UNIVERSIDADE D
COIMBRA

Pedro Carvalho Gerardo

Expressão e Reconhecimento de emoções para crianças autistas

Coimbra

Dissertação no âmbito do Mestrado Integrado em Engenharia Eletrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra orientada pelo professor Doutor Paulo Jorge Carvalho Menezes

Setembro de 2019

Expressão e Reconhecimento de Emoções para Crianças Autistas

Orientador:

Prof. Dr. Paulo Jorge Carvalho Menezes

Júri:

Prof. Dr. Jorge Manuel Moreira de Campos Pereira Batista

Prof. Dr. Paulo José Monteiro Peixoto

Dissertação no âmbito do Mestrado Integrado em Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Coimbra, Setembro de 2019

Agradecimentos

Este trabalho não teria sido realizado sem o apoio e inspiração de um certo número de pessoas. O meu obrigado a todos por terem feito parte desta fase da minha vida e tornarem a presente dissertação possível.

Em primeiro lugar gostaria de agradecer ao meu orientador Prof. Paulo Menezes por ter acreditado nas minhas capacidades e por todo o apoio que foi dado ao longo deste processo.

Um especial obrigado aos meus pais, pois sem eles nada teria sido possível. Sempre me encorajaram a explorar novas direções e a procurar o meu próprio destino. Ajudaram-me a ultrapassar todos os obstáculos e a concluir esta etapa.

Á Sofia por ter estado sempre presente em todas os momentos, pela ajuda e carinho demonstrado e acima de tudo pelo apoio que me deu ao longo destes meses.

A todos os meus colegas do Instituto de Sistemas de Robótica pelo excelente ambiente de trabalho e acima de tudo pela amizade que foi desenvolvida no processo. Com a sua ajuda foi possível discutir novas ideias e partilhar conhecimento.

Por fim, um grande obrigado a todos os meus amigos mais chegados pela sua amizade incondicional ao longo de todos estes anos, pelos momentos que partilhámos e por terem tornado esta jornada inesquecível.

Resumo

Autismo é um distúrbio neurológico caracterizado pelo comprometimento da interação social, comunicação verbal e não-verbal e comportamento restritivo e repetitivo. Muitas vezes, é difícil para indivíduos com autismo interpretar e até expressar emoções básicas como felicidade ou tristeza. Além disso, manter o contacto visual com outra pessoa pode tornar-se uma tarefa árdua. É extremamente difícil analisar o olhar, tornando-se penoso interpretar o que a outra pessoa está a tentar expressar. Embora tenham algumas carências, é bem conhecido que pessoas com autismo podem aprender e superar algumas dessas ambiguidades. Na presente dissertação foi desenvolvido um ambiente que pode ser usado para ensinar estes indivíduos a expressar emoções básicas. Para tal, foi desenvolvidos um conjunto de jogos sérios, onde um sistema de reconhecimento automático de expressões faciais (RAEF) é usado. Este trabalho é marcado por uma pesquisa detalhada sobre os conceitos e metodologias existentes por trás dos sistemas RAEF, bem como uma avaliação da sua eficácia. Os modelos desenvolvidos foram testados, a fim de escolher o mais adequado para o reconhecimento de expressões faciais. Aqui, foi explorado o valor da aprendizagem profunda, focando nos recentes avanços tecnológicos, particularmente com Redes Neuronais Convolucionais (RNC). Etapas incrementais foram realizadas de forma a implementar a melhor solução para a arquitectura da rede.

Palavras Chave: Autismo, Reconhecimento Automático de Expressões Faciais, Jogos Sérios, Redes Neuronais Convolucionais

Abstract

Autism is a neurodevelopmental disorder characterized by impaired social interaction, impaired verbal and non-verbal communication, and restricted and repetitive behavior. It is often difficult for autistic individuals to interpret and even express human basic emotions like happiness and sadness. In addition, maintain gaze interaction with another person is not an easy task for autistic patients. They often find it extremely difficult to interpret a person's gaze, making it hard to follow it and interpret what the other person is trying to point out. Although they have these impairments, it is well known that people with autism can learn and overcome to some degree these ambiguities. The proposed work focused on developing an environment, which can be used to teach these individuals how to express basic emotions. In order to achieve this a series of serious games were created where an Automatic Facial Expression Recognition (AFER) system is used. This work is marked by a detailed research about the concepts and the existent methodologies behind the AFER systems, as well as an evaluation of their effectiveness. Additionally, relevant models were tested, in order to choose the most adequate for facial expressions recognition. Here, we explored the value of Deep Learning by focusing on recent technological breakthroughs, particularly with Convolutional Neural Networks (CNN). Incremental steps were made in order to deploy the better solution to the network architecture.

Keywords: Autism, Automatic Facial Expression Recognition, Serious Games, Convolutional Neural Networks

Conteúdo

Lista de Acrónimos

AFER	Automatic Facial Expression Recognition
AU	Action Unit
CNN	Convolutional Neural Network
FACS	Facial Action Coding System
FER	Facial Expression Recognition
GDE	Gradiente Descendente Estocástico
HMM	Hidden Markov Model
ML-HMM	Multi-Level Hidden Markov Model
NB	Naive Bayes
RAEF	Reconhecimento Automático de Expressões Faciais
RNA	Redes Neuronais Artificiais
RNC	Redes Neuronais Convolucionais
SVM	Support Vector Machine
SSS	Stochastic Structure Search
TAN	Tree Augmented Naive Bayes

Lista de Figuras

Lista de Tabelas

1 Introdução

O reconhecimento de expressões faciais tem sido uma área sujeita a intensa pesquisa nos últimos 10 anos, com extensas áreas de aplicação, incluindo animação de avatares, neuromarketing e robôs sociáveis. Não é um problema simples, mesmo para métodos de aprendizagem de máquina, uma vez que as pessoas podem variar significativamente a maneira de exibir as suas expressões devido aos muitos fatores que podem afetá-la. Estado de saúde, cansaço, contexto social, entre muitos outros, podem modular a ativação muscular e deformação tecidual e que se traduzirão numa variabilidade da forma da face para uma dada emoção. Naturalmente imagens de um rosto capturadas em diferentes circunstâncias exibirão variações, mas também serão influenciadas por outros fatores externos, como iluminação, fundo e posicionamento em relação à câmera [27]. No entanto, os seres humanos são particularmente bons em reconhecer expressões faciais, independentemente dos fatores acima descritos. Tal facto pode servir como uma motivação para explorar redes neuronais artificiais com o mesmo propósito. Emoções estão presentes no quotidiano humano e influenciam a nossa perceção ou relação a estímulos externos, mas também têm um papel muito importante na comunicação entre as pessoas. O reconhecimento das emoções entre pessoas permite ao indivíduo entender o que não foi dito, contextualizar o que foi dito e até mesmo responder ou comportar-se de acordo. Essa capacidade pode ser vista como um tipo de ajuda preditiva para inferir as ações de outra pessoa. Por outro lado, a pessoa que expressa alguma emoção também espera que o seu interlocutor reconheça e responda de acordo, caso contrário, a empatia é quebrada e a comunicação deteriora-se.

Ao longo dos anos, este tópico recebeu perceções notáveis, mas o trabalho de Darwin em 1872 foi definitivamente o contributo que estabeleceu as regras base [13]. A sua pesquisa, em conjunto com outras descobertas contribuíram para a criação de um método científico de categorização de expressões, que tem sido amplamente explorado e considerado um passo importante no campo do reconhecimento automático de expressões faciais.

Outra grande contribuição para o estudo das expressões faciais deve-se ao trabalho de

Paul Ekman e os seus colegas [16]. Desde 1970, este grupo desenvolveu ferramentas para a interpretação de expressões faciais, nomeadamente o mapeamento dos músculos mensuráveis e o espaço emocional, encontrando estados discretos ou protótipos de emoções.

1.1 Contexto

As contribuições mencionadas anteriormente serviram de motivação para o estudo e interpretação de expressões faciais. Estas são deveras importantes no nosso dia-a-dia pois são um indicador do estado emocional, no entanto existem casos, como o de indivíduos autistas, em que a capacidade de se expressar e reconhecer expressões facias é comprometida devido ao distúrbio que apresentam. Autismo é um distúrbio neurológico caracterizado pelo comprometimento da interação social, comunicação verbal e não verbal, comportamento restrito e repetitivo. Os sintomas manifestam-se na infância, geralmente antes dos três anos de idade. Muitas vezes é difícil para estes indivíduos interpretar, avaliar e até mesmo expressar emoções básicas como felicidade ou tristeza. Manter contacto visual com outra pessoa pode tornar-se uma tarefa bastante complexa para pacientes com autismo e por vezes é extremamente difícil interpretar o olhar de uma pessoa, tornando-se penoso segui-la e analisar o que está a sentir ou expressar. Embora com estas carências, é sabido que pessoas com autismo podem aprender e superar essas ambiguidades.

1.2 Objetivos

A presente dissertação foca-se no desenvolvimento de um ambiente que possa ser usado para ensinar indivíduos autistas a reconhecer e expressar emoções básicas. Para isso, pretende-se criar um conjunto de jogos terapêuticos capazes de ajudar os pacientes a ultrapassar tais limitações. Usando um sistema automático de reconhecimento de expressões faciais em complementaridade com os jogos, espera-se que o utilizador possa aprender a reconhecer e expressar com precisão expressões faciais.

2 Estado da Arte

Emoções são a essência que nos torna humanos. Têm um impacto direto na nossa rotina diária, interação social, atenção, perceção e memória. O nosso rosto é um dos grandes indicadores. Sempre que rimos ou choramos manifestamos as nossas emoções permitindo, assim, que outras pessoas compreendam o que se passa na nossa mente.

A nossa face é um sistema bastante complexo, composto por 40 músculos autónomos completamente estruturais e funcionais, cada um dos quais pode ser acionado independentemente de outros. O sistema muscular facial é o único lugar no nosso corpo onde os músculos estão ligados a um tecido ósseo e facial, ou apenas a tecido facial (outros músculos no corpo humano estão ligados a dois ossos) [6].

A análise de expressões faciais desempenha um papel fundamental em várias áreas, como na psicologia clínica, na deteção de mentiras, na avaliação de dor e em muitos outros campos em que o conhecimento depende diretamente da informação extraída da face humana.

Existiram várias tentativas de categorizar as expressões humanas, Paul Ekman e os seus colegas provaram que, pelo menos para as seis expressões básicas, é possível alcançar um consenso multicultural. Felicidade, tristeza, raiva, surpresa, aversão e medo, segundo os seus estudos, podem ser discrimináveis em qualquer cultura letrada [17].

Os atuais desenvolvimentos da computação afetiva têm essas seis emoções básicas como meta final de reconhecimento, no entanto, existe uma outra expressão que deve ser considerada, uma vez que foi encontrada em mais de 75% das culturas ocidentais e não-ocidentais - desprezo [18].

De acordo com alguns estudos [11], as expressões podem ser geridas independentemente de outras categorias de análise facial, como a idade, sexo ou contexto de identidade. Desta forma, este trabalho segue uma estrutura baseada em expressões, embora as outras categorias, mencionadas acima, sejam usadas para fornecer diversidade aos conjuntos de dados usados na análise de expressões faciais.

Figura 2.1: Estrutura básica de um sistema de análise de expressões faciais

2.1 Emoção e Expressão

Em termos neurobiológicos, as emoções podem ser definidas como reações desencadeadas pela presença de certos estímulos externos ou internos [6]. Podem conter os seguintes elementos:

- **Sintomas corporais**, como o aumento do batimento cardíaco. Estes sintomas, são essencialmente inconscientes e involuntários.
- **Tendência para agir**, como fugir de uma situação de perigo ou preparar-se fisicamente para o ataque de um adversário.
- **Expressões faciais**
- **Avaliações cognitivas** de eventos, estímulos ou objetos.

Alguns cientistas acreditam que os seres humanos estão permanentemente num estado emocional, no entanto, essas emoções são praticamente impercetíveis, de tal forma que podemos considerá-las inexistentes.

Em 1978, com base no trabalho do anatomista sueco Carl-Herman Hjortsjö, Paul Ekman e Wallace Friesen desenvolveram um método para codificar o comportamento facial. A sua abordagem, denominada *Facial Action Coding System* (FACS) representa um sistema padronizado de classificação de expressões faciais com base em características anatômicas. Descrevem qualquer ocorrência de expressões faciais como combinações de componentes elementares chamadas de unidades de ação (*Action Units* - AUs). Com a codificação das unidades de acção facial conseguimos distinguir 3 categorias de expressões faciais:

- **Macroexpressões** que ocorrem nas nossas interações diárias e geralmente são óbvias a olho nu. Têm uma duração de 0.5-4 segundos.

- **Microexpressões** ocorrem quando tentamos, consciente ou inconscientemente, esconder ou reprimir o nosso estado emocional. Estas são mais difíceis de ser detetadas e tem uma duração mais curta que as anteriores, menos de meio segundo.

- **Expressões subtis** que estão associadas à intensidade da emoção subjacente. Expressões subtis indicam o início de uma expressão facial onde a intensidade da expressão ainda é baixa.

Com raras exceções, todas as pessoas têm os mesmos músculos faciais. As unidades de ação presentes no FACS são baseadas no que os músculos faciais permitem fazer. Para determinar as mudanças de expressão associadas a cada músculo, Ekman e Friesen começaram por estimular eletricamente os músculos de indivíduos aprendendo a controlar tais movimentos voluntariamente. Após vários testes aperceberam-se que cada unidade de ação está associada a um ou mais músculos faciais.

O sistema de codificação está neste momento dividido em 3 grandes áreas, as unidades de acção principal, composto por 46 AUs, as unidades de acção do movimento da cabeça, caracterizadas por 8 AUs e por fim as unidade de acção relativas ao movimento dos olhos, composto por 4 AUs.

2.2 Expressões Faciais

De seguida serão descritas as sete expressões faciais, que permitem alcançar um consenso multicultural, segundo o ponto de vista de Paul Ekman [16].

Felicidade

Segundo Ekman, a palavra *happiness* bem como *enjoyment* não são específicas o suficiente, ambas podem exprimir muitas emoções agradáveis como diversão, *fiero* (satisfação pessoal de realizar uma tarefa difícil), *naches* (uma sensação de prazer e orgulho), contentamento, excitação, prazeres sensoriais, alívio, admiração, *schadenfreude* (sentir-se melhor perante o infortúnio dos outros), êxtase e gratidão. Além da sua singularidade, todas as emoções mencionadas acima envolvem sorrisos, que podem ser considerados um sinal controverso de felicidade. Embora o sorriso possa ser falso, é conhecido um método que permite determinar se o sorriso é natural, ou *Duchenne* - nome atribuido por Ekman em memória do neurologista francês, Duchenne de Boulogne. Após analisar fotografias onde o músculo zigomático principal é estimulado, Duchenne afirmou que a emoção de alegria é expressa

pela contração combinada do músculo zignomático (responsável pelo levantamento dos cantos da boca) e do músculo orbicular (músculo ao redor dos olhos). A veracidade do sorriso é revelada sabendo que o músculo ao redor dos olhos não obedece à vontade.

Figura 2.2: Sorriso Duchenne [16]

Tristeza

A tristeza é uma das emoções mais duradouras e, como qualquer outra emoção, não pode ser completamente descrita com palavras. Está associada a sentimentos de desapontamento, desânimo, depressão, aflição e miséria. Para ser completamente compreendida é necessário ser vivenciada e pode ser revivida através de outras expressões.

Podemos iludir o nosso cérebro, fazendo certos movimentos faciais, desencadeando algumas mudanças fisiológicas, como forçar o sentimento de tristeza (ou outra emoção), porém este é apenas um exercício de memória, o verdadeiro sentimento tem de ser vivenciado antes.

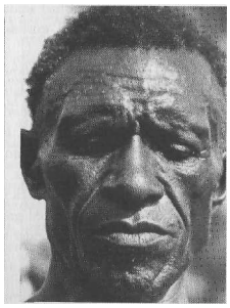


Figura 2.3: Expressão de Tristeza [16]

Raiva

A raiva pode ser desencadeada como uma reação a um desacordo, um desafio, um insulto ou a uma pequena frustração. Pertence aos sentimentos negativos gerados pelo que a pessoa infligida interpretou como ofensivo.

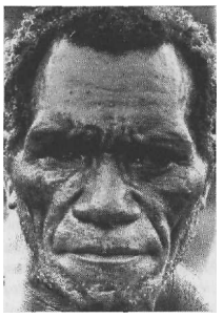


Figura 2.4: Expressão de Raiva [16]

Surpresa

A surpresa é uma reação a um evento inesperado. É a mais breve de todas as emoções, geralmente dura enquanto a pessoa toma consciência da situação. Esta emoção é rapidamente seguida de outra emoção que, por sua vez, depende da situação.

As características que permitem definir esta emoção são habitualmente confundidas com as características que caracterizam medo, apresentando pequenas diferenças entre si.

Figura 2.5: Expressão de Surpresa [16]

Medo

O medo é a emoção mais pesquisada, devido à sua ampla existência em quase todos os animais. É desencadeada quando existe uma ameaça que possa colocar em causa a integridade física ou mental do indivíduo.

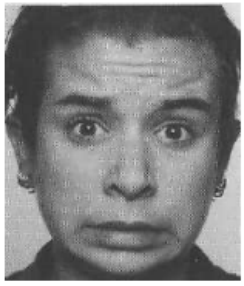


Figura 2.6: Expressão de Medo [16]

Aversão

Os estudos conduzidos por Ekman têm em conta as ideias do psicólogo Paul Rozin, sugerindo a existência de dois tipos de aversão. O primeiro, que envolve uma componente oral, é considerada ofensiva e contaminante. O segundo caso é desencadeado pelo sentimento de estranheza, doença, infelicidade ou outra ação eticamente incorreta.

Figura 2.7: Expressão de Aversão [16]

Desprezo

O desprezo é muitas vezes confundido com aversão, é uma expressão exclusiva da interação humana que representa um julgamento moral e traduz um sentimento de superioridade.

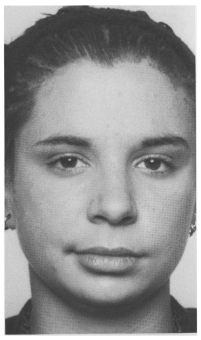


Figura 2.8: Expressão de Desprezo [16]

2.3 Unidades de Ação Facial

As pessoas podem ter diferentes intensidades emocionais, mas os movimentos base são semelhantes entre si. O conjunto dos seguintes movimentos deve ser visto como um todo, pois alguns pontos da expressão, se examinados separadamente, podem induzir erro na emoção final. O quadro seguinte dita quais as unidades de ação facial necessárias para caracterizar cada uma das emoções [6] descritas na secção anterior.

Emoção	**Unidades de Ação Facial**	**Descrição**
Felicidade	6 + 12	Levantamento da bochecha Levantamento do canto do lábio
Tristeza	1 + 4 + 15	Levantamento sobrancelha interna Depressão da sobrancelha Depressão do canto do lábio
Surpresa	1 + 2 + 5 + 26	Levantamento sobrancelha interna Levantamento da sobrancelha externa Levantamento da pálpebra superior Queixo caído
Medo	1 + 2 + 4 + 5 + 7 + 20 + 26	Levantamento sobrancelha interna Levantamento sobrancelha externa Depressão da sobrancelha Levantamento sobrancelha superior Compressão da sobrancelha Extensão do lábio Queixo caído
Fúria	4 + 5 + 7 + 23	Depressão das sobrancelhas Levantamento da pálpebra superior Compressão da sobrancelha Compressão do lábio
Aversão	9 + 15 + 16	Nariz enrugado Depressão do canto do lábio Depressão do lábio inferior
Desprezo	12 + 14	Levantamento do canto do lábio Covinha na bochecha

Tabela 2.1: Unidades de ação e respetivas emoções

2.4 Aquisição da face

A aquisição do rosto pode envolver vários processos, como a deteção da face, segmentação ou mesmo normalização geométrica. De modo a manipular e melhorar a aquisição de face nos sistemas de reconhecimento automático de expressões faciais é necessário ter em atenção dois pontos, perceção e reconhecimento (Descritos na secção 2.4.1).

A deteção de faces é uma das áreas mais exploradas em sistemas de reconhecimento facial. O quadro seguinte apresenta alguns dos algoritmos mais utilizados de deteção de faces.

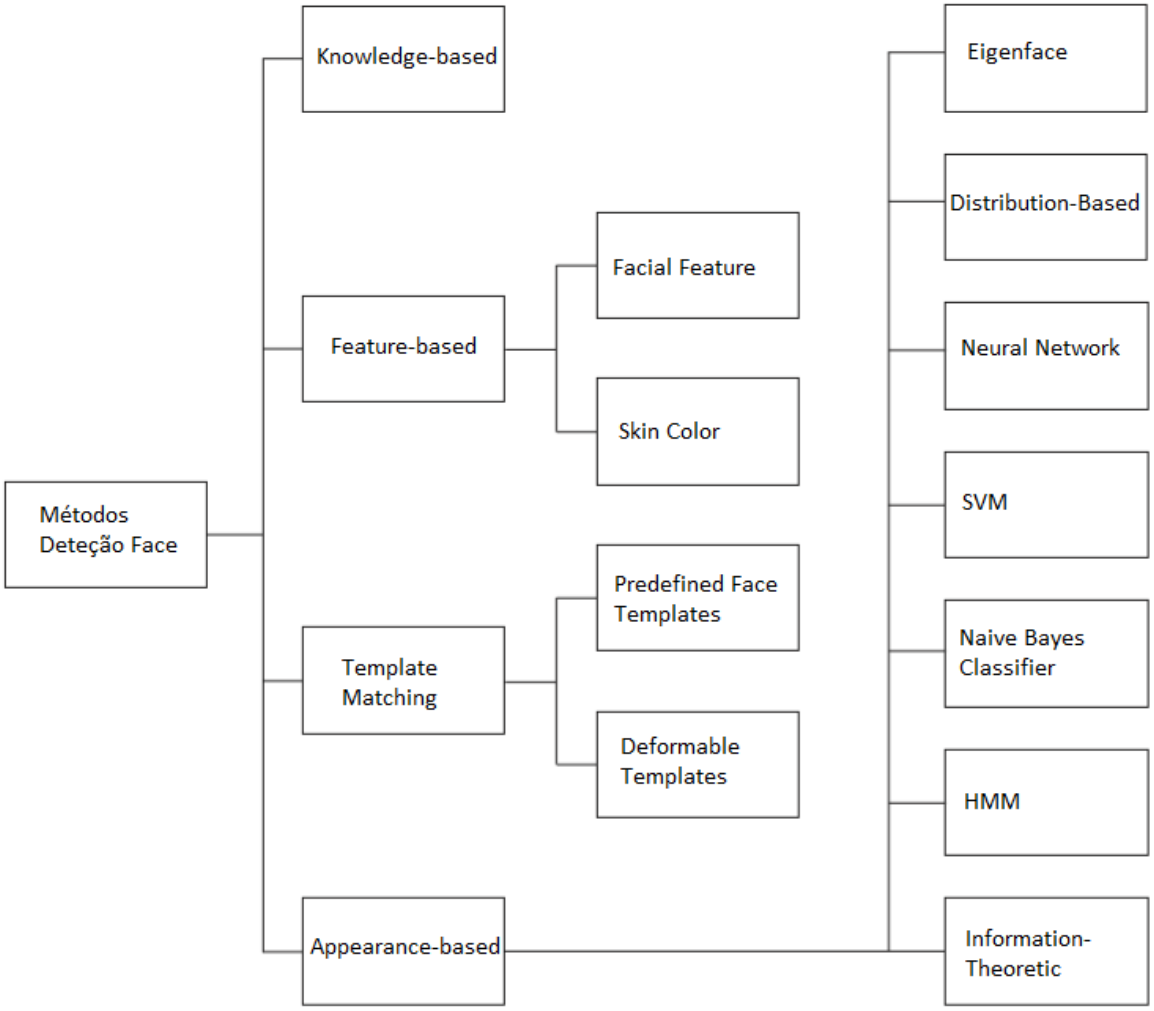


Figura 2.9: Métodos de deteção de faces

2.4.1 Perceção e Reconhecimento

De forma a compreender como é feita a aquisição da face humana é necessário distinguir dois conceitos chave, a Perceção Humana e como é feito o Reconhecimento. A Perceção e o Reconhecimento nem sempre foram vistos como conceitos separados sendo que os primeiros passos que levaram à sua separação foram dados no século XIX [7].

A Perceção é um processo que funciona como reação a um estímulo inicial, processa as características e configurações de uma imagem visual. Funciona como um conjunto de dados para a interpretação humana. Este tópico pode seguir diversos canais, como visual, auditivo, olfativo, háptico e gustativo. Neste caso estamos interessados na perceção visual.

O processo em si é mais complexo do que qualquer definição que possa ser dada. De acordo com alguns neurocientistas, o processo de perceção visual estimula metade do córtex humano [31] e parte dessa atividade depende do significado atribuído a esse estímulo.

A perspetiva tradicional dita que quando olhamos para um objeto conseguimos extrair pequenos excertos de informação específica, como a sua localização, forma, textura, tamanho e, para objetos conhecidos ou familiares, nomes. Neste campo não devemos esperar um consenso sobre como esta informação é adquirida, pois pode ser influenciada por experiências passadas.

A abordagem clássica à definição de perceção define que a partir de um objeto físico ou evento do mundo exterior capaz de refletir luz ou energia, o chamado estímulo distal (*distal stimulus*), o sistema visual recebe e regista as informações obtidas, desenvolve um estímulo proximal (*proximal stimulus*), e a partir daí a retina cria uma imagem. Esta imagem é exibida de cabeça para baixo e invertida, aguardando uma interpretação e posterior reconhecimento. O reconhecimento é visto como o objetivo da Perceção.

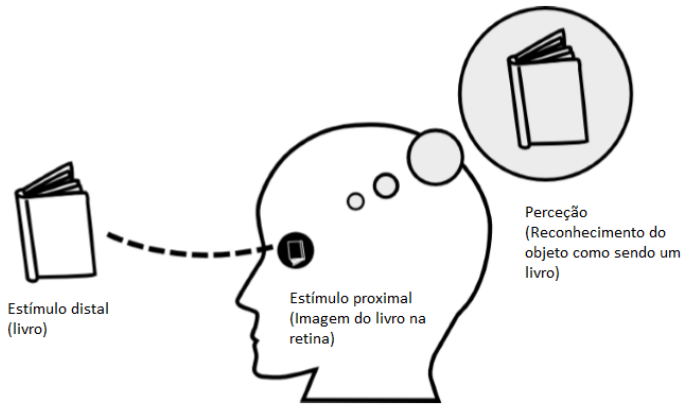


Figura 2.10: Estímulo distal, estímulo proximal e perceção

Embora não exista consenso em relação à abordagem usada para o estudo da perceção, existe consenso relativamente ao estímulo proximal e sobre as suas fases distintas, assim como os excertos de informação alcançam a perceção, o que é comumente descrito como modelos *Bottom-Up* e *Top-Down*. Estes modelos são estratégias de processamento de informação e organização de conhecimento.

Relativamente aos modelos **Top-Down** (*Knowledge-based*), os métodos de detecção da face são desenvolvidos com base em regras derivadas do conhecimento de especialistas sobre o rosto humano. Foi necessário criar regras para descrever o rosto e as suas ligações. Por exemplo, uma cara geralmente aparece numa imagem com dois olhos simétricos, um nariz e uma boca. As relações entre certas características podem ser representadas pelas distâncias e posições relativas. Um problema associado a esta abordagem é a dificuldade em traduzir o conhecimento humano em regras bem definidas. As regras sendo bem detalhadas podem falhar na deteção de caras que não passem por todas os parâmetros. Se as regras são muito gerais, podem detetar muitos falsos positivos [42].

Os métodos **Bottom-Up** (*Feature-based*) são utilizados essencialmente para a localização de faces e visam encontrar características invariantes no rosto, estas características existem mesmo quando ocorrem mudanças de ambiente. O pressuposto subjacente baseia-se na observação de que os seres humanos podem facilmente detetar rostos e objetos em diferentes poses e condições de iluminação e, portanto, devem existir certas características que são invariáveis ao ambiente. Numerosos métodos foram propostos para detetar características faciais e inferir a presença de um rosto. Características faciais como sobrancelhas, olhos, nariz, boca e linha de cabelo são frequentemente extraídas por detetores de borda (*edge detectors*). Com base nas características extraídas, um modelo estocástico é construído para descrever as suas relações e verificar a existência de uma face [42]. Um problema destes algoritmos baseados em características é que os recursos da imagem podem ser corrompidos de maneira geral devido à iluminação, ruído e oclusão. Os principais ramos deste modelo são os métodos de correspondência (*Template Matching*) que utilizam técnicas de processamento de imagens de modo a encontrar certos traços característicos que correspondem a uma imagem modelo. Assim, é possível obter uma correlação entre imagens de modo a encontrar contornos faciais, olhos, boca, nariz, etc.

Existem ainda modelos baseados em aparência (*Appearance-Based*), que incluem um amplo espetro de métodos. Nestes encontramos as redes neuronais. Estas redes replicam uma analogia entre os neurónios biológicos humanos. O conceito de "pesos"introduzidos computacionalmente e a maneira como mudam dependendo da entrada, imitam a dinâmica

real e as transmissões de informação reproduzidas pelo cérebro humano.

2.4.2 Método de Deteção de Faces

De modo a testar o modelo, foi usado o método *Viola-Jones Face Tracker*. Este método foi proposto por Paul Viola e Michael Jones em 2001 e baseia-se numa função em cascata, treinada a partir de um conjunto de imagens positivas e negativas que, posteriormente, é usada para detetar objetos em imagens.

Este modelo tem três contribuições principais. Primeiro é necessário introduzir o conceito de imagem integral, inspirado pelas características Haar (*Haar features*). Cada pixel é igual à soma total de todos os pixeis acima e à esquerda do pixel em questão.

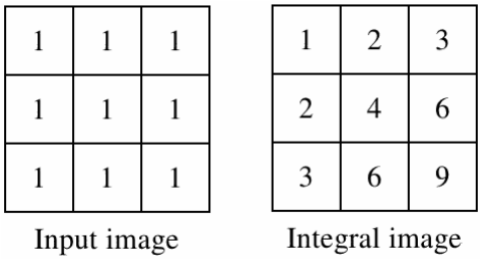


Figura 2.11: Imagem Integral [25]

Com esta região, podemos selecionar os valores específicos para o cálculo da soma de todos os píxeis, permitindo resultados constantes ao longo do tempo. Esses valores são os píxeis na imagem integral que coincidem com os cantos dos retângulos definidos na imagem de entrada.

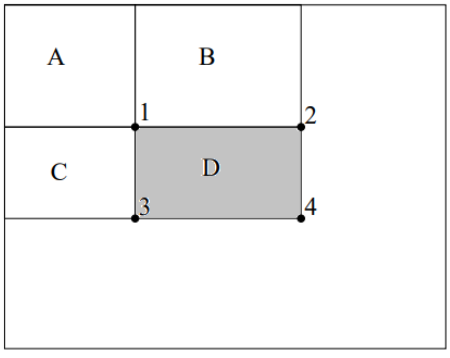


Figura 2.12: O valor da imagem integral na posição 1 é a soma dos pixeis no retângulo A. O valor na posição 2 é A+B, na posição 3 é A+C e na posição 4 é A+B+C+D. A soma dentro de D (zona a cinzento) pode ser calculada por 4+1-(2+3) [25]

O classificador analisa uma determinada sub-janela usando características que consistem em dois ou mais retângulos.

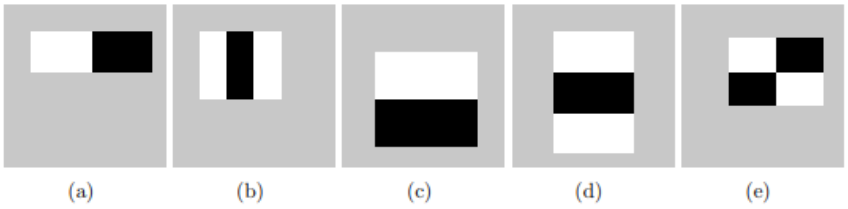


Figura 2.13: Diferentes tipos de características usados pelo classificador [25]

Cada característica resulta num único valor que é calculado subtraindo a soma dos retângulos brancos da soma dos retângulos pretos.

A segunda contribuição é um algoritmo de aprendizagem, baseado em *AdaBoost* que seleciona um pequeno número de características críticas e produz classificadores extremamente eficientes. Este método combina *weak classifiers* restringindo-os a uma característica.

Um *weak classifier*, $h(x, f, p, \theta)$ é definido por:

$$h(x, f, p, \theta) = \begin{cases} 1, pf < p\theta \\ 0, \text{ caso contrário} \end{cases}$$

De seguida é apresentado, em pseudo-código, o método AdaBoost [39]:

- Dado um conjunto de imagens exemplo $(x_1, y_1), ..., (x_n, y_n)$ onde $y_1 = 0, 1$ para negativo e positivo respetivamente

- Inicialização dos pesos $\omega_{1,i} = \frac{1}{2m}, \frac{1}{2l}$ para $y_i = 0, 1$ respetivamente, onde m e l são o numero de negativos e positivos respetivamente.

- Para $t = 1, ..., T$:

  - Normalizar pesos (*weights*),

$$\omega_{t,i} \leftarrow \frac{\omega_{t,i}}{\sum_{j=1}^{n} \omega_{t,j}}$$

  - Selecionar o melhor *weak classifier* tendo em conta o erro dos pesos,

$$\varepsilon_t = \min_{f,p,\theta} \sum_i \omega_i |h(x, f, \theta) - y_i|$$

– Defina $h_t(x) = h(x, f, p, \theta)$ onde f_t, p_t e θ_t são os minimizadores de ε_t

– Atualizar pesos:

$$\omega_{t+1,i} = \omega_{t,i}\beta_t^{1-e_i}$$

onde $e_i = 0$ se x_i for classificado corretamente, $e_i = 1$ caso contrário, e $\beta_t = \frac{\varepsilon_t}{1-\varepsilon}$

- O classificador final é:

$$C(x) = \begin{cases} 1, \sum_{t=1}^{T} \alpha_t \mathrm{h}_t(x) \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0, \text{ caso contrário} \end{cases}$$

onde $\alpha_t = \log \frac{1}{\beta_t}$

Este algoritmo introduz ainda uma terceira contribuição para melhorar a eficiência computacional. O método inclui progressivamente classificadores mais complexos com uma estrutura em cascata, aumentando o número de recursos processados com base em taxas de deteção anteriores.

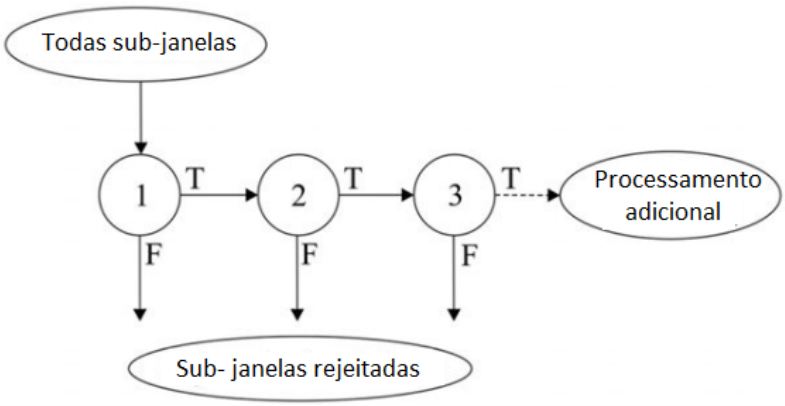


Figura 2.14: Estrutura em cascata [39]

2.4.3 Métodos Tradicionais de Reconhecimento de Expressões Faciais

Segundo a estrutura básica de um sistema AFER esta será a etapa onde será recolhida toda a informação relevante de modo a extrair expressões faciais ou unidades de ação facial. A fase de reconhecimento é fortemente acoplada a decisões temporais, ou seja, os métodos são categorizados segundo a dependência do tempo. Se a base é independente do tempo e usa apenas quadros podemos designar estes métodos por métodos baseados em quadros (*frame-based*), caso contrário assume uma designação baseada em sequência (*sequence-based*), ou seja, a fonte de informação é constituída por uma sequência de quadros e dinâmica temporal.

Para atribuir métodos baseados em frames, são utilizados, maioritariamente, classificadores estáticos, como, Naive Bayes (NB), Support Vector Machine (SVM), Tree Augmented Naive Bayes (TAN), Stochastic Structure Search (SSS).

Para os métodos baseados em sequências são utilizados classificadores dinâmicos, que são amplamente utilizados para classificar expressões, como o Single Hidden Markov Model (HMM) e Multi-Level Hidden Markov Model (ML-HMM).

Método	**Performance (Taxa média)**	**Dataset**
SVM	76.11%	Cohn-Kanade
NB-R	72.50%	Cohn-Kanade
TAN-R	72.90%	Cohn-Kanade
NB-RSR	69.10%	Cohn-Kanade
TAN-RSR	69.30%	Cohn-Kanade
SSS-RSR	74.80%	Cohn-Kanade
HMM	78.49%	L.Shao-Hsien Chen
ML-HMM	82.46%	L.Shao-Hsien Chen
NB+HMM (Híbrido)	73.22%	Cohn-Kanade

Tabela 2.2: Métodos de reconhecimento de expressão facial com rótulos (R), sem rótulos (SR), ou ambos (RSR)

A tabela 2.2 foi construída tendo em conta a informação em [10] e de seguida são descritos os métodos que apresentaram melhores resultados.

- **Support Vector Machine - SVM**

  SVM [30] é um algoritmo de aprendizagem de máquina que pode ser implementado para propósitos de classificação e regressão. Os algoritmos SVM focam-se em encontrar um hiperplano que melhor divida um conjunto de dados em diversas classes. Um hiperplano pode ser explicado como sendo uma linha que separa e classifica linearmente um conjunto de dados, assim sendo, quanto mais longe do hiperplano os pontos de dados se encontrarem, maior será a confiança que os dados foram classificados corretamente. Desta forma, quando são introduzidos novos dados, a classe atribuída será determinada dependendo do lado do hiperplano onde estão inseridos. A distância entre o hiperplano e o ponto de dados mais próximo é conhecido como margem. O objetivo será escolher um hiperplano com a maior margem possível, existindo assim uma maior probabilidade dos novos dados serem corretamente classificados.

- **Multi-Level Hidden Markov Models - HMM**

  HMM [23] têm sido amplamente utilizados para problemas de classificação e modelagem. O processo de aprendizagem é baseado em probabilidades condicionais. A capacidade de modelar sinais ou eventos não estacionários é uma das muitas vantagens dos HMMs. No entanto, existem algumas desvantagens, que na maioria dos casos estão relacionadas com aspetos temporais.

  Relativamente ao reconhecimento de expressões faciais, o sinal pode ser visto como uma medida de movimento facial, ou seja, o reconhecimento de expressões é feito através da descodificação do estado ativo, de acordo com o seu tempo. Por natureza, este sinal é não-estacionário, uma vez que uma expressão pode ser exibida em taxas variáveis, com diferentes intensidades em diferentes indivíduos.

2.5 Aprendizagem Profunda

Durante décadas, a humanidade sonhou em construir máquinas inteligentes que conseguissem imitar as capacidades cognitivas do cérebro humano. No entanto, para construir essas máquinas artificialmente inteligentes, há que resolver alguns dos problemas computacionais mais complexos com os quais já lidámos, e novas abordagens para programá-los têm de ser adotadas. De forma a lidar com tais problemas foram desenvolvidos mecanismos de aprendizagem de máquina (*Machine learning*). Aprendizagem de máquina é uma das subáreas do campo da inteligência artificial e o seu principal objetivo [40] é prever a função de mapeamento $\gamma = f(x)$, onde:

- x são os dados de entrada, podendo ser leituras de sensores, imagens, etc.
- γ é um valor discreto (trata de resolver um problema de classificação), ou um valor contínuo (lida com um problema de regressão).

A função de mapeamento (f) é estimada usando um conjunto de amostras de treino, também conhecido como conjunto de dados de treino e pode ser caracterizado com um conjunto de parâmetros γ. A partir do conjunto de dados de treino x_i, com as suas saídas anotadas y_i, γ é estimada na fase de treino.

A função alvo (f) obtida a partir das amostras de treino nem sempre generaliza bem com novos dados. A generalização refere-se à qualidade dos conceitos aprendidos quando se aplicam novos exemplos que não foram usados na fase de treino. O modelo deve generalizar os dados a partir dos conjunto de treino pois, para a maioria dos casos, o número de entradas possíveis é extenso, e certamente novos exemplos, que não foram usados na fase de treino, serão introduzidos. Isto permite fazer a previsão de novos dados, nunca antes vistos pelo modelo.

O modelo pode falhar na resolução do problema devido a:

- O erro no treino, ou seja, o erro obtido ao validar o modelo no subconjunto de treino, permanece grande, o que significa que o modelo não foi capaz de escolher as características adequadas ao problema e não consegue resolvê-lo de maneira eficiente - *underfitting*

- O erro no treino é pequeno mas existe uma discrepância grande entre os erros de treino e teste. O erro de teste permite validar o modelo com dados que não foram utilizados na fase de treino. É uma métrica que dá uma ideia de como o modelo consegue generalizar os conceitos. Como seria de esperar, na maioria dos casos, o modelo obtém erros no treino mais pequenos do que no teste, no entanto, se esta diferença for significativa, o modelo não está a generalizar bem os dados - *overfitting*

A capacidade de um modelo pode ser vista como a sua aptidão de ajustar os seus parâmetros [21]. Existe a possibilidade de controlar se um modelo tem maior probabilidade de *underfitting* ou *overfitting* alterando a sua capacidade. Um algoritmo de aprendizagem de máquina terá melhor desempenho, quando a sua capacidade for apropriada à complexidade do problema e à quantidade de dados de treino fornecida.

À medida que a dimensionalidade dos dados de treino aumenta, o número de parâmetros, bem como a capacidade de aprendizagem de f também aumenta.

Hoje em dia, grandes conjuntos de dados estão disponíveis para o treino de algoritmos de *machine learning* e observamos que o desempenho de f, relativamente ao treino, melhorou quando a dimensionalidade dos dados aumentou. Isto pode ser explicado pelo facto de conjuntos de dados maiores, terem maior disparidade, e como tal necessitam de capacidades de aprendizagem superiores. No entanto, alguns modelos acabam por saturar quando existem muitos dados devido às suas limitadas capacidades de aprendizagem. Neste caso, acontece *underfitting* e o modelo obtido não representa uma boa solução para o problema em causa.

2.5.1 Redes Neuronais Artificiais

Redes Neuronais Artificiais (*RNAs*) são modelos computacionais de aprendizagem de máquina inspirados em redes neuronais biológicas. Foram mencionadas, pela primeira vez, em 1943, no trabalho de McCulloch e Pitts [29]. Estas redes são amplamente utilizadas no reconhecimento de padrões, como reconhecimento de objetos e de fala.

Assim, como numa rede neuronal biológica, as unidades básicas de uma RNA são conhecidas como neurónios. Cada neurónio pode ser representado como na figura seguinte.

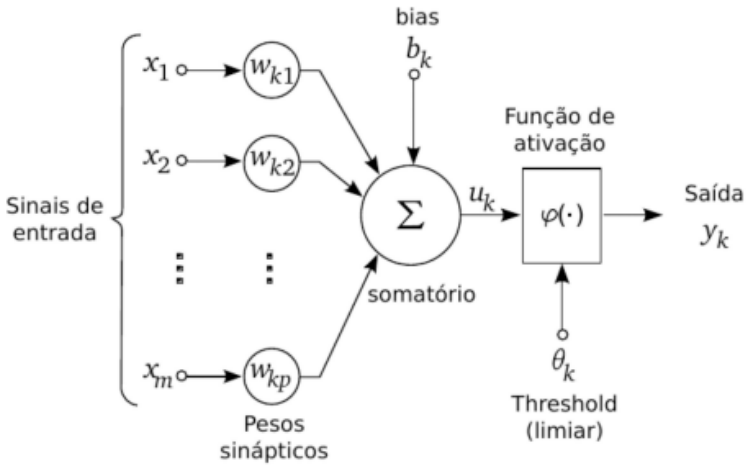


Figura 2.15: Neurónio Artifical [36]

Onde:

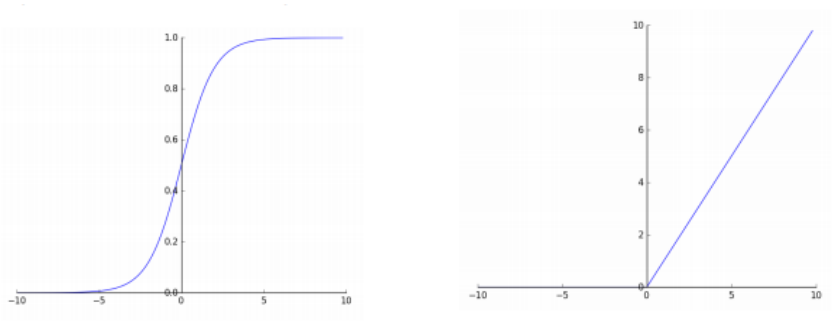
- Sinal de entrada (x_m): conjunto de dados de entrada que servem para o treino e funcionamento da rede.
- Pesos sinápicos (w_{kp}): cada sinal de entrada tem associado um peso de forma a determinar a sua influência na rede.
- Limiar de ativação (b_k): parâmetro que permite uma melhor adaptação da rede.
- Somatório: realiza a soma de todas as entradas multiplicadas pelos pesos associados.
- Função de ativação ($\varphi(.)$): função que permite determinar como os neurónios são ativados.
- Saída (y_k)

Em termos matemáticos, as unidades básicas de processamento podem ser descritas pela seguinte equação:

$$y_k = \varphi\left(\sum_{i=1}^{m} x_i w_{ki} + b_k\right) \tag{2.1}$$

Funções de Ativação

As funções de ativação são funções não-lineares presentes no final da estrutura de um neurónio e definem a saída com base nos dados de entrada e no limiar de ativação.



(a) Função Sigmóide (b) Função ReLu

Figura 2.16: Exemplos de funções de ativação [19]

A função sigmóide (a) está presente nas camadas de saída de uma rede neuronal, sendo amplamente utilizada em problemas de classificação, tem como saída valores entre 0 e 1 e resulta na probabilidade dos dados de entrada estarem contidos na classe analisada. Esta função é do tipo:

$$\varphi(x) = \frac{1}{1 + e^{-x}} \tag{2.2}$$

A função Linear Retificada (*Rectified Linear Unit - ReLu*) é uma função de ativação de aprendizagem rápida, provando ser uma função bem sucedida e extensamente utilizada. Oferece um bom desempenho e generalização comparativamente à função sigmóide, pois não faz uso de expoentes como podemos verificar em (b). Permite preservar as propriedades de modelos lineares, o que os torna fáceis de otimizar. É do tipo:

$$\varphi(x) = max(0, x) \tag{2.3}$$

A função *softmax* pode assumir valores entre 0 e 1, sendo que a soma de todas as probabilidades de todas as classes é igual a 1. É usada em modelos multi-classe retornando a probabilidade de cada classe, com a classe alvo obtendo uma maior probabilidade. A função *softmax* aparece em quase todas as camadas de saída das arquiteturas de aprendizagem profunda. É calculada usando a relação:

$$\varphi(x_i) = \frac{exp(x_i)}{\sum_j exp(x_j)} \tag{2.4}$$

2.5.2 Redes Neuronais com Multicamadas

Uma rede neuronal composta por apenas um neurónio é capaz de classificar padrões, no entanto tem uma capacidade limitada. Como tal surgiram novas arquiteturas mais robustas constituídas por conjuntos de neurónios distribuídos em camadas. A figura seguinte mostra como é feita essa distribuição, sendo que a primeira camada representa as entradas, a última a saída e todas as camadas intermediárias são designadas como camadas ocultas.

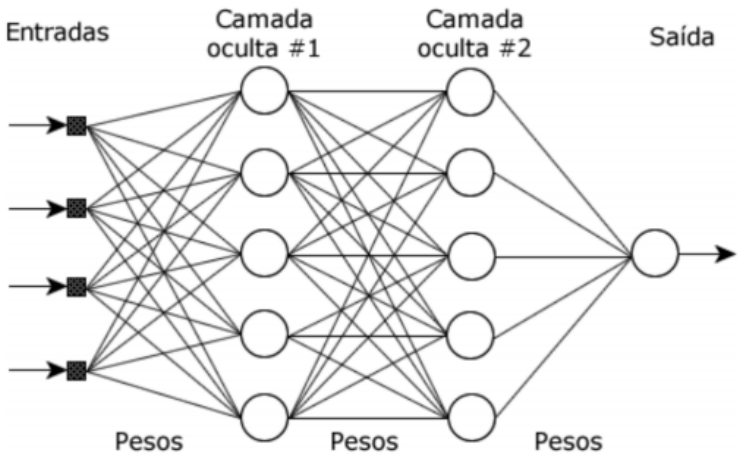


Figura 2.17: Rede neuronal composta por várias camadas [8]

2.5.3 Codificação

Numa fase inicial é atribuída a cada categoria um valor inteiro. Por exemplo, "vermelho"é 1, "azul"é 2 e "verde"é 3. Este tipo de codificação é conhecida como codificação através de inteiros. Para certas variáveis, isto pode ser suficiente uma vez que os valores inteiros têm uma relação ordenada natural entre si o os algoritmos de aprednizagem de máquina conseguem entender e aproveitar essa relação. Para variáveis categóricas, onde não existe uma relação ordinal, a codificação através de inteiros não é suficiente. Usar este tipo de codificação e permitir que o modelo assuma uma ordenação natural entre categorias poderá resultar num baixo desempenho ou resultados inesperados. De modo a superar este problema é necessário abordar o problema sob outra perspetiva, a codificação *one-hot* permite que os valores inteiros sejam convertidos em vetores binários. No exemplo anterior da variável "cor", existem 3 categorias e, portanto, são necessárias 3 variáveis binárias. A figura seguinte exemplifica esta conversão.

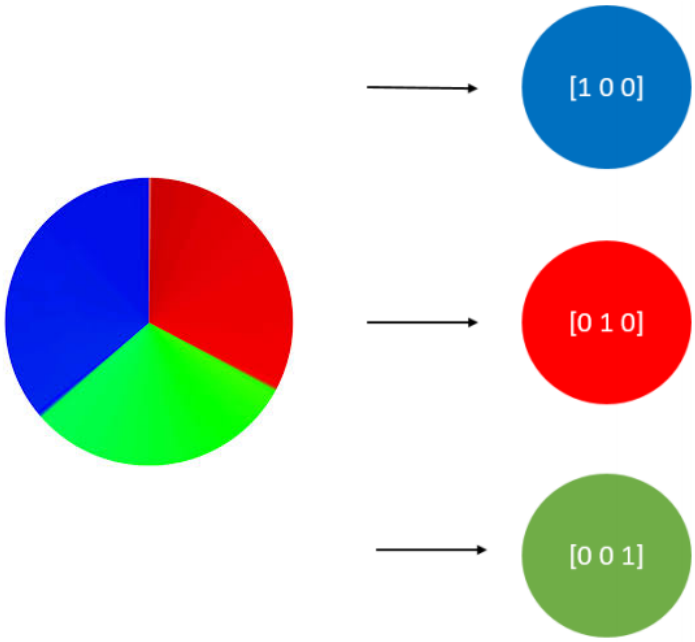


Figura 2.18: Codificação One-Hot

A codificação *one-hot* está fortemente relacionada com a codificação *multi-label*, sendo ambas variantes do problema de classificação, onde múltiplos rótulos podem ser atribuídos a cada instância. A classificação *multi-label* é uma generalização da classificação *one-hot* onde não existem restrições sobre quantas classes uma determinada instância pode ter. Este tipo de codificação tem como objetivo encontrar um modelo que mapeie as entradas x para vetores binários y atribuindo o valor 0 ou 1 a cada elemento. A figura seguinte pretende demonstrar esse princípio.

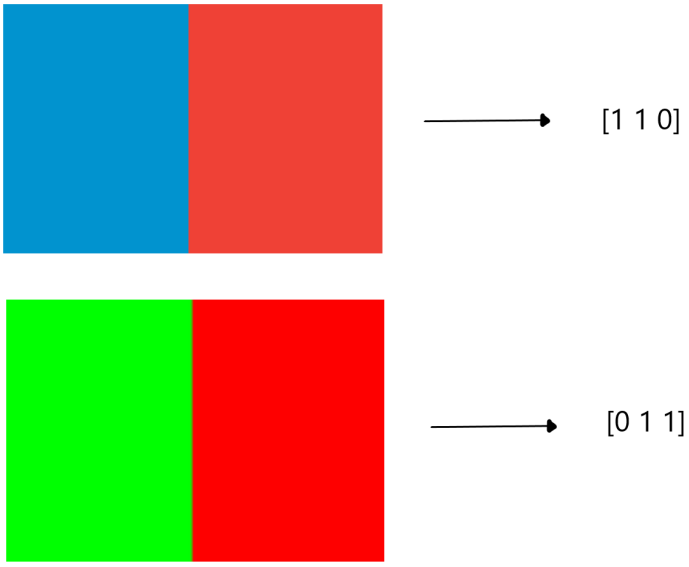


Figura 2.19: Codificação Multi-Label

2.5.4 Entropia Cruzada

Entropia Cruzada (*Cross Entropy*) permite medir a disparidade entre os valores da codificação e a saída da rede, ou seja, o número médio de bits necessário para identificar um evento a partir de um conjunto de possibilidades. Assim sendo, o vetor de saída da rede representa uma distribuição de probabilidades e a medida de erro (neste caso, entropia cruzada) indica a distância entre o que a rede acredita que essa probabilidade deve ser e o que realmente é. É representada pela seguinte fórmula:

$$D(S, L) = -\sum_{i}^{n} L_i log(S_i) \tag{2.5}$$

Onde:

- "D"representa o parâmetro Entropia Cruzada
- "S"é o vetor de saída
- "L"é o vetor de codificação

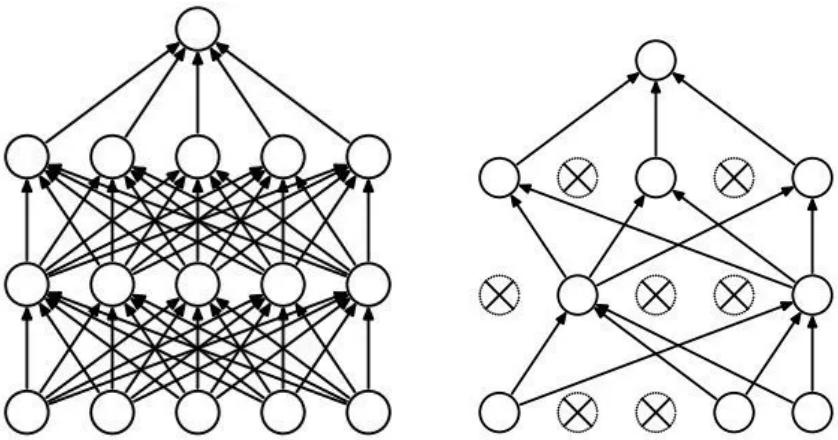
2.5.5 Função de Perda

A Função de Perda (*Loss Function*) permite determinar o quão longe a predição da rede está do resultado esperado. Para tal é feita uma média das distâncias entre os vetores de saída e a saída desejada de toda a rede. A equação seguinte permite observar que o cáculo é realizado através da média de cada Entropia Cruzada

$$L = \frac{1}{N} \sum_{i} D(S(wx_i + b), L_i) \tag{2.6}$$

2.5.6 Dropout

O *Dropout* é uma técnica de regularização que consiste em remover, aleatoriamente, certos neurónios de uma camada juntamente com todas as conexões, ou seja, a sua contribuição é temporariamente removida. A escolha das unidades descartadas é um processo aleatório. Esta técnica permite à rede ter a aptidão de aprender atributos robustos, uma vez que um neurónio não vai depender da presença específica de outro. Assim sendo a rede torna-se menos sensível a pesos específicos, resultando numa melhor generalização e redução do *overfitting*.



(a) Rede Neuronal Standard [37] (b) Rede Neuronal após Dropout [37]

Os neurónios removidos são posteriormente reintroduzidos na rede com os seus pesos inalterados. Esta técnica pode ser implementada em diversos tipos de camadas, como nas camadas convolucionais e camadas totalmente conectadas.

2.5.7 Otimizadores

Durante o processo de treino são ajustados e alterados diversos parâmetros de modo a minimizar a função de perda (descrita no ponto 2.5.5) de forma a que o modelo consiga generalizar os dados de entrada. Os otimizadores permitem interligar a função de perda e os parâmetros da rede, atualizando o modelo dependendo do resultado da função de perda [5]. A função de perda vai "informar"os otimizadores sobre o "caminho"a seguir, de forma a obter melhores resultados. De seguida são apresentados dois dos otimizadores mais utilizados:

Gradiente Descendente Estocástico - GDE

O método GDE [41] é um método de otimização que tenta minimizar o erro da rede. Esta minimização realiza-se modificando os pesos e os limiares de ativação, tendo como objetivo encontrar o mínimo local da função de perda. É uma adaptação do método Gradiente Descendente (GD) e visa resolver os problemas que este método apresenta em conjuntos de dados muito grandes. Enquanto que o método Gradiente Descendente percorre todas as amostras do conjunto de treino para fazer apenas uma atualização de um parâmetro numa iteração em particular, o método Gradiente Descendente Estocástico utiliza apenas uma amostra do conjunto de treino para fazer a atualização de um parâmetro numa iteração em particular. É chamado de estocástico pois as amostras selecionadas são aleatórias. O método GDE converge muito mais rápido comparativamente ao GD, no entanto a função de perda não é tão bem minimizada. Na maioria dos casos, a aproximação obtida pelo GDE é suficiente para alcançar os valores ideais.

Adam

Adam [24] é um algoritmo de otimização simples de implementar, computacionalmente eficiente, com poucos requisitos de memória e é utilizado em problemas com grandes conjuntos de dados e/ou parâmetros. Numa primeira fase o método calcula uma média exponencial dos gradientes (m_t) e de seguida calcula uma média exponencial dos quadrados dos gradientes (v_t). Os hiper-parâmetros β_1, β_2 controlam o decaimento exponencial dessas duas médias. A atualização da taxa de aprendizagem do algoritmo é feita da seguinte forma:

$$\Delta_t = \alpha \frac{m_t}{v_t} \tag{2.7}$$

Onde:

- "α"é a taxa de aprendizagem
- "v_t"é o quadrado do gradiente
- "m_t"é o média exponencial do gradiente

Os parâmetros são atualizados com base na combinação dos cálculos anteriores, no entanto é necessário ter em atenção os limites superiores que devem ser respeitados na atualização:

$$\begin{aligned} |\Delta_t| \leq \frac{\alpha(1-\beta_1)}{\sqrt{1-\beta_2}} \text{ , quando } (1-\beta_1) > \sqrt{1-\beta_2} \\ |\Delta| \leq \alpha \text{ , caso contrário} \end{aligned} \tag{2.8}$$

Este método é adequado para problemas onde é necessário fazer uma análise não estacionária, bem como para problemas com gradientes muito ruidosos e/ou dispersos. Os hiper-parâmetros apresentam interpretações intuitivas e geralmente necessitam de poucos ajustes.

2.5.8 Redes Neuronais Convolucionais - RNC

As RNCs são das arquiteturas mais estudadas em aprendizagem profunda e resultam, geralmente, da combinação de três camadas:

- Camada convolucional

  Como o nome sugere, nestas camadas são realizadas operações de convolução. As camadas convolucionais são conjuntos de filtros não lineares que percorrem sequencialmente os dados de entrada e produzem matrizes chamadas mapas de características (*feature maps*). A figura que se segue exemplifica uma operação de convolução, onde um filtro de tamanho 3x3 sobrepõe uma região de dados, onde a multiplicação matricial é computada e os valores somados são passados para o mapa de características.

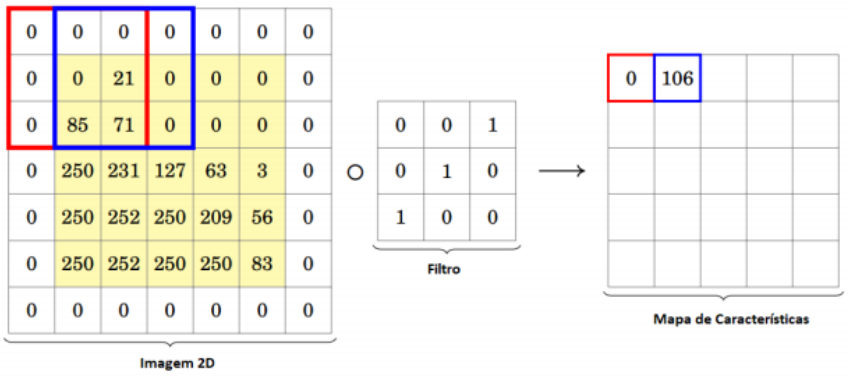


Figura 2.20: Operação de convolução [34]

Durante o processo de treino estes filtros são ajustados automaticamente de forma a serem ativados na presença de certas características relevantes. Em cada uma destas camadas são utilizados filtros e os mapas de características obtidos são agrupados. As camadas convolucionais podem variar em profundidade (*depth*) consoante a forma como os mapas são empilhados, altura (*width*) e altura (*height*) conforme os dados percorrem a RNC. Estas variações ocorrem quando existem variações em dois dos parâmetros das operações de convolução: o passo dos filtros (*stride*) e o preenchimento da camada (*padding*).

  - Passo (*stride*)

    Como foi falado anteriormente os filtros nas operações de convolução percorrem as matrizes de forma sequencial. Este processo ocorre em passos, de pixel para pixel numa imagem ou de posição para posição numa matriz. Quando o passo é igual a

um, a altura e a largura da camada de saída será igual à entrada. Se o passo for igual a dois, a saída terá metade do tamanho de entrada e assim sucessivamente.

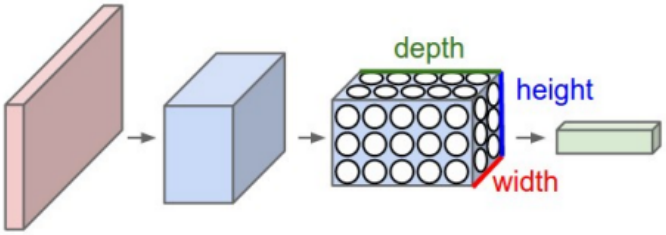


Figura 2.21: Camadas convolucionais [2]

– Preenchimento (*padding*)

Ao aplicar os filtros descritos anteriormente é necessário lidar com as bordas das imagens. Os processos mais utilizados são o ***valid padding*** ou ***same padding***. Relativamente ao *valid padding* as bordas do filtro não ultrapassam as bordas da imagem, enquanto que no *same padding* as fronteiras da imagem são preenchidas com 0 de modo a controlar a altura e largura da camada de saída. Este preenchimento é realizado através da equação $P = \frac{K-1}{2}$, onde K representa o tamanho do filtro. De seguida é apresentado um exemplo de como o preenchimento é feito numa imagem de 32x32 com um filtro de 5x5.

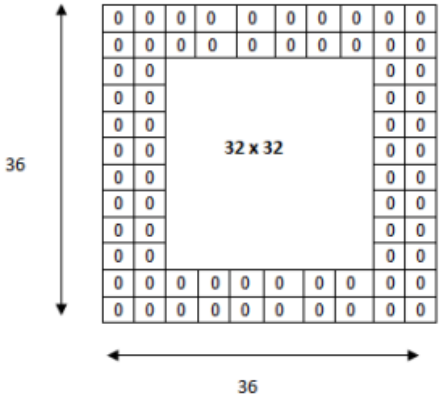


Figura 2.22: Preenchimento para filtro 5x5 [15]

Após ter os parâmetros de convolução estarem definidos, é possível determinar a dimensão de saída da camada de convolução através da equação $O = \frac{W-K-2P}{s} + 1$, onde O representa a dimensão de saída, W a dimensão de entrada, K o tamanho do filtro, P o preenchimento e S o passo do filtro.

- Camada de Pooling

  A camada de *pooling* tem como principal objetivo reduzir a dimensão da camada de entrada de forma a reduzir o custo computacional e evitar *overfitting*. O método, chamado de *Max Pooling*, consiste em reduzir a dimensão das camadas utilizando o valor máximo de cada região. Assim, suprime valores desprezíveis, criando uma invariância a pequenas mudanças e distorções locais [9].

- Camada Totalmente Conectada

  As camadas totalmente conectadas (*fully-connected layer - FCL*) são camadas onde os neurônios das camadas anteriores estão conectados com os neurônios desta camada. As características extraídas nas camadas convolucionais e de *pooling* são classificadas e utiliza-se uma função de ativação para prever a classe do objeto.

2.5.9 Redes Neuronais Convolucionais Mais Populares

Quase todas as arquiteturas seguem os mesmos princípios gerais, onde são aplicadas sucessivas camadas convolucionais, reduzindo periodicamente as dimensões espaciais e aumentando os mapas de características.

ImageNet Large Scale Visual Recognition Challenge (ILSVRC) é um grande banco de dados visuais, uma referência na classificação e deteção de objetos, contendo milhões de imagens, tornando-se num dos desafios mais importantes em visão computacional, tendo atraído ao longo dos anos muita atenção. Muitas arquiteturas foram testadas e as seguintes obras merecem ser destacadas:

- **AlexNet**

  AlexNet [20] é uma arquitetura composta por 5 camadas convolucionais e 3 camadas totalmente conectadas. A função *ReLu* é aplicada depois de cada camada convolucional e totalmente conectada. São utilizados *dropouts* antes da primeira e segunda camada totalmente conectada. Esta arquitetura venceu, em 2012, o *ILSVRC* obtendo uma taxa de erro de 15.3% superando a segunda melhor por mais de 10%. Pela primeira vez foi utilizado GPU na sua implementação, bem como *dropouts* de forma a evitar o *overfitting*. Ficou ainda provado que a função de ativação *ReLu* é mais eficaz que a tangente hiperbólica nas camadas convolucionais, podendo tornar o treino mais rápido.

- **VGGnets**

  A arquitetura VGG [14] faz o aprimoramento relativamente à AlexNet, substituindo grandes filtros (*kernel-size* 11 e 5 na primeira e segunda camada convolucional respetivamente) por vários filtros de tamanho 3x3, um após o outro. Com um dado campo recetivo (tamanho efetivo da área da imagem de entrada da qual a saída depende) , filtros de menores dimensões são melhores , uma vez que várias camadas não lineares aumentam a profundidade da rede o que permite aprender recursos mais complexos, com um custo computacional menor.

- **GoogLeNet**

  GoogLeNet [38], também conhecida por *Inception V1*, venceu o concurso *ILSVRC* em 2014 atingindo uma taxa de erro de 6.67%. Esta arquitetura, inspirada na *LeNet* introduziu módulos baseados em várias convoluções de pequenas dimensões, de modo a reduzir de forma significativa o número de parâmetros. Esta arquitetura consiste numa rede neuronal com 22 camadas capaz de reduzir o número de parâmetros de 60 milhões (*AlexNet*) para 4 milhões.

- **ResNet**

  Esta arquitetura [22] foi introduzida em 2015 e a principal ideia por trás da *ResNet* é que cada camada deve aprender apenas uma correção residual da camada anterior, permitindo treinar redes muito profundas de forma rápida e eficiente. Sendo esta uma arquitetura bastante profunda, podendo ter até 152 camadas, apresentou um desempenho excelente no que diz respeito à generalização dos dados, o que a levou a ganhar vários prémios em 2015.

3 Jogos Terapêuticos

Os jogos muitas das vezes não têm associado um propósito particular e são motivados pelo desejo do utilizador se divertir. Na sua grande maioria, os jogos são formais e têm regras bem definidas sobre como o jogo é jogado. Embora possam transmitir ao utilizador um sentimento de diversão, geralmente envolvem algum tipo de competição entre os jogadores, existindo uma disputa entre poderes, definida por regras, de forma a produzir um resultado desequilibrado (um dos jogadores vence o jogo). Isto permite que exista um maior envolvimento por parte dos participantes e, consequentemente, um sentimento de prazer acrescido. Um jogo requer um processo de aprendizagem, de modo a que sejam desenvolvidas capacidades que permitam ao utilizador superar as dificuldades/níveis apresentados. Essas dificuldades poderão ser ultrapassadas através de processos de imitação (observação do comportamento de outro jogador) ou através de processos de repetição. No âmbito da terapia, é usual a utilização de diversos jogos de forma a ajudar os pacientes a superar certas ambiguidades. Como é de esperar, os jogos são diferentes de caso para caso, dependendo do tipo limitação/doença. Segundo alguns especialistas [28] poderá ser através de jogos que o paciente pode atribuir certas características e funções a objetos ou personagens, possibilitando um desenvolvimento comportamental.

Como discutido anteriormente, autismo é um distúrbio neurológico caracterizado pelo comprometimento da interação social, comunicação verbal e não verbal, comportamento restrito e repetitivo. É difícil para estes indivíduos interpretar, avaliar e até mesmo expressar emoções básicas como felicidade ou tristeza. Como tal, de seguida, serão propostos diversos jogos terapêuticos com o objetivo de ajudar estes indivíduos a superar tais ambiguidades.

- Jogo 1

  Jogo de correspondência, onde existirá uma coluna com os diferentes rótulos de emoções e um exemplo (estímulo visual) associado. Ao lado um conjunto de imagens sem rótulo e o objetivo seria mover essas imagens para o local correto. Caso o utilizador coloque uma carta no sítio errado, esta volta ao local original e a pontuação não é alterada. O jogo teria uma duração limite e à medida que o paciente move as imagens sem rótulo para o espaço designado a sua pontuação aumentará.

- Jogo 2

  Jogo de imitação, onde irão aparecer diversas imagens da mesma emoção e respetivo rótulo. O paciente terá que imitar a expressão facial, esta será analisada pelo software de reconhecimento automático de expressões faciais e quando coincidir, outra expressão será apresentada. Mais uma vez, o tempo será limitado e quantas mais expressões o indivíduo conseguir imitar maior a sua pontuação.

- Jogo 3

  Jogo de memória, onde um conjunto de imagens são apresentadas com uma determinada emoção, após alguns segundos desaparecem e no seu lugar surgem diferentes rótulos. O utilizador terá que recordar qual a emoção apresentada e selecionar o rótulo que a define a emoção. O tempo só começa a contar a partir do momento em que as imagens desaparecem e surgem os rótulos. Com tempo limitado, a pontuação será superior quanto maior for o número de respostas corretas.

- Jogo 4

  Jogo de atribuição, onde um conjunto de imagens irão ser apresentadas com os respetivos rótulos. Os rótulos poderão estar incorretos e o objetivo será trocar a sua ordem de forma a associar os rótulos ás imagens correspondentes. Quando estiver satisfeito com a disposição dos rótulos o utilizador carrega num botão que permite fazer a avaliação, de seguida será apresentado um quadro que resume o seu desempenho e a pontuação obtida. Este jogo terá uma duração limite e a pontuação irá depender do número de rótulos que o utilizador acertar.

4 Implementação Sistema automático de reconhecimento de expressões faciais

Neste capítulo serão descritas as metodologias utilizadas para a solução do problema apresentado.

4.1 Bases de dados

A qualidade dos sistemas AFER depende significativamente da escolha do conjunto de dados. A crescente necessidade, no campo do reconhecimento facial, de ferramentas de comparação para algoritmos de avaliação levou à criação de diferentes bases de dados.

A primeira tentativa proeminente para a criação de um conjunto de dados surgiu com *Facial Recognition Technology dataset* (FERET) que já não se encontra disponível. Este foi seguido por muitos outros como *Conh-Kanade* ou *AR face*. Devido à natureza complexa das expressões, algumas restrições podem estar presentes no conjunto de dados. De modo a alcançar uma maior robustez, o conjunto de dados tem de ser versátil e possuir uma grande variedade. O objetivo é reduzir a distância do contexto real, exibindo expressões espontâneas, estando ciente das mudanças de iluminação, rotações, oclusões e outros aspetos que podem afetar os sistemas de reconhecimento.

Seguindo esta direções , surgiram algumas bases de dados, *MMI Facial Expression Dataset* foi um deles. Segundo [32] esta base de dados tem como objetivo fornecer grandes volume de dados visuais para a comunidade de análise de expressões faciais. Provaram que este conjunto de dados é bastante abrangente e composto por dados relevantes e adequados para a análise de expressões, apesar de apresentar algumas limitações rela-

tivas ás fases das expressões. Mais recentemente surgiram novos conjuntos de dados, como *Facial Expression Recognition* (FER), que merecem ser mencionados devido à sua tentativa significativa de fornecer dados com expressões faciais mais autênticas ou naturais.

Existem ainda dois aspetos que requerem alguma atenção quando consideramos conjuntos de dados faciais, a necessidade de dados rotulados, bem como a ausência de uma das três fases de uma expressão facial (*onset, apex e offset*)

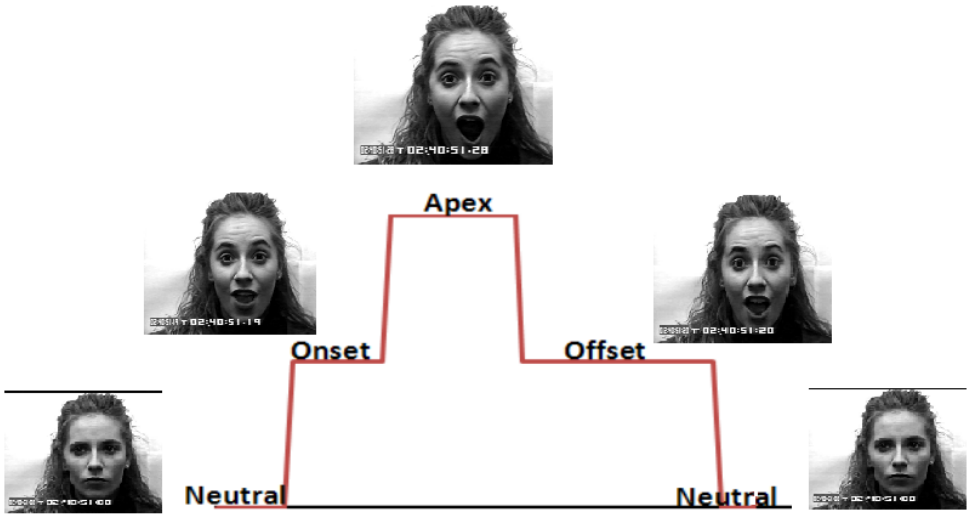


Figura 4.1: Fases de uma expressão facial

Rotular dados relativos a expressões faciais é um campo complexo, além disso, observações de contexto e não-linearidade podem ser uma preocupação. As questões acima mencionadas poderão ser minimizadas através de uma fonte de dados confiável e consistente.

Atualmente, existe uma grande variedade de conjuntos de dados para o reconhecimento de expressões, onde um grande número de imagens são rotuladas com emoções. No entanto, o caso difere quando o reconhecimento de expressões é realizado a partir de unidades de ação. Embora existam alguns conjuntos de dados rotulados com AUs, como a base de dados *Extended Cohn-Kanade*, estes apresentam poucos dados e/ou são criados sob ambientes controlados, o que os torna pouco adequados para treinar um classificador robusto com suporte adequado para as características de interesse. Como tal, na presente dissertação, é proposto um método de reconhecimento de unidades de ação com RNC treinadas a partir de conjuntos de dados rotulados com emoções. Os testes realizados ao longo deste trabalho utilizaram o conjunto de dados *FER2013*.

Base de dados FER2013

O conjunto de dados FER2013 [12] foi desenvolvido por Pierre-Luc Carrier e Aaron Courville como parte de um projeto de pesquisa. Os dados consistem em imagens de faces em tons de cinza (48x48 pixeis). As faces dos indivíduos foram registadas de forma a que fiquem centradas e ocupem aproximadamente o mesmo espaço em cada imagem. A base de dados foi desenvolvida com o objetivo de categorizar cada rosto com base na expressão facial, segundo sete categorias (0=Raiva, 1=Aversão, 2=Medo, 3=Felicidade, 4=Tristeza 5=Surpresa, 6=Neutro).

Esta base de dados está disponível num ficheiro .csv contendo duas colunas, "emoção"e "pixeis". A primeira coluna contém um código numérico que varia de 0 a 6 que representa a emoção presente na imagem. A segunda coluna contém as imagens.

O conjunto de dados foi dividido em três grupos: treino, validação e teste. Para o treino da rede foram usadas 28,709 imagens, para validar o modelo foram usadas 3,589 imagens e, por fim, de forma a testar o modelo, foram usadas 3,589 imagens.

4.2 Processamento do Conjunto de Dados

Numa fase inicial, de forma a construir o modelo desejado, verificou-se a distribuição de dados.

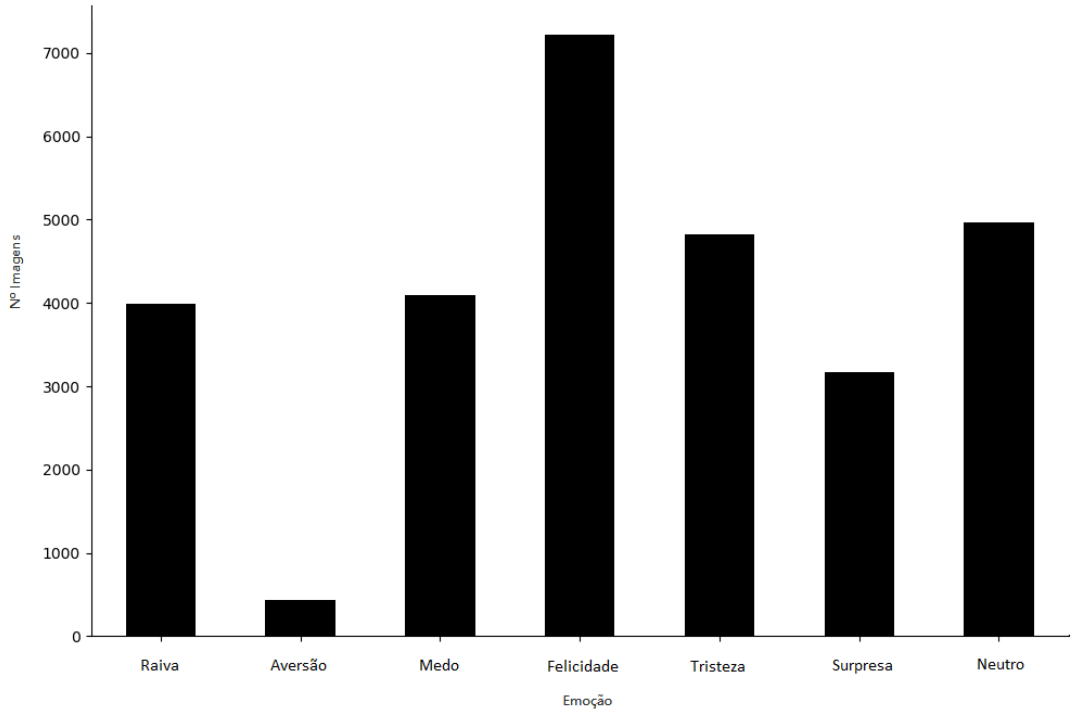


Figura 4.2: Distribuição de dados

Após análise, é possível verificar que o número de amostras pertencentes à classe "Aversão"é significativamente menor do que qualquer outra classe. Como tal, os dados pertencentes a esta classe foram removidos, uma vez que não apresentam amostras suficientes para uma classificação fidedigna e o conjunto de dados torna-se mais equilibrado.

O conjunto de dados FER2013 foi, originalmente, codificado da seguinte forma:

Emoção	**Codificação**
Raiva	[1,0,0,0,0,0,0]
Aversão	[0,1,0,0,0,0,0]
Medo	[0,0,1,0,0,0,0]
Felicidade	[0,0,0,1,0,0,0]
Tristeza	[0,0,0,0,1,0,0]
Surpresa	[0,0,0,0,0,1,0]
Neutro	[0,0,0,0,0,0,1]

Tabela 4.1: Codificação original

Posteriormente foi necessário alterar a codificação original, uma vez que os dados pertencentes à classe "Aversão"foram eliminados. Assim, o modelo permite analisar 6 expressões faciais, ao invés das 7 originalmente presentes no conjunto de dados.

Emoção	**Codificação**
Raiva	[1,0,0,0,0,0]
Medo	[0,1,0,0,0,0]
Felicidade	[0,0,1,0,0,0]
Tristeza	[0,0,0,1,0,0]
Surpresa	[0,0,0,0,1,0]
Neutro	[0,0,0,0,0,1]

Tabela 4.2: Codificação após eliminar classe

De forma a verificar a existência de unidades de ação é necessário estabelecer uma correspondência entre emoções e AUs. Segundo o método apresentado por Paul Ekman e Wallace Friesen, denominado de FACS (descrito na secção 2.1), é possível descrever qualquer ocorrência de expressões faciais como combinações de unidades de ação. Esta abordagem não permite interpretar o significado das expressões, possibilitando apenas reconhecer emoções através da agregação de diferentes unidades de ação.

A tabela seguinte, baseada na informação apresentada na tabela 2.1, indica as diferentes combinações de unidades de ação que permitem reconhecer emoções.

Emoção	**AU**
Raiva	4+5+7+23
Medo	1+2+4+5+7+20+26
Felicidade	6+12
Tristeza	1+4+15
Surpresa	1+2+5+26
Neutro	None

Tabela 4.3: Emoção e respetivas unidades de ação

Ao verificar a Tabela 4.2 é possível concluir que o modelo terá de analisar 11 unidades de ação (1, 2, 4, 5, 6, 7, 12, 15, 20, 23, 26). Como é observável, a expressão "Neutro"não apresenta qualquer unidade de ação, como tal, será necessário criar uma nova classe para esta expressão, diferente de qualquer outra. É importante salientar que podem existir mais unidades de ação que caracterizam cada uma das expressões, no entanto, nesta abordagem apenas serão consideradas aquelas que estão presentes independentemente da intensidade da emoção.

Agora que é conhecida a relação entre unidades de ação e emoções é necessário alterar a codificação inicial, de forma a que o modelo consiga avaliar cada uma das expressões através das unidades de ação presentes na face de cada indivíduo. A tabela seguinte indica como foram codificados os novos rótulos:

Emoção	**Codificação**
Raiva	[0,0,1,1,0,1,0,0,0,1,0,0]
Medo	[1,1,1,1,0,1,0,0,1,0,1,0]
Felicidade	[0,0,0,0,1,0,1,0,0,0,0,0]
Tristeza	[1,0,1,0,0,0,0,1,0,0,0,0]
Surpresa	[1,1,0,1,0,0,0,0,0,0,1,0]
Neutro	[0,0,0,0,0,0,0,0,0,0,0,1]

Tabela 4.4: Codificação das emoções segundo unidades de ação

Cada elemento do vetor representa uma unidade de ação. As imagens irão receber estes rótulos, em vez dos apresentados na Tabela 4.2, e ambos os parâmetros (imagens e rótulos) serão a entrada da rede.

Após alterar a codificação verificou-se a seguinte distribuição de dados:

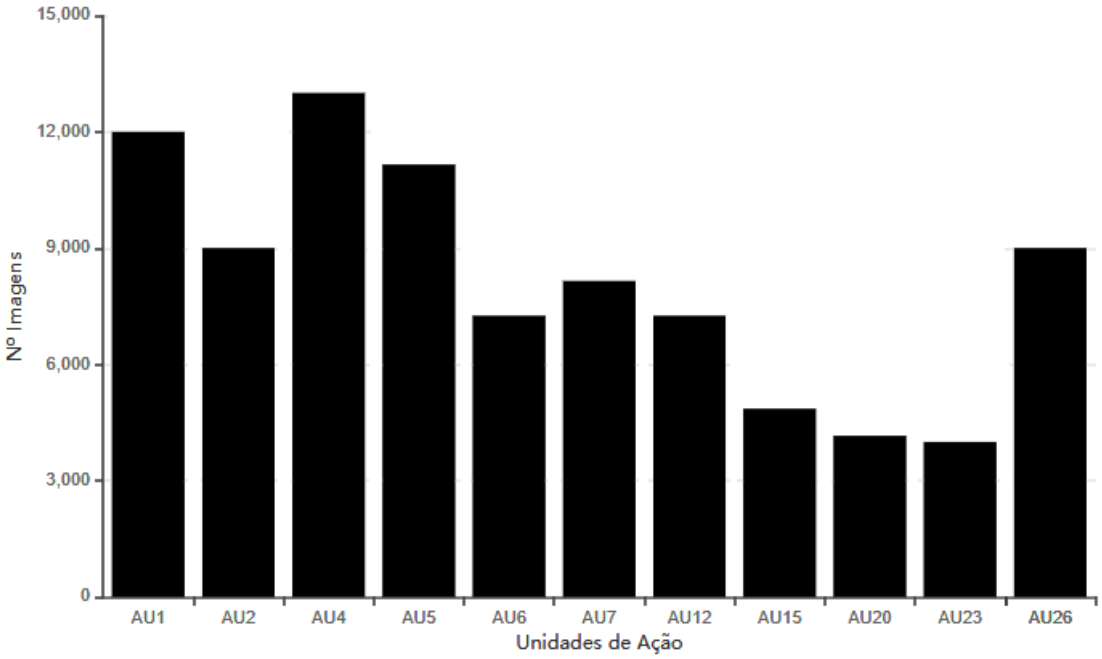


Figura 4.3: Distribuição de AUs

5 Resultados Experimentais

5.1 Sistema AFER

Treino Completo da Rede

Ao longo da presente dissertação diversos testes foram realizados, utilizando diferentes arquiteturas. Numa fase inicial o treino foi realizado de raiz, ou seja, os pesos foram inicializados aleatoriamente e otimizados considerando diretamente a entrada e saída da rede.

Nas tabelas seguintes serão apresentados os resultados obtidos:

Arquitetura	**AU1**	**AU2**	**AU4**	**AU5**	**AU6**	**AU7**	**AU12**	**AU15**	**AU20**	**AU23**	**AU26**
LeNet-5	71.68%	64.30%	76.32%	71.36%	80.05%	54.31%	79.90%	29.46%	29.00%	42.91%	64.50%
AlexNet	76.74%	71.68%	79.79%	76.52%	85.00%	63.86%	84.78%	41.06%	46.61%	53.99%	71.65%
VGG16	**78.21**%	74.56%	81.94%	78.51%	85.75%	66.91%	85.81%	**56.50**%	54.58%	60.24%	74.72%
VGG19	77.50%	**75.18**%	**82.64**%	**79.48**%	**85.83**%	**70.23**%	**85.86**%	53.90%	**57.89**%	**63.51**%	**75.14**%
Xception	73.86%	68.95%	77.86%	74.62%	81.60%	58.04%	81.33%	41.51%	33.97%	52.38%	68.87%
ResNet50	75.49%	71.76%	78.89%	77.23%	84.06%	66.14%	84.24%	53.43%	52.62%	57.51%	71.78%

Tabela 5.1: Resultados obtidos por unidade de ação

Arquitetura	**Classe Neutro**
LeNet-5	51.77%
AlexNet	59.86%
VGG16	**64.48**%
VGG19	63.48%
Xception	54.32%
ResNet50	58.24%

Tabela 5.2: Resultados obtidos para a classe Neutro

Como é possível observar, ao analisar as tabelas acima apresentadas, a arquitetura que, em média, apresenta melhores resultados é a *VGG19* com uma precisão de 75.81% no conjunto de dados de teste, seguida da *VGG16*, que obteve uma precisão de 73.53%. Ambas as arquiteturas apresentam uma boa generalização dos dados e conseguinte classificação de unidades de ação. A arquitetura *AlexNet* apresentou uma precisão de 71.44%, *ResNet50* obteve 70.93%, *Xception* atingiu os 67.09% e por fim a *LeNet-5* que alcançou os 65.95%.

Transferência de Conhecimento

Posteriormente, foram aplicados métodos de *transfer learning* nas arquiteturas que apresentaram melhores resultados, neste caso, a *VGG19* e *VGG16*. Estes métodos são aplicados através de modelos pré-treinados, ou seja, um modelo que foi treinado com um grande conjunto de dados que permitem resolver problemas semelhantes ao pretendido. Uma rede neuronal é geralmente composta por duas partes, **base convolucional** e o **classificador**. A base convolucional consiste num conjunto de camadas convolucionais e de *pooling* e o objetivo é gerar características a partir das imagens do conjunto de dados enquanto que o classificador é composto pelas camadas totalmente conectadas e o seu propósito é classificar a imagem segundo as características extraídas. Os dois modelos pré-treinados utilizados foram **ImageNet** e **VGGFACE**.

- ImageNet [35]

  Conjunto de dados com mais de 15 milhões de imagens com o respetivo rótulo, pertencentes a cerca de 22 mil categorias. As imagens foram obtidas online e rotuladas por rotuladores humanos usando a ferramenta *Mechanical Turk* da Amazon. Surgiu em 2010, fazendo parte da competição anual *ImageNet Large-Scale Visual Recognition Challenge* (ILSVRC).

- VGGFACE [33]

  Modelo pré-treinado com imagens de faces. A arquitetura do modelo é baseada na *VGG16* e é pré-treinada num conjunto de dados com 2.6 milhões de imagens de 2600 pessoas diferentes.

Aplicando métodos de transferência de conhecimento apenas parte da rede foi treinada, sendo que parte das características extraídas pertencem aos dois modelos discutidos previamente. Numa primeira fase foram treinadas as últimas **3 bases convolucionais e o classificador** de cada arquitetura. Os resultados obtidos foram os seguintes:

Arquitetura	AU1	AU2	AU4	AU5	AU6	AU7	AU12	AU15	AU20	AU23	AU26
VGG16 (ImageNet)	**77.88**%	**74.82**%	81.30%	77.72%	**84.96**%	67.37%	**84.94**%	52.59%	56.70%	59.25%	**74.87**%
VGG16 (VGGFace)	72.85%	69.09%	76.27%	74.07%	80.00%	63.33%	79.95%	46.27%	48.34%	55.11%	69.35%
VGG19 (ImageNet)	77.18%	74.72%	**82.31**%	**78.88**%	84.51%	**69.11**%	84.50%	**53.90**%	**57.30**%	**62.40**%	74.75%

Tabela 5.3: Resultados obtidos por unidade de ação

Arquitetura	Classe Neutro
VGG16 (ImageNet)	57.87%
VGG16 (VGGFace)	55.35%
VGG19 (ImageNet)	**60.44**%

Tabela 5.4: Resultados obtidos para a classe Neutro

A arquitetura que apresenta melhores resultados é a *VGG19* pré-treinada com a base de dados *ImageNet* que apresenta uma precisão de 73.69%, seguida da *VGG16* pré-treinada com a base de dados *ImageNet* com 72.83% e por fim a arquitetura *VGG16* pré-treinada com a base de dados *VGGFACE* com uma precisão de 68.10%.

De seguida, as últimas **2 bases convolucionais e o classificador** foram treinados. Os resultados obtidos foram os seguintes:

Arquitetura	AU1	AU2	AU4	AU5	AU6	AU7	AU12	AU15	AU20	AU23	AU26
VGG16 (ImageNet)	76.13%	**72.87**%	**80.56**%	**76.72**%	**83.19**%	**66.67**%	**83.29**%	**51.69**%	**55.10**%	57.85%	**72.86**%
VGG16 (VGGFace)	71.55%	70.74%	73.73%	73.73%	79.42%	61.34%	78.42%	41.36%	48.10%	53.41%	70.63%
VGG19 (ImageNet)	**76.27**%	71.25%	78.95%	76.62%	82.35%	66.60%	82.26%	47.45%	53.48%	**59.70**%	70.95%

Tabela 5.5: Resultados obtidos por unidade de ação

Arquitetura	Classe Neutro
VGG16 (ImageNet)	58.10%
VGG16 (VGGFace)	53.85%
VGG19 (ImageNet)	**58.27**%

Tabela 5.6: Resultados obtidos para a classe Neutro

Neste caso, a arquitetura que apresenta melhores resultados é a *VGG16* pré-treinada com a base de dados *ImageNet* que apresenta uma precisão de 71.27%, seguida da *VGG19* pré-treinada com a base de dados *ImageNet* com 70.46% e por fim a arquitetura *VGG16* pré-treinada com a base de dados *VGGFACE* com uma precisão de 64.97%.

Posteriormente apenas a ultima **base convolucional e o classificador** foram treinados. Os resultados são os seguintes:

Arquitetura	AU1	AU2	AU4	AU5	AU6	AU7	AU12	AU15	AU20	AU23	AU26
VGG16 (ImageNet)	71.77%	69.60%	74.32%	72.73%	77.40%	62.03%	76.98%	45.07%	50.43%	**51.04**%	69.38%
VGG16 (VGGFace)	69.97%	67.85%	69.67%	70.59%	71.71%	56.52%	71.60%	45.15%	45.94%	48.39%	67.85%
VGG19 (ImageNet)	**73.58**%	**71.60**%	**75.81**%	**74.77**%	**77.61**%	**62.36**%	**77.47**%	**46.62**%	**51.67**%	50.41%	**71.61**%

Tabela 5.7: Resultados obtidos por unidade de ação

Arquitetura	Classe Neutro
VGG16 (ImageNet)	54.84%
VGG16 (VGGFace)	48.14%
VGG19 (ImageNet)	**54.92**%

Tabela 5.8: Resultados obtidos para a classe Neutro

Neste contexto, a arquitetura que apresenta melhores resultados é a *VGG19* pré-treinada com a base de dados *ImageNet* que apresenta uma precisão de 69.13%, seguida da *VGG16* pré-treinada com a base de dados *ImageNet* com 65.37% e por fim a arquitetura *VGG16* pré-treinada com a base de dados *VGGFACE* com uma precisão de 63.17%.

Por fim, apenas as camadas pertencentes ao **classificador** foram treinadas. As tabelas seguintes apresentam os resultados obtidos::

Arquitetura	AU1	AU2	AU4	AU5	AU6	AU7	AU12	AU15	AU20	AU23	AU26
VGG16 (ImageNet)	54.79%	47.87%	58.92%	53.72%	**55.69**%	30.91%	**55.60**%	**19.92**%	14.36%	**23.03**%	47.20%
VGG16 (VGGFace)	59.14%	47.65%	**62.14**%	59.30%	46.92%	37.33%	46.83%	13.43%	12.67%	17.55%	47.86%
VGG19 (ImageNet)	**59.44**%	**54.35**%	59.98%	**60.56**%	53.24%	**37.48**%	52.24%	13.33%	**22.22**%	16.22%	**54.39**%

Tabela 5.9: Resultados obtidos por unidade de ação

Arquitetura	Classe Neutro
VGG16 (ImageNet)	**33.79**%
VGG16 (VGGFace)	17.98%
VGG19 (ImageNet)	29.33%

Tabela 5.10: Resultados obtidos para a classe Neutro

Assim sendo, a arquitetura que, em média, apresenta melhores resultados é a *VGG16* pré-treinada com a base de dados *ImageNet* atingindo uma precisão de 50.79%, de seguida a *VGG19* pré-treinada com a base de dados *ImageNet* com uma precisão de 48.10% e por fim a arquitetura *VGG16* pré-treinada com a base de dados *VGGFACE* com 47.31%.

Como é possível verificar qualquer um dos modelos que utilizou métodos de *transfer learning* apresenta precisões mais baixas que os modelos treinados de raiz. Isto pode ser explicado na medida em que, embora o pré-treino possa evitar parte dos problemas associados ao treino de uma rede com poucos dados, como o *overfitting*, a informação presente é dominada pelo conjunto de dados utilizado para o pré-treino que pode enfraquecer a capacidade da rede [26]. Neste caso o conjunto de dados é favorável ao treino completo da rede, como tal, é possível afirmar que a arquitetura que melhor generaliza os dados de treino é a *VGG19* que apresentou uma precisão de 75.81% no conjunto de dados de teste.

Todos os testes foram realizados com mecanismos de *Early Stopping*, que permite monitorizar diferentes parâmetros de treino de forma a reduzir o *overfitting*, sendo que. neste contexto, o treino da rede seria interrompido caso erro associado à generalização dos dados aumentasse.

De seguida são apresentados alguns exemplos do sistema AFER em funcionamento:

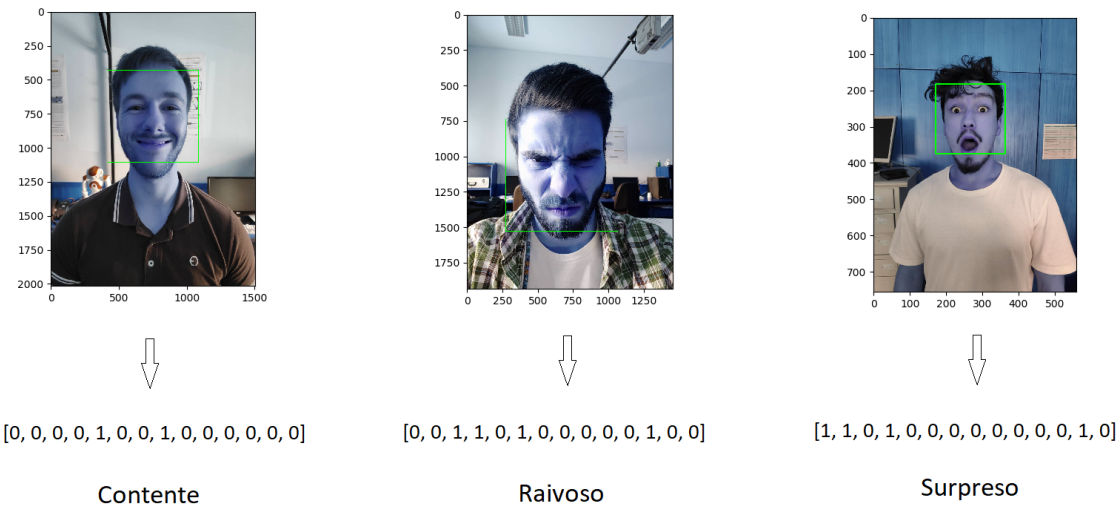


Figura 5.1: Sistema AFER em funcionamento

5.1.1 Arquitetura da rede

Perante os resultados obtidos após o treino da rede convolucional, é possível afirmar que a arquitetura VGG19 foi a que obteve melhores resultados, como tal, será descrita de seguida.

Durante o processo de treino, as imagens são passadas por um conjunto de camadas convolucionais, onde são usados filtros com um campo recetivo muito pequeno (3x3). O passo (*stride*) convolucional é fixo (1 pixel), fazendo com que a altura e largura da camada de saída seja igual à entrada. O preenchimento (*padding*) é 1 para os filtros 3x3 utilizados, permitindo que a resolução espacial seja preservada após a convolução. Nas camadas de pooling é utilizado o método *Max Pooling* realizado ao longo de janelas 2x2 pixels, com passo (*stride*) 2. O conjunto de camadas convolucionais é seguido de três camadas Totalmente Conectadas, a primeira contêm 512 canais, a segunda 128 canais e por fim, a terceira que apresenta 12 canais (um por cada classe analisada). A camada final é uma camada sigmóide que permite analisar a probabilidade dos dados de entrada estarem contidos na classe analisada. Todas as camadas ocultas apresentam a função Linear Retificada (*ReLu*) como função de ativação.

Resumindo:

- Tamanho da entrada: 48x48;
- Tamanho do campo recetivo: 3x3;
- Passo convolucional: 1 pixel;
- Preenchimento: 1;
- *Max pooling* 2x2 com passo de 2 pixels;
- 1ª camada Totalmente Conectada: 512 canais
- 2ª camada Totalmente Conectada: 128 canais
- 3ª camada Totalmente Conectada é uma camada sigmóide com 12 canais (um por cada classe analisada)
- Função de ativação: *ReLu*

5.2 Jogos Desenvolvidos e Aplicações

Para o desenvolvimento dos jogos propostos, foi utilizada a biblioteca de jogos **Pygame**, que pode ser utilizada com a linguagem de programação **Python**. Foi criado um jogo composto por 3 níveis de dificuldade com um sistema de pontuação e tempo, de modo a testar e desenvolver as capacidades de indivíduos autistas em reconhecer e expressar emoções básicas. Inicialmente o utilizador tem de introduzir o nome e no final do jogo, tanto o nome como a pontuação do jogador é guardada de forma a analisar o seu desenvolvimento.

5.2.1 Jogo 1

Relativamente ao primeiro jogo, foram criadas duas colunas, cada uma com 3 imagens com a respetiva emoção associada. Para além das duas colunas foram geradas um conjunto de imagens aleatórias no centro da área de jogo e o objetivo consiste em arrastar essas imagens para os lugares correspondentes. O jogo tem um tempo limite e à medida que o jogador coloca as imagens no lugar correto a sua pontuação vai aumentando. O jogo chega ao fim quando o utilizador consegue arrastar todas as imagens para o local correto ou quando o tempo chega ao fim.

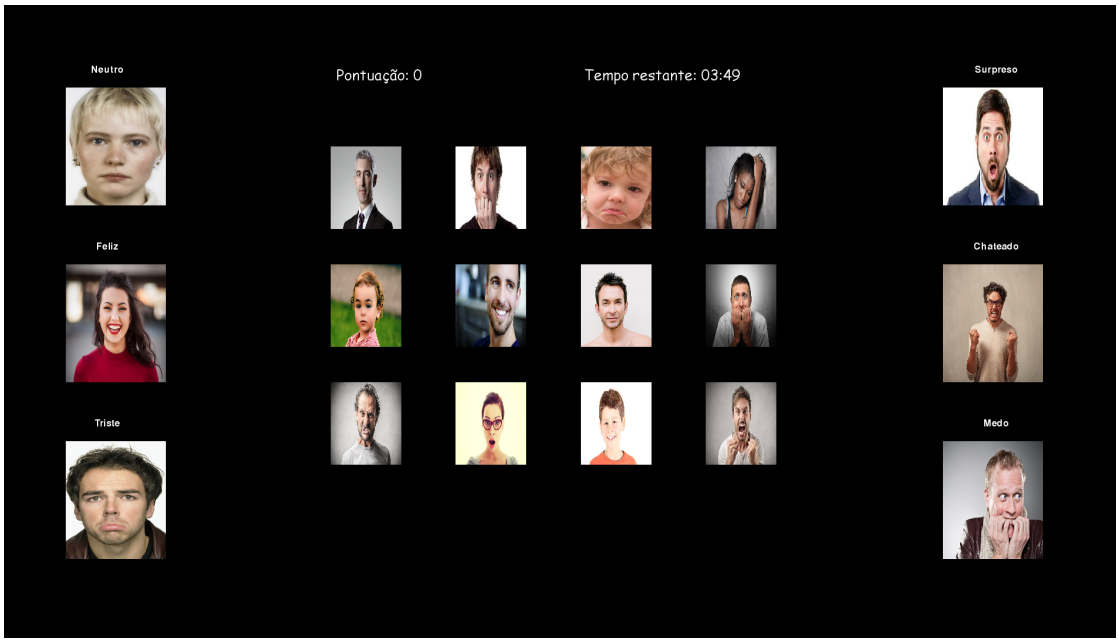


Figura 5.2: Nível 1

5.2.2 Jogo 2

No que diz respeito ao segundo jogo o principio de funcionamento e jogabilidade é idêntico ao descrito anteriormente, no entanto as duas colunas que previamente eram compostas por imagens e a emoção associada deixam de ter um estímulo visual, ou seja, as imagens desaparecem ficando apenas o nome da emoção. Assim o jogador terá que recordar quais as características de cada emoção e arrastar as imagens que se encontram no centro da área de jogo para o local correto. Mais uma vez o jogo tem um tempo limite e à medida que o jogador coloca as imagens no lugar correto a sua pontuação vai aumentando. O jogo chega ao fim quando o utilizador consegue arrastar todas as imagens para o local correto ou quando o tempo chega ao fim.

Figura 5.3: Nível 2

5.2.3 Jogo 3

Por fim, o terceiro jogo faz uso do sistema AFER desenvolvido. O objetivo será que o jogador imite a emoção apresentada durante um certo período de tempo. O sistema de reconhecimento automático de expressões faciais deteta qual a emoção expressa pelo utilizador e em caso de sucesso passa para a emoção seguinte. A área de jogo começa por ter apenas o nome da emoção e à medida que o tempo passa (caso o jogador não consiga ter sucesso) vão aparecendo imagens correspondentes a essa emoção de modo a dar um estímulo visual tornando-se, assim, mais fácil imitar a expressão pretendida.

Mais um vez o jogo tem um tempo limite e um sistema de pontuação que vai aumentando em caso de sucesso (jogador imita corretamente a emoção). O jogo chega ao fim quando o jogador consegue imitar todas as expressões pretendidas ou quando o tempo acaba.

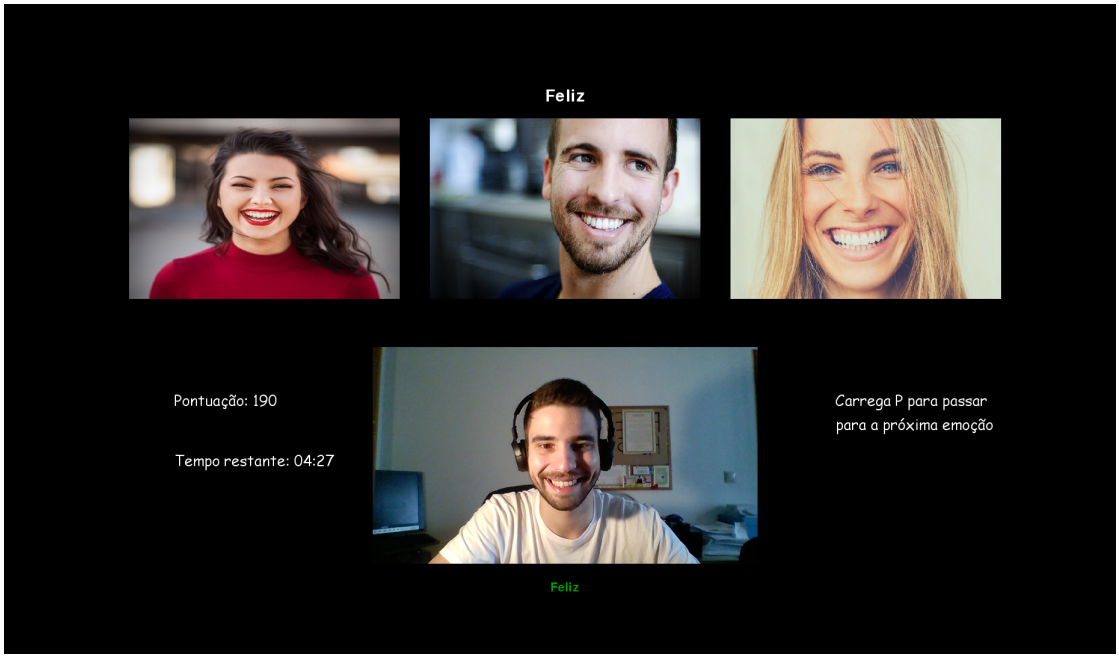


Figura 5.4: Nível 3

Todos os jogos desenvolvidos apresentam diferentes estímulos sonoros que permitem ao utilizador ter uma experiência mais imersiva.

5.3 Avaliação

De forma a obter validação relativamento ao trabalho desenvolvido foram realizados diversos testes com potenciais utilizadores na Associação Portuguesa de Pais e Amigos do Cidadão Deficiente Mental (APPACDM), bem como uma análise por parte de um especialista em neurociências do Instituto de Ciências Nucleares Aplicadas à Saúde (ICNAS). Posteriormente, o software desenvolvido foi implementado no projeto EuroAGE, cujo um dos parceiros é a Universidade de Coimbra.

5.3.1 APPACDM Coimbra

A APPACDM (Associação Portuguesa de Pais e Amigos do Cidadão Deficiente Mental) é uma instituição de solidariedade social que tem como objetivo criar condições para que pessoas com deficiência mental possam atingir a sua plenitude como ser humano, potenciando a sua individualidade e consolidando a sua participação efetiva na sociedade [1]. Em particular, a APPACDM Coimbra, no dia 17 Julho de 2019 abriu as portas a uma das suas instalações (lar residencial de Montes Claros) de modo a dar a conhecer todos os jovens e adultos que se encontram ao seu cuidado. Foi-nos concedido o privilégio de testar os jogos desenvolvidos com todos os presentes, possibilitando assim, validar o software neste contexto. Durante a tarde foram 8 os jovens/adultos que testaram os jogos acompanhados pela Dra.Carla Ribeiro que desde o início se mostrou disposta a colaborar e ajudar.

Figura 5.5: Visita APPACDM

Figura 5.6: Visita APPACDM

Figura 5.7: Visita APPACDM

No final da visita foi pedido à Dra.Carla que respondesse a um questionário de modo a avaliar o produto, tendo em conta o seu ponto de vista bem como o de todos os jovens/adultos presentes. O questionário é composto por pares de opostos relativos às propriedades do *software*. É importante referir que não existem respostas "certas"ou "erradas", a avaliação pretende analisar se o *software* atinge os objetivos propostos. De seguida são apresentados os resultados obtidos:

	1	2	3	4	5	6	7	
Desagradável	○	○	○	○	○	○	◉	Agradável
Incompreensível	○	○	○	○	○	○	◉	Compreensível
Criativo	○	◉	○	○	○	○	○	Sem criatividade
De Fácil aprendizagem	○	◉	○	○	○	○	○	De difícil aprendizagem
Valioso	○	◉	○	○	○	○	○	Sem valor
Aborrecido	○	○	○	○	○	◉	○	Excitante
Desinteressante	○	○	○	○	○	◉	○	Interessante
Imprevisível	○	○	◉	○	○	○	○	Previsível
Rápido	○	○	◉	○	○	○	○	Lento
Original	○	◉	○	○	○	○	○	Convencional
Obstrutivo	○	○	○	○	◉	○	○	Condutor
Bom	◉	○	○	○	○	○	○	Mau
Complicado	○	○	○	○	○	◉	○	Fácil
Desinteressante	○	○	○	○	○	◉	○	Atrativo
Comum	○	○	○	○	○	◉	○	Vanguardista
Incómodo	○	○	○	○	◉	○	○	Cómodo
Seguro	◉	○	○	○	○	○	○	Inseguro
Motivante	○	◉	○	○	○	○	○	Desmotivante
Atende as expectativas	○	◉	○	○	○	○	○	Não atende as expectativas
Ineficiente	○	○	○	○	○	◉	○	Eficiente
Evidente	○	◉	○	○	○	○	○	Confuso
Impraticável	○	○	○	○	○	○	◉	Prático
Organizado	◉	○	○	○	○	○	○	Desorganizado
Atraente	◉	○	○	○	○	○	○	Feio
Simpático	◉	○	○	○	○	○	○	Antipático
Conservador	○	○	○	○	○	◉	○	Inovador

Figura 5.8: Questionário realizado

A avaliação obtida foi bastante positiva, tendo atingido os objetivos propostos segundo a opinião da Dr.Carla Ribeiro. Todos os intervenientes gostaram da experiência e acima de tudo foi possível proporcionar uma tarde diferente e educativa a todos eles.

5.3.2 ICNAS

O ICNAS (Instituto de Ciências Nucleares Aplicadas à Saúde) é uma organização orgânica de investigação da Universidade de Coimbra que colabora com uma vasta rede de parceiros nacionais e internacionais em várias áreas médicas, permitindo desenvolver conhecimentos e competências para uma aplicação biomédica translacional, com grande ênfase nas neurociências e doenças crónicas [3]. No dia 18 de Julho de 2019 foi marcada

uma reunião com o Professor Doutor Miguel Castelo Branco onde foram discutidos os jogos desenvolvidos, bem como o sistema automático de reconhecimento de expressões faciais. É de salientar o interesse apresentado, essencialmente pelo 3º jogo, por parte do Prof. Doutor Miguel Castelo Branco, cujo principal objetivo é reproduzir diversas expressões faciais. Foi discutida a possibilidade de fazer um ensaio clínico, onde, numa primeira fase, o ideal seria conseguir colocar o programa desenvolvido em diversas associações de forma a recolher dados de forma independente. A avaliação obtida por parte dos intervenientes foi bastante positiva e é esperada uma colaboração entre o ISR e o ICNAS de forma a dar continuidade ao trabalho desenvolvido.

5.3.3 Integração no Projeto EuroAGE

O projeto EuroAGE permite promover o envelhecimento ativo nas vertentes de atividade física, cognitiva e sócio emocional, tendo como principal objetivo melhorar a qualidade de vida dos utentes e aumentar a esperança de vida saudável. Um dos seus parceiros é a Universidade de Coimbra que está a desenvolver uma solução integrada de uma casa inteligente com robôs móveis que permitem verificar condições de saúde bem como estimular física e cognitivamente os pacientes [4].

O software de reconhecimento automático de expressões faciais desenvolvido foi integrado nos robôs de modo a verificar o estado emocional do idoso de forma a comportar-se de acordo com a situação.

6 Conclusão e Trabalho Futuro

O principal objetivo da presente dissertação foi desenvolver um ambiente capaz de ensinar indivíduos autistas a reconhecer e expressar expressões faciais, para tal foram desenvolvidos diversos jogos, bem como um sistema de reconhecimento automático de expressões faciais baseado numa rede neuronal convolucional. Com base nos dados obtidos foi possível alterar a codificação original de uma base de dados com imagens estáticas estabelecendo uma correspondência entre emoções e unidades de ação.

Após diversas melhorias, o modelo desenvolvido obteve uma precisão na ordem dos 76% no conjunto de dados de treino. Ao aplicar diversas transformações ao conjunto de dados verificou-se uma melhoria substancial, o que levou a um aumento no desempenho do modelo. Embora apresente algumas limitações, essencialmente em termos de iluminação (a face da pessoa deve estar bem iluminada), o modelo é capaz de reconhecer as 6 expressões faciais através de captura de vídeo.

Os jogos foram analisados, tanto pela Dra. Carla Ribeiro, como pelo Prof. Doutor Miguel Castelo Branco e em ambos os casos os comentários recebidos foram bastante positivos, reconhecendo potencial no trabalho desenvolvido.

Futuramente será necessário definir métricas de modo a realizar um ensaio clínico onde o ideal seria começar por colocar os jogos elaborados em diversas associações de modo a recolher dados de forma independente.

Resultado do trabalho realizado, foi publicado um artigo na conferencia internacional SMC2019 (*Systems, Man, and Cybernetics*) que será apresentado em Outubro de 2019 em Bari, Itália.

Appendices

Classification of FACS-Action Units with CNN Trained from Emotion Labelled Data Sets

Pedro Carvalho Gerardo[1] Paulo Menezes[2]

Abstract— **This paper explores the adaptation of a convolutional neural network (CNN) designed for emotion classification to the extraction of human expressions action units. An adaptation of the network structure transforms a single label into a multi-label classifier which supports the simultaneous recognition of multiple action units that compose human expressions, according to Ekman's FACS. The dataset used for this work was FER-2013 that includes exemplars of seven basic expressions (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral). In order to enhance the quality of the results, the dataset was augmented with random perturbations from a wide set including: translation, scale and horizontal flip. The results obtained demonstrate that it is possible to train a CNN for multi-label expression classification from an emotion-labelled dataset.**

Keywords: FACS, Emotion recognition, Convolutional Neural Networks

I. INTRODUCTION

Facial expression recognition has been an important subject of research over the last 10 years, with extensive application areas like avatar animation, neuromarketing and social robots. It is not a simple problem, even for machine learning methods, since people can significantly vary the way they display their expressions. In fact, in different images of the same person expressing some given emotion, we can observe a substantial variability due to the many factors that may affect it. Health state, tiredness, social context, among many others, will modulate muscle activation and tissue deformation characteristics, that will result in a fluctuation of face's shapes for a given emotion. Naturally images of faces captured in different circumstances will also be influenced by other external factors such as illumination, background and even relative position with respect to the camera. Despite this, humans are particularly good at recognizing facial expressions regardless of the factors described above. This can serve as a motivation to explore artificial neural networks for the same purpose.

Emotions are always present in humans' daily lives and, not only influence the way we perceive and react to external stimuli, but also play a very important role in the communication between people. The recognition of someone's emotions enables us to understand what has not been said, contextualize what was said, and even lead to adjustments in the way we respond or behave. This capability can also be seen as a sort of predictive aid to infer which will be the next behaviours or actions of another person. Conversely, the person expressing some emotion also expects his/her interlocutor to acknowledge them and respond in accordance, otherwise empathy is broken and communication deteriorates.

There are six consensual basic emotion-related expressions in any culture [4], namely *Anger*, *Disgust*, *Fear*, *Happiness*, *Sadness*, *Surprise*, which have being used by the majority of Facial Expression Recognition (FER) systems. Typically, FER systems are based on a three-stage structure, starting with facial detection, feature extraction, and expression recognition. The facial detection stage is one of the most explored and can be considered as an exhausted topic. The Viola and Jones classifier [16] is well known and widely used, although newly proposed methods seem to perform better at the expense of some additional computational power required [5]. Most of the problems lie in the dataset construction, as they are created under controlled environments (positions, lightning, no occlusions), thus putting in question the robustness of the system in terms of invariance.

In recent years, through technological innovations and the development of deep learning models, it has been possible to extract increasingly complex data from large datasets. More robust architectures appeared, among them the *Convolutional Neural Networks*, that brought significant results to image-based classification fields. In this respect, there are some architectures that deserve to be highlighted, given their high performances in the image classification problem, such as *AlexNet* [10] from 2012 and a more recent, *GoogLeNet* [14] with the concept of "Network inside Network".

Several works have been conducted to explore these architectures to recognize emotions from face images, without paying attention to the details that compose the related face expressions.

This is where this work differs from the previous. Although starting with emotion related expressions, the goal is to extract the so called facial *action units*. Based on the work of Swedish anatomist Carl-Herman Hjortsjö, Paul Ekman and Wallace Friesen developed a method to objectively code facial behavior. Their approach, named Facial Action Coding System (FACS), represents a standardized classification system of facial expressions based on anatomic features. They describe any occurrence of facial expressions as combinations of elementary components called Action Units (AU).

This work was supported by Institute of Systems and Robotics (ISR) University of Coimbra via OE - national funds of FCT/MCTES (PIDDAC) under project UID/EEA/00048/2019.

[1]Pedro Carvalho Gerardo is with Institute of Systems and Robotics, University of Coimbra pedro.gerardo@isr.uc.pt

[2]Paulo Menezes is with the Institute of Systems and Robotics and with the Department of Electrical and Computer Engineering, University of Coimbra, Polo II, 3030-290 Coimbra, Portugal PauloMenezes@isr.uc.pt

This paper presented our work inspired on FACS, where a Convolutional Neural Network was developed for facial expression recognition based on mixtures of Ekman's action units.

In the next section discusses some of the related works about CNNs and emotion recognition, followed by the description of the proposed model, and the pre-processing of the dataset that will be analyzed. Then the obtained results will be presented and finally the conclusions and future works.

II. Related Work

Diversity of neural networks. Convolutional Neural Networks are commonly used to extract certain characteristics, and composed of a set convolution filters, feature reduction (e.g. maxpooling) and of neurons distributed in layers. The last layers, the fully-connected layers, contain most of the parameters of a CNN. In particular, the VGG16 [13] architecture contains close to 90% of its parameters in the last layers. More recent architectures, such as Inception V3 [15], have reduced the number of parameters in the last layers, including a Global Average Pooling operation that reduces each feature map to a scalar value, considering only the average of all elements. This allows the network to be able to extract global resources from the input data. Modern architectures, such as Xception [2], make use of residual modules [7] and depth-wise separable convolutions [8]. Depth-wise separable convolutions allow to reduce the number of parameters by separating extraction processes and combining characteristics within a convolutional layer.

Deep learning for facial expression recognition. In the past few years applying deep learning techniques to FER has been a subject frequently addressed. Traditionally, FER systems used handcrafted features or shallow learning, such as local binary patterns (LBP) [12], non-negative matrix factorization (NMF) [17] and sparse learning [18]. However, with the emergence of competitions, such as, FER2013 [6] and Emotion Recognition in the Wild (EmotiW) [3], [1] in 2013, begin to appear sufficient training data, in real-world conditions, which promote the transition of FER systems from lab-controlled environments to "in-the-wild" settings. Along with this, due to the increasing chip processing abilities and the emerge of new well-designed network architectures, studies in various fields started to use deep learning methods which exceeded past results [10], [13],[7].

III. A Model For Action Units Classification

In order to build the classification model, several architectures were tested (LeNet-5, AlexNet, VGG16, ResNet50). With the obtained results we opted to base it on the VGG16 architecture since it was the one that presented the best results.

The classic VGG architecture, introduced in 2014, offers a deeper yet simpler variant of the convolutional structures. The remarkable thing about this architecture is that instead of having many hyper parameters, it always uses 3×3 filters with stride of 1 in the convolution layer and uses *SAME* padding in pooling layers 2×2 with stride of 2 [13].

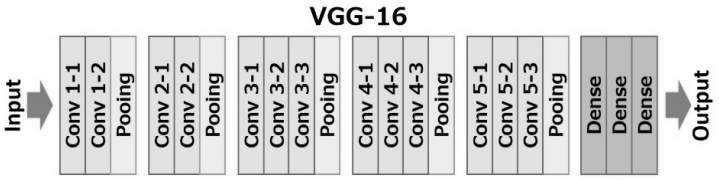


Fig. 1. Classic VGG16 architecture

In this workm and for the training, the input of the neural network was changed to accept 48×48 pixel grayscale images. These images are passed through several convolutional layers, where 3×3 filters are used. The convolutional stride was set to 1 pixel. Max-pooling is performed over 2×2 pixel window, with stride 2.

The stack of convolutional layers is followed by three fully connected (FC) layers, where in the case of the present work, these last three layers have been altered in order to address the problem in hand. The first one has 512 channels, the second one 128 and the final layer contain 14 (one for each action unit analyzed). In the last layer, the activation function used was the sigmoid function. This is the most adequate choice, since this is a problem where multiple labels may be assigned to each instance, or in other words, there is no constraints on how many of the classes an instance can be assigned to.

All the layers between the entrance and the FC layers, that is, the hidden layers, are equipped with the activation function Rectified Linear Units (ReLU).

Summarizing:

- Input Size: 48×48 (grayscale images)
- Receptive Field: 3×3
- Convolutional Stride: 1 pixel
- Padding: 1
- Max Pooling: 2×2 with stride of 2 pixel
- First FC Layer: 512 units
- Second FC Layer: 128 units
- Final FC Layer: Sigmoid classification with 14 units
- Activation Function: ReLU

As for the optimization algorithm, Adaptive Moment Estimation (ADAM) was chosen. ADAM computes adaptive learning rates for each parameter and compares favorably to other adaptive learning-method algorithms, as it converges very fast and the learning speed of the model is quite fast and efficient [9].

IV. Learning AUs from Emotion-Labelled Datasets

Over time, a wide variety of datasets have emerged for expression recognition, where a large number of images are labelled with emotions. The great majority presents a wide diversity of individuals with both posed and unposed faces. However the case is different when we talk about action units recognition. Here, although there are some datasets

labeled with action units, such as the Extended Cohn-Kanade dataset [11], they present little data and/or are created under controlled environments, which makes them less adequate for training robust classifiers with the adequate invariant support for the features of interest.

The choice to have an AU-based solution is twofold: first it allows to extract the AU components that are present on the expression under analysis, and secondly it still supports the recognition of expressed emotions. In fact, FACS does not interpret the meaning of expressions but allows us to recognize emotions based on the combination of AUs, or to state which AUs are dominantly present for each emotion. The following table shows the combination of action units related to the six basic emotions.

Emotion	AU
Anger	4+5+7+23
Disgust	9+15+16
Fear	1+2+4+5+7+20+26
Happy	6+12
Sad	1+4+15
Surprise	1+2+5+26
Neutral	None

TABLE I: Emotions and respective AUs

As we can see through the analysis of Table I it will be necessary to analyze 13 action units (1, 2, 4, 5, 6, 7, 9, 12, 15, 16, 20, 23, 26). The neutral expression does not present any action unit so it will be necessary to create a new class only for this expression, unlike any other. It is important to say that there may be more action units associated with each of the expressions but, in this approach we consider only those that are always present independently of the intensity of expression.

For training purposes, in this implementation, the original emotional labels were replaced by the corresponding combinations of action units, in accordance to Table I. This was done using the FER2013 dataset, which consists of 28,000 labelled images in the training set, 3,500 labelled images in the validation set, and 3,500 images in the test set. The images in this dataset are labelled as one of seven emotions (Angry, Disgust, Fear, Happy, Sad, Surprise, Neutral). It contains images of both posed and unposed faces, which are 48x48 pixels grayscale, and it was created by gathering the results of a ”Google” image search of each emotion and synonyms of the emotions. In order to enhance the quality of the results, the dataset was augmented with random perturbations from a wide set including: translation, scale and horizontal flip. The final augmentation version has 15,044,814 trainable parameters. The dataset labels were originally coded as follows:

Emotion	Coding
Anger	[1,0,0,0,0,0,0]
Disgust	[0,1,0,0,0,0,0]
Fear	[0,0,1,0,0,0,0]
Happy	[0,0,0,1,0,0,0]
Sad	[0,0,0,0,1,0,0]
Surprise	[0,0,0,0,0,1,0]
Neutral	[0,0,0,0,0,0,1]

TABLE II: Emotions Coding

Now, having established the relation between action units and emotions, it was necessary to change the initial coding so that the model can evaluate each of the expressions through the action units instead of the emotions. For that, new labels were assigned based on the following new codebook:

Emotion	Coding
Anger	[0,0,1,1,0,1,0,0,0,0,0,1,0,0]
Disgust	[0,0,0,0,0,0,1,0,1,1,0,0,0,0]
Fear	[1,1,1,1,0,1,0,0,0,0,1,0,0,0]
Happy	[0,0,0,0,1,0,0,1,0,0,0,0,0,0]
Sad	[1,0,1,0,0,0,0,0,1,0,0,0,0,0]
Surprise	[1,1,0,1,0,0,0,0,0,0,0,0,1,0]
Neutral	[0,0,0,0,0,0,0,0,0,0,0,0,0,1]

TABLE III: AU Coding

As we can see, each element of the vector represents an AU. The images will receive these labels instead of those present in Table II and both these parameters (images and labels) will be the input of the network.

V. Results and Discussion

In this section will be discussed the results from the experiments with static-images. The model was trained over 400 iterations, with ADAM optimization algorithm to update network weights iterative based in training data.

After the training of the neural network, the model presented an accuracy rate of 91% in training and 73% in the test set (images not used in the training phase).

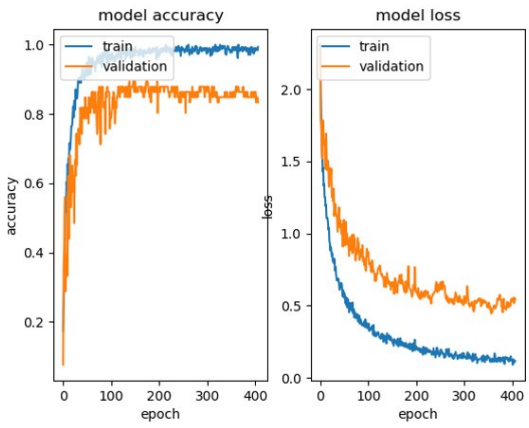


Fig. 2. Model accuracy vs Model loss

The table below show the accuracy of each action unit and the neutral expression after the training of the network.

Action Unit	Precision
AU1	78,44%
AU2	71,14%
AU4	81,24%
AU5	77,04%
AU6	87,22%
AU7	64,17%
AU9	69,77%
AU12	87,22%
AU15	56,34%
AU16	69,77%
AU20	45,82%
AU23	56,45%
AU26	73,68%
Neutral	63,36%

TABLE IV: Model Results

As results some images will be presented, in order to demonstrate the operation of the model.

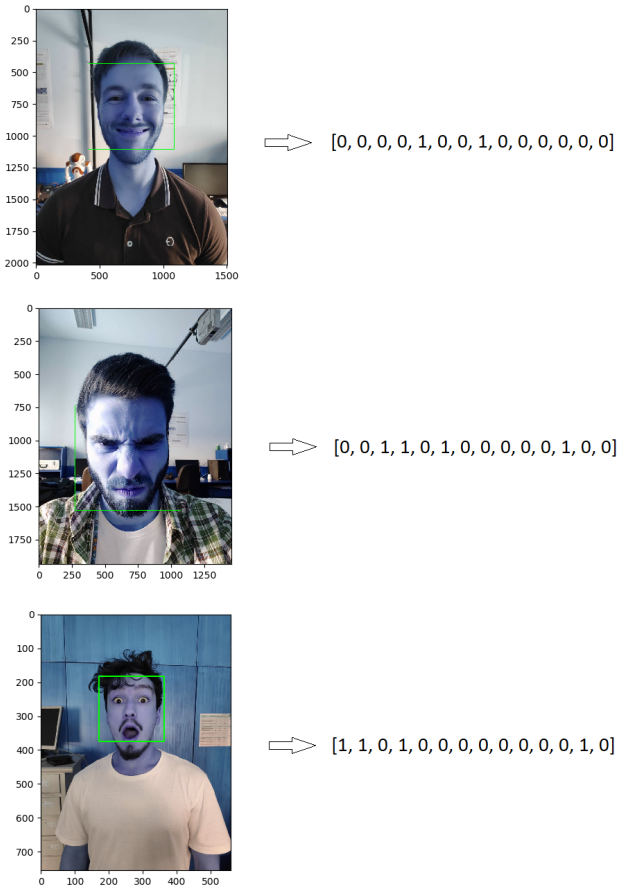


Fig. 3. Sample images and classifier outputs

After analyzing the output of the model for each of the images presented in Fig. 3 we can verify that the classifier correctly identifies every action unit presented. In the first image we have the action units 6 and 12, which together correspond to Happiness, in the second image we have the action units 4, 5, 7 and 23 that correspond to Anger and finally in the last image the model identifies the action units 1, 2, 5 and 26 that correspond to Surprise .

VI. Conclusions and Future Work

In this paper, we presented a model for facial expressions recognition from the analysis of action units present on the face of an individual. A convolutional neural network was developed for the problem in question from static images using the FER2013 dataset. After some improvements, such as the augmentation performed on the dataset with several transformations our model obtained 73% of accuracy in the test set. The results are remarkable as it shows that, although it is difficult to obtain an appropriate AU-labeled data set, it is possible to use a cross labeling method to train a AU-recognizer network from an emotion-labeled database. The proposed model has been implemented in a set of serious games and the ultimate goal is to create an environment that can be used to teach autistic individuals to express/recognize basic emotions.

References

[1] Dhall Abhinav, Goecke Roland, Ghosh Shreya, Joshi Jyoti, Hoey Jesse, and Gedeon Tom. From individual to group-level emotion recognition: Emotiw 5.0. *ACM ICMI 2017*, 2017.

[2] François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1251–1258, 2017.

[3] Abhinav Dhall, Roland Goecke, Jyoti Joshi, Jesse Hoey, and Tom Gedeon. Emotiw 2016: Video and group-level emotion recognition challenges. In *Proceedings of the 18th ACM International Conference on Multimodal Interaction*, pages 427–432. ACM, 2016.

[4] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124, 1971.

[5] Sachin Sudhakar Farfade, Mohammad J Saberian, and Li-Jia Li. Multi-view face detection using deep convolutional neural networks. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pages 643–650. ACM, 2015.

[6] Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International Conference on Neural Information Processing*, pages 117–124. Springer, 2013.

[7] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[8] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[9] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[10] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[11] Cohn J. F. Kanade T. Saragih J. Ambadar Z. Matthews I Lucey, P. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[12] Caifeng Shan, Shaogang Gong, and Peter W McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and vision Computing*, 27(6):803–816, 2009.

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[15] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.

[16] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[17] Ruicong Zhi, Markus Flierl, Qiuqi Ruan, and W Bastiaan Kleijn. Graph-preserving sparse nonnegative matrix factorization with application to facial expression recognition. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(1):38–52, 2010.

[18] Lin Zhong, Qingshan Liu, Peng Yang, Bo Liu, Junzhou Huang, and Dimitris N Metaxas. Learning active facial patches for expression analysis. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2562–2569. IEEE, 2012.

7 Bibliografia

[1] Appacdm coimbra. http://www.appacdmcoimbra.pt/ (Acedido em 18/07/2019).

[2] Convolutional neural networks for visual recognition. https://cs231n.github.io/convolutional-networks/ (Acedido em 12/09/2018).

[3] Instituto de ciências nucleares aplicadas à saúde. https://www.uc.pt/icnas (Acedido em 19/07/2019).

[4] Projeto euroage. https://euroage.eu/pt/home/ (Acedido em 27/06/2019).

[5] Introduction to optimizers, Dec 2018. https://blog.algorithmia.com/introduction-to-optimizers/ (Acedido em 22/12/2018).

[6] Facial expression analysis: The complete pocket guide. *iMotions*, Feb 2019. https://imotions.com/blog/facial-expression-analysis/ (Acedido em 05/17/2019).

[7] Ralph Adolphs. Recognizing emotion from facial expressions: psychological and neurological mechanisms. *Behavioral and cognitive neuroscience reviews*, 1(1):21–62, 2002.

[8] Thomaz Almeida. Uma metodologia de reconhecimento de caracteres manuscritos utilizando redes neurais embarcadas. 06 2014.

[9] Flavio HD Araujo, Allan C Carneiro, Romuere RV Silva, Fatima NS Medeiros, and Daniela M Ushizimau. Redes neurais convolucionais com tensorflow: Teoria e pratica. *Sociedade Brasileira de Computacao. III Escola Regional de Informatica do Piaui*, 1:382–406, 2017.

[10] Vinay Bettadapura. Face expression recognition and analysis: The state of the art. *CoRR*, abs/1203.6722, 2012.

[11] Vicki Bruce and Andy Young. Understanding face recognition. *British journal of psychology*, 77(3):305–327, 1986.

[12] Tianyuan Chang, Guihua Wen, Yang Hu, and JiaJiong Ma. Facial expression recognition based on complexity perception classification algorithm. 02 2018.

[13] Charles Darwin and Phillip Prodger. The expression of the emotions in man and animals. 1998.

[14] Siddharth Das. Cnn architectures: Lenet, alexnet, vgg, googlenet, resnet and more, Nov 2017. https://medium.com/@sidereal/cnns-architectures-lenet-alexnet-vgg-googlenet-resnet-and-more-666091488df5 (Acedido em 05/18/2019).

[15] Adit Deshpande. A beginner's guide to understanding convolutional neural networks part 2. https://adeshpande3.github.io/A-Beginner's-Guide-To-Understanding-Convolutional-Neural-Networks-Part-2/ (Acedido em 05/18/2019).

[16] Paul Ekman. *Emotions revealed: Recognizing faces and feelings to improve communication and emotional life*. Macmillan, 2007.

[17] Paul Ekman and Wallace V Friesen. Constants across cultures in the face and emotion. *Journal of personality and social psychology*, 17(2):124, 1971.

[18] Paul Ekman and Karl G Heider. The universality of a contempt expression: A replication. *Motivation and emotion*, 12(3):303–308, 1988.

[19] Arthur Emídio Teixeira Ferreira. Estimação do ângulo de direção por vídeo para veículos autônomos utilizando redes neurais convolucionais multicanais. 2017.

[20] Hao Gao. A walk-through of alexnet, Aug 2017. https://medium.com/@smallfishbigsea/a-walk-through-of-alexnet-6cbd137a5637 (Acedido em 05/18/2019).

[21] Bengio Yoshua Goodfellow, Ian and Aaron Courville. Deep learning. 2016.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[23] A. Garg L.S. Chen I. Cohen, N. Sebe and T.S. Huang. Facial expression recognition from video sequences: Temporal and static modeling, 2003. https://pdfs.semanticscholar.org/dcc8/ae510349715df09c3f24d08c42ce6e7def2c.pdf (Acedido em 06/14/2019).

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[25] Seyed Lajevardi, Mehdi and Zahir Hussain. A novel gabor filter selection based on spectral difference and minimum error rate for facial expression recognition. pages 137–140, 12 2010.

[26] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *arXiv preprint arXiv:1804.08348*, 2018.

[27] Aguiar Edilson de Souza Alberto F. de Lopes, André Teixeira and Thiago Oliveira-Santos. Facial expression recognition with convolutional neural networks: Coping with few data and the training sample order, 2017. https://www.sciencedirect.com/science/article/abs/pii/S0031320316301753 (Acedido em 07/11/2018).

[28] Stéphanie Mader, Stéphane Natkin, and Guillaume Levieux. How to analyse therapeutic games: the player/game/therapy model. In *International Conference on Entertainment Computing*, pages 193–206. Springer, 2012.

[29] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5:115–133, 1943.

[30] Aylien Noel Bambrick. Support vector machines: A simple explanation, 2016. https://www.kdnuggets.com/2016/07/support-vector-machines-simple-explanation.htm (Acedido em 06/14/2019).

[31] Thomas J Palmeri and M Tarr. Visual object perception and long-term memory. *Visual memory*, pages 163–207, 08 2008.

[32] M. Pantic, M. Valstar, R. Rademaker, and L. Maat. Web-based database for facial expression analysis. pages 5 pp.–, July 2005.

[33] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, et al. Deep face recognition. 1(3):6, 2015.

[34] Vojtech Pavlovsky. Introduction to convolutional neural networks. https://www.vaetas.cz/posts/intro-convolutional-neural-networks/ (Acedido em 05/17/2019).

[35] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[36] Fernando Schimidt and Saulo Rodrigues. Redução de variáveis de entrada de redes neuronais artificiais a partir de dados de análise de componentes principais na modelagem de oxigênio dissolvido. *Química Nova*, Apr 2016.

[37] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[38] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[39] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[40] Xiaogang Wang. Deep learning in object recognition, detection, and segmentation. *Foundations and Trends® in Signal Processing*, 8:217–382, 2016.

[41] Lei Wu, Chao Ma, and E Weinan. How sgd selects the global minima in over-parameterized learning: A dynamical stability perspective. In *Advances in Neural Information Processing Systems*, pages 8279–8288, 2018.

[42] David J Yang, Ming-Hsuan; Kriegman and Narendra Ahuja. Detecting faces in images: A survey. *IEEE Transactions on pattern analysis and machine intelligence*, 24(1):34–58, 2002.