-research-methods//i5129.xml (acedido em 21/03/2022).

[24] *Facial Action Coding System (FACS) - A Visual Guidebook*. en. Ago. de 2019. URL: https://imotions.com/blog/facial-action-coding-system/ (acedido em 21/03/2022).

[25] *Facial expression recognition software | FaceReader*. URL: https://www.noldus.com/facereader (acedido em 21/03/2022).

[26] *FER-2013*. en. URL: https://www.kaggle.com/msambare/fer2013 (acedido em 06/04/2022).

[27] *Five Ways to Run a Program On Your Raspberry Pi At Startup*. en-US. URL: https://www.dexterindustries.com/howto/run-a-program-on-your-raspberry-pi-at-startup/ (acedido em 13/07/2022).

[28] *Google Colaboratory*. pt. URL: https://colab.research.google.com/?utm_source=scs-index (acedido em 13/07/2022).

[29] Ralph Gross et al. «Multi-PIE». en. Em: *Image and Vision Computing* 28.5 (mai. de 2010), pp. 807–813. ISSN: 02628856. DOI: 10.1016/j.imavis.2009.08.002. URL: https://linkinghub.elsevier.com/retrieve/pii/S0262885609001711 (acedido em 06/04/2022).

[30] *Hands-On Guide To Loss Functions*. en. URL: https://kaggle.com/code/viveknimsarkar/hands-on-guide-to-loss-functions (acedido em 07/07/2022).

[31] *Healthcare AI*. en. URL: https://www.nviso.ai/smart-health (acedido em 06/07/2022).

[32] HubSpot. *Customer Feedback Software | HubSpot*. en-US, en. URL: https://www.hubspot.com/products/service/customer-feedback (acedido em 05/07/2022).

[33] *ImageNet*. URL: https://www.image-net.org/ (acedido em 13/07/2022).

[34] *images (309×163)*. URL: https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQSwJXhkSQ551h3cmoEhI7eBhTBpvbhEqEecg&usqp=CAU (acedido em 06/07/2022).

[35] Michal Kawulok, M. Emre Celebi e Bogdan Smolka, eds. *Advances in Face Detection and Facial Image Analysis*. en. Cham: Springer International Publishing, 2016. DOI: 10.1007/978-3-319-25958-1. URL: http://link.springer.com/10.1007/978-3-319-25958-1 (acedido em 22/03/2022).

[36] Aimal Khan. *10 Best Microcontroller Boards for Engineers and Geeks*. en-us. Jan. de 2021. URL: https://www.engineeringpassion.com/10-best-microcontroller-boards-for-engineers-and-geeks/ (acedido em 21/03/2022).

[37] *landmarks.1.jpg (496×329)*. URL: https://docs.microsoft.com/pt-pt/azure/cognitive-services/computer-vision/media/landmarks.1.jpg (acedido em 05/07/2022).

[38] Patrick Lucey et al. «Painful data: The UNBC-McMaster shoulder pain expression archive database». en. Em: *Face and Gesture 2011*. Santa Barbara, CA, USA: IEEE, mar. de 2011, pp. 57–64. ISBN: 978-1-4244-9140-7. DOI: 10.1109/FG.2011.5771462. URL: http://ieeexplore.ieee.org/document/5771462/ (acedido em 12/07/2022).

[39] Nomidl. *Difference between Sigmoid and Softmax activation function?* en-US. Abr. de 2022. (Acedido em 06/07/2022).

[40] *Optimizers in Deep Learning. What is an optimizer? | by Musstafa | MLearning.ai | Medium*. URL: https://medium.com/mlearning-ai/optimizers-in-deep-learning-7bf81fed78a0 (acedido em 06/07/2022).

[41] *Papers with Code - CK+ Dataset*. en. URL: https://paperswithcode.com/dataset/ck (acedido em 05/07/2022).

[42] *Papers with Code - JAFFE Dataset*. en. URL: https://paperswithcode.com/dataset/jaffe (acedido em 05/07/2022).

[43] *Quick Search - #1 social media search engine - Talkwalker*. URL: https://www.talkwalker.com/quick-search (acedido em 05/07/2022).

[44] *RASPBERRY PI Placa de desenvolvimento Raspberry Pi CM4 IO Board*. pt. URL: https://mauser.pt/catalog/product_info.php?products_id=096-9234 (acedido em 04/07/2022).

[45] *Reconhecimento Facial | Microsoft Azure*. pt. URL: https://azure.microsoft.com/pt-pt/services/cognitive-services/face/ (acedido em 21/03/2022).

[46] *result_1.png (350×239)*. URL: https://algotech.netlify.app/img/image-colorization/result_1.png (acedido em 06/07/2022).

[47] *Rotulagem humana | AutoML Vision*. pt-BR. URL: https://cloud.google.com/vision/automl/docs/human-labeling?hl=pt-br (acedido em 31/03/2022).

[48] David Sandberg. *Face Recognition using Tensorflow*. original-date: 2016-02-12T22:14:40Z. Jul. de 2022. URL: https://github.com/davidsandberg/facenet (acedido em 07/07/2022).

[49] Florian Schroff, Dmitry Kalenichenko e James Philbin. «FaceNet: A unified embedding for face recognition and clustering». en. Em: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Boston, MA, USA: IEEE, jun. de 2015, pp. 815–823. ISBN: 978-1-4673-6964-0. DOI: 10.1109/CVPR.2015.7298682. URL: http://ieeexplore.ieee.org/document/7298682/ (acedido em 13/07/2022).

[50] *See How Retirees Spend Their Newfound Free Time*. en. URL: https://money.usnews.com/money/retirement/slideshows/12-ways-retirees-spend-their-newfound-free-time (acedido em 18/05/2022).

[51] *Sentiment Analysis Guide*. en. URL: https://monkeylearn.com/sentiment-analysis/ (acedido em 21/03/2022).

[52] Sefik Ilkin Serengil. *deepface: A Lightweight Face Recognition and Facial Attribute Analysis Framework (Age, Gender, Emotion, Race) for Python*. URL: https://github.com/serengil/deepface (acedido em 22/03/2022).

[53] Atul Singh. *How to implement Face Recognition using VGG Face in Python 3.8 and Tensorflow 2.0*. en. Mai. de 2020. URL: https://medium.com/analytics-vidhya/how-to-implement-face-recognition-using-vgg-face-in-python-3-8-and-tensorflow-2-0-a0593f8c95c3 (acedido em 07/07/2022).

[54] *Solutions | Consumer Intelligence*. en-US. URL: https://www.brandwatch.com/suite/consumer-intelligence/ (acedido em 06/07/2022).

[55] *Standard Arduino Comparison Guide - SparkFun Electronics*. URL: https://www.sparkfun.com/standard_arduino_comparison_guide (acedido em 04/07/2022).

[56] Yaniv Taigman et al. «DeepFace: Closing the Gap to Human-Level Performance in Face Verification». en. Em: *2014 IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, OH, USA: IEEE, jun. de 2014, pp. 1701–1708. ISBN: 978-1-4799-5118-5. DOI: 10.1109/CVPR.2014.220. URL: https://ieeexplore.ieee.org/document/6909616 (acedido em 22/03/2022).

[57] Keras Team. *Keras documentation: EarlyStopping*. en. URL: https://keras.io/api/callbacks/early_stopping/ (acedido em 06/07/2022).

[58] TechVidvan Team. *Advantages and Disadvantages of TensorFlow*. en-US. Ago. de 2020. URL: https://techvidvan.com/tutorials/pros-and-cons-of-tensorflow/ (acedido em 12/07/2022).

[59] *TensorFlow*. URL: https://www.tensorflow.org/ (acedido em 31/03/2022).

[60] *TensorFlow*. URL: https://www.tensorflow.org/ (acedido em 04/07/2022).

[61] *Text Analytics*. en. URL: https://www.meaningcloud.com/solutions/text-analytics (acedido em 06/07/2022).

[62] *Text Mining Software*. en-US. URL: https://rapidminer.com/solutions/text-mining/ (acedido em 06/07/2022).

[63] *The Facial Action Coding System*. en. Jan. de 2022. URL: https://www.eiagroup.com/the-facial-action-coding-system/ (acedido em 21/03/2022).

[64] *Top Machine Learning Algorithms Explained: How Do They Work?* en. Section: Machine Learning. Ago. de 2020. URL: https://monkeylearn.com/blog/machine-learning-algorithms/ (acedido em 21/03/2022).

[65] *Transferência de aprendizado e ajuste fino | TensorFlow Core*. pt-BR-x-mtfrom-en. URL: https://www.tensorflow.org/tutorials/images/transfer_learning?hl=pt-br (acedido em 30/03/2022).

[66] *Twiggle*. en-US. URL: http://www.twiggle.com/ (acedido em 06/07/2022).

[67] *Understand Loss Functions for Classification Task | Data Science and Machine Learning*. en. URL: https://www.kaggle.com/getting-started/a (acedido em 07/07/2022).

[68] Ben Virdee-Chapman. *Kairos Facial Recognition Demos*. en-gb. URL: https://www.kairos.com/demos (acedido em 05/07/2022).

[69] *Webcam c/ Microfone Imilab - Full HD 90º*. pt. URL: https://www.kontrolsat.com/pt/gaming/perifericos/webcams/webcam-imilab-full-hd?utm_source=kuantokusta&utm_medium=cpc&utm_campaign=catalogo (acedido em 10/07/2022).

[70] *What is MonkeyLearn?* en. URL: http://help.monkeylearn.com/en/articles/2174206-what-is-monkeylearn (acedido em 21/03/2022).

[71] *YouTube Faces DB*. pt. URL: https://docs.google.com/forms/d/e/1FAIpQLScV6lOMO61pkr_LNbW2hNW8pQyGiPAZ9pidOEPEupRFcXon7w/viewform?embedded=true&usp=embed_facebook (acedido em 22/03/2022).

APÊNDICES

A

APÊNDICE A

Serve o presente apêndice para descrever e acrescentar informação relativa à descrição da base de dados da plataforma SmartEmotion referida no Secção 5.3. A base de dados é onde se encontra toda a informação da plataforma, permitindo a persistência e consulta de dados a qualquer momento em qualquer lugar com acesso à internet. Com o auxílio da Figura 75 serão descritas algumas tabelas e relações consideradas importantes.

Com base na Figura 75 realça-se as seguintes tabelas:

clients – Um registo da tabela corresponde a uma instância do sistema SmartEmotion, uma conta de cliente permite a autenticação de um ou mais sistemas Elder Home e acesso a uma conta de cuidador.**users** – Esta tabela é dona de uma difusão com as tabelas: 'clients' e 'administrators'. Guarda todos os dados comuns entre as tabelas da sua difusão. **iterations** – Um registo de uma iteração pertence a um cliente; Um registo de uma iteração contém vários *frames*; Esta tabela guarda e relaciona os registos da deteção de expressões faciais com o cliente. **frames** – Um *frame* apenas pertence a uma iteração; Um *frame* pode ter várias classificações; Guarda informação relativa à deteção de expressões faciais; **notifications** – Um registo desta tabela guarda informação relativa à deteção continua de determinada emoção configurada pelo cliente/cuidador.

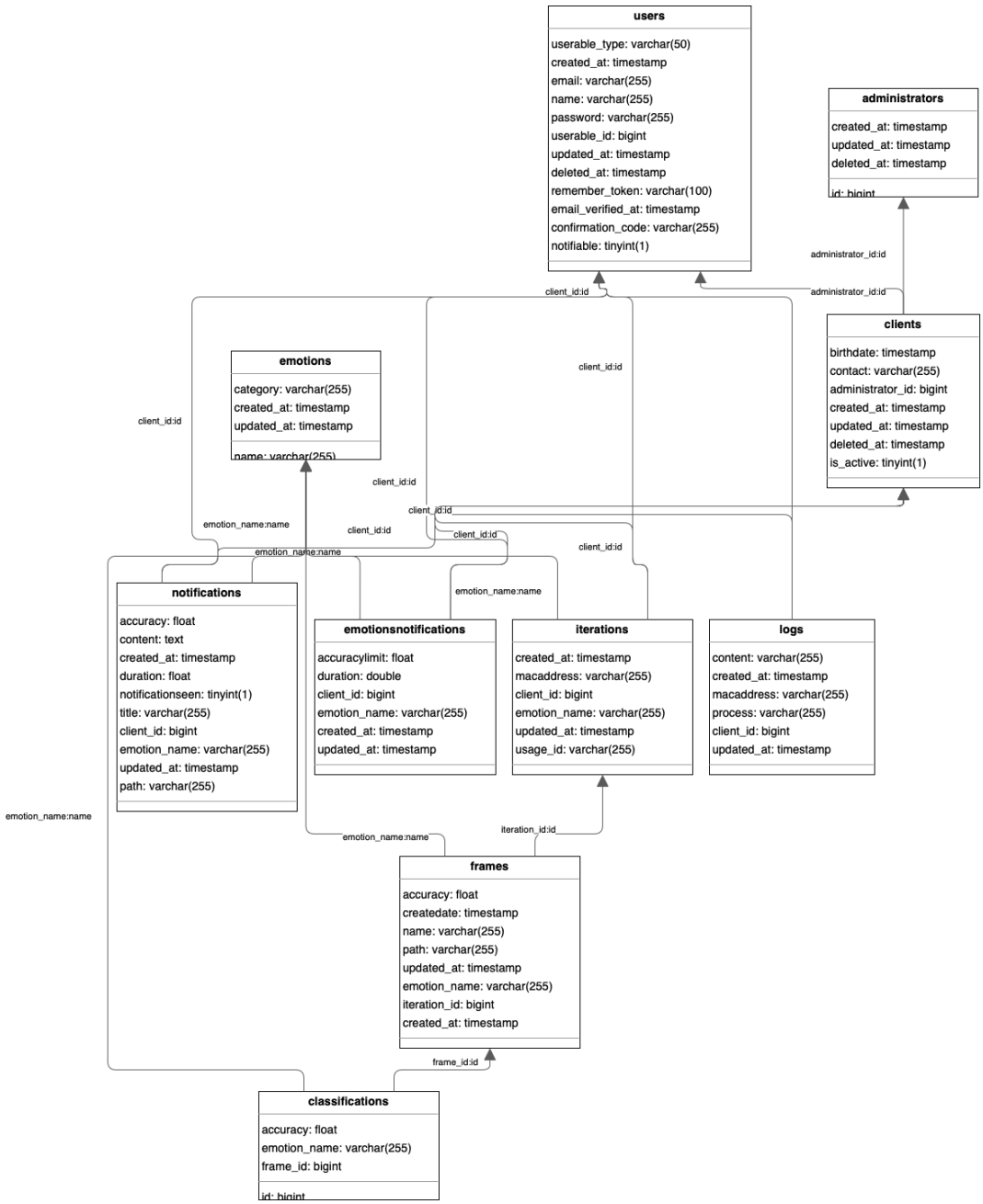


Figura 75: Estrutura relevante da base de dados na Plataforma SmartEmotion.

DECLARAÇÃO

Declaramos, sob compromisso de honra, que o trabalho, denominado de *"Face Detection and Emotion Recognition 4 AAL"*, apresentado neste relatório é original e da autoria dos estudantes Carla Sofia Crespo Mendes (2191170) e Rafael Mendes Pereira (2191266) sob orientação de Professor Doutor António Manuel de Jesus Pereira (apereira@ipleiria.pt), Professor Doutor Roberto Aguiar Ribeiro (roberto.ribeiro@ipleiria.pt) e Professor Doutor José Carlos Bregieiro Ribeiro (jose.ribeiro@ipleiria.pt).

Leiria, Julho de 2022

Carla Sofia Crespo Mendes

Rafael Mendes Pereira