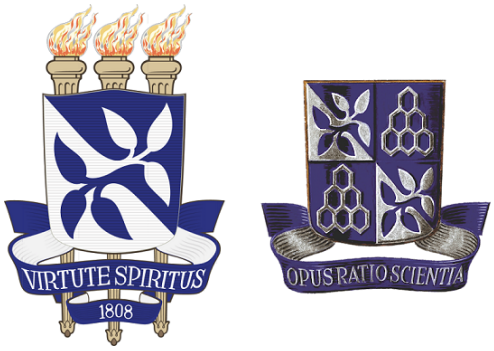

Universidade Federal da Bahia
Escola Politécnica
Departamento de Engenharia Elétrica
Programa de Pós-Graduação em Engenharia Elétrica

Detecção de Novidades Aplicada ao Reconhecimento de Expressões Faciais em Fluxo de Vídeo

Autor: Márcio da Silva Pereira Bove
Orientador: Jés de Jesus Fiais Cerqueira
Coorientador: Eduardo Furtado de Simas Filho

Salvador - BA
Março de 2021

Márcio da Silva Pereira Bove

Detecção de Novidades Aplicada ao Reconhecimento de Expressões Faciais em Fluxo de Vídeo

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal da Bahia para a obtenção do Título de Mestre em Engenharia Elétrica.

Universidade Federal da Bahia – UFBA

Departamento de Engenharia Elétrica

Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Jés de Jesus Fiais Cerqueira

Coorientador: Eduardo Furtado de Simas Filho

Salvador - BA

Março de 2021

B783 Bove, Márcio da Silva Pereira.
Detecção de novidades aplicada ao reconhecimento de expressões faciais em fluxo de vídeo / Márcio da Silva Pereira Bove. – Salvador, 2021.
118 p: il. color.

Orientador: Prof. Dr. Jés de Jesus Fiais Cerqueira.
Coorientador: Prof. Dr. Eduardo Furtado de Simas Filho.

Dissertação (mestrado) – Universidade Federal da Bahia. Escola Politécnica, 2021.

1. Redes neurais artificiais. 2. Vídeo – fluxo. 3. Expressão facial - reconhecimento. I. Cerqueira, Jés de Jesus Fiais. II. Simas Filho, Eduardo Furtado de. III. Universidade Federal da Bahia. IV. Título.

CDD: 006.3

Márcio da Silva Pereira Bove

Detecção de Novidades Aplicada ao Reconhecimento de Expressões Faciais em Fluxo de Vídeo

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal da Bahia para a obtenção do Título de Mestre em Engenharia Elétrica.

Salvador - BA, 12 de Março de 2021:

Jés de Jesus Fiais Cerqueira
Orientador - UFBA

Eduardo Furtado de Simas Filho
Coorientador - UFBA

Antonio Carlos Lopes Fernandes Junior
Avaliador - UFBA

Allan Edgar Silva Freitas
Avaliador - IFBA

Salvador - BA

Março de 2021

Agradecimentos

A Deus e seu filho Amado Senhor Jesus Cristo pelo seu amor, amizade, cuidado e condução na produção desta pesquisa.

A Deus pela vida da minha querida e amada esposa Jamile Olivia Bove Silva Pereira que me auxilia sempre com muito amor, carinho e cuidado.

A Deus pela vida da minha pequena estrelinha Esther Abraão Pereira Bove, que encheu a minha vida de alegria e mudou tudo em mim.

A Deus pela vida dos meus amados pais Isídio Pereira Pinto e Quitéria Rodrigues da Silva, que sempre me educaram com muito amor, carinho e cuidado.

A Deus pela vida das minhas amadas irmãs Islaine da Silva Pereira e Viviam Rodrigues da Silva, que sempre estão ao meu lado me apoiando com muito amor e carinho.

A Deus pela vida do meu orientador Professor Jés de Jesus Fiais Cerqueira, pela sua competência, suas orientações e cuidado na condução desta pesquisa.

A Deus pela vida do meu coorientador Professor Eduardo Furtado de Simas Filho, pela suas valiosas orientações e cuidado na condução desta pesquisa.

A Deus pela vida dos Professores Jés de Jesus Fiais Cerqueira, André Gustavo Scolari Conceição, Tito Luís Maia Santos, José Mário Araújo e Augusto Cesar Pinto Loureiro da Costa pelos valiosos ensinamentos durante o período do mestrado.

"Eu imagino um mundo
onde a inteligência artificial
nos permitirá ser mais
produtivos, viver mais, e ter
energia mais limpa."
(Fei-Fei Li)
professor de ciência da computação na Universidade de Stanford

Resumo

Este trabalho investiga as redes Perceptron de Múltiplas Camadas (MLP) e Função de Base Radial (RBF) na tarefa de Detecção de Novidade (DN) aplicada ao reconhecimento de expressões faciais em fluxo de vídeo. O conjunto de dados de vídeo utilizado é produzido por atores profissionais em estúdio com os estados afetivos básicos do rosto humano. Os algoritmos *Viola-Jones*, *Kanade-Lucas-Tomasi* (KLT) e Análise de Componentes Principais (PCA) são usados na fase de pré-processamento para extração de atributos da face. Os resultados avaliam o desempenho das redes MLP e RBF na tarefa DN, usando novas expressões faciais compatíveis com os estados afetivos utilizados na fase de treinamento e também examinam a performance das redes em DN usando as faces de atores nunca antes vistos pelas redes. Neste processo, as redes MLP e RBF apresentam uma acurácia de 99,8% e 99,9% para tarefa de classificação, 79,8% e 85,2% para DN com dados semelhantes aos dados usados na fase de treinamento, por último 98,5% e 98,9% para DN com dados totalmente novos. Assim, esta pesquisa reúne métodos e técnicas aplicados na DN utilizando Redes Neurais Artificiais (RNA), visando a produção de sistemas interativos de cognição no campo da computação afetiva, baseados em técnicas de Inteligência Artificial (IA) e Visão Computacional.

Palavras-chaves: Detecção de Novidade. Redes Neurais Artificiais. Viola-Jones. Kanade-Lucas-Tomasi. Análise de Componentes Principais.

Abstract

This work investigates the Multilayer Perceptron (MLP) and Radial Base Function (RBF) networks in the Novelty Detection (DN) task applied to the recognition of facial expressions in video stream. The video data set used is produced by professional actors in the studio with the basic affective states of the human face. The Viola-Jones, Kanade-Lucas-Tomasi (KLT) and Principal Component Analysis (PCA) algorithms are used in the pre-processing phase to extract features from the face. The results evaluate the performance of the MLP and RBF networks in the ND task, using new facial expressions compatible with the affective states used in the training phase and also examine the performance of the networks in DN using the faces of actors never before seen by the networks. In this process, the MLP and RBF networks have an accuracy of 99,8% and 99,9% for classification task, 79,8% and 85,2% for ND with data similar to the data used in the training phase, lastly 98,5% and 98,9% for ND with totally new data. Thus, this research brings together methods and techniques applied in ND using Artificial Neural Networks (ANN) aiming at the production of interactive cognition systems in the field of affective computing, based on techniques of Artificial Intelligence (AI) and Computer Vision.

Keywords: Novelty Detection. Artificial Neural Networks. Viola-Jones. Kanade-Lucas-Tomasi. Principal Component Analysis.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

AUs	*Actios Units*
AUC	*Area Under the Curve*
ANN	*Artificial Neural Networks*
ACC	Acurácia de Classificação
AdaBoost	*Adaptive Boosting*
CNN	*Convolution Neural Networks*
DN	Detecção de Novidades
TDN	Taxa de Detecção de Novidades
FACS	*Facial Action Coding System*
FPS	*Frames Per Second*
FER	*Facial Expression Recognition*
HRI	*Human-Robot Interaction*
IA	*Artificial Intelligence*
KLT	*Kanade-Lucas-Tomasi*
MLP	*Multilayer Perceptron*
MSE	*Mean Square Error*
ND	*Novelty Detection*
NCC	*Normalized Cross Correlation*
NSSD	*Normalized Sum of Squared Differences*
PCA	*Principal Component Analysis*
RNA	*Rede Neural Artificial*
ROC	*Receiver Operating Characteristic*
RBF	*Radial Basis Function*

RAVDESS	*Ryerson Audio-Visual Dataset of Emotional Speech and Song*
SOM	*Self-Organized Map*
SAR	*Socially Assistive Robotics*
SVD	*Singular Value Decomposition*
TEA	*Transtorno do Espectro do Autista*
TA	*Total de Acertos*
TE	*Total de Exemplos*
TF	*Total de Frames*
TFN	*Total de Frames Novos*
VJ	*Viola-Jones*

Lista de símbolos

$ii(x, y)$	Imagem integral
h_t	Classificador fraco
$C(x_i)$	Classificador forte
f_t	Característica *Haar*
p_t	Paridade de direção
θ_t	Limite de classificação
α_t	Peso atribuído ao classificador fraco
R_t	Área da face atual em quadro de vídeo
R_{t-1}	Área da face anterior em quadro de vídeo
C_t	Centro de posição atual dos atributos em quadro de vídeo
C_{t-1}	Centro de posição anterior dos atributos em quadro de vídeo
f_t	Pontos característicos atuais em quadro de vídeo
f_{t-1}	Pontos característicos anteriores em quadro de vídeo
$\mathbf{X}$	Matriz para representação de uma imagem
$\mathbf{N}$	Matriz para representação de uma imagem normalizada
$\mathbf{M}$	Matriz de média amostral
$\mathbf{C}$	Matriz de covariância
$\mathbf{v}$	Vetor singular à direita
$\mathbf{Y}$	Componente principal da imagem
$\mathbf{A}$	Matriz de entrada (m x n)
$\mathbf{U}$	Matriz ortogonal (m x m)
$\mathbf{S}$	Matriz diagonal (m x m)
$\mathbf{V}$	Matriz ortogonal (n x n)

$\mathbf{I}$	Matriz identidade
$\mathbf{x}$	Vetor de atributos
ϕ	Conjunto de neurônios da camada oculta
ϕ_i	Função de ativação
$\mathbf{y}$	Conjunto de neurônios da camada de saída
y_k	Neurônio da camada de saída
w_{ln}	Peso sináptico conectado a camada de entrada e oculta
w_{kl}	Peso sináptico conectado a camada oculta e saída
$v_i(n)$	Campo local induzido do neurônio
γ_i	Constante de ajuste do raio da função de ativação da rede MLP
$\mathbf{c}$	Centro dos dados observados da rede RBF
σ_i	Desvio padrão de ajuste do raio da função de ativação da rede RBF
$\| \, . \, \|$	Distância euclidiana
$softmax(y_i)$	Saída da rede neural na forma probabilística
δ_i	Limiar para detecção de novidades
$z(y_i)$	Saída da rede neural para detecção de novidades

Sumário

1 Introdução

A Detecção de Novidades (DN) visa a identificação de situações novas ou desconhecidas que diferem do modelo aprendido como padrão normal, sendo empregada na solução de problemas que possuem uma quantidade expressiva de exemplos normais para treinamento e dispõe de uma quantidade insuficiente ou inexistente de dados que descrevem as condições anormais, caracterizando-se como uma tarefa desafiadora e complexa (PIMENTEL et al., 2014; DOMINGUES et al., 2018; FARIA et al., 2016; Ouafae et al., 2020; Amorim et al., 2019).

A detecção de novos eventos pode ser comparada a um classificador que produz resultados para padrões normais e outro para padrões desconhecidos, onde uma descrição da condição de normalidade é aprendida ajustando um modelo ao conjunto de exemplos normais e padrões previamente desconhecidos são testados comparando a sua pontuação de novidade com algum limite de decisão (SAMEER; MARKOU, 2004).

Abordagens baseadas em diferentes categorias como probabilidade, reconstrução, domínio, teoria da informação e distância podem ser utilizadas para DN (PIMENTEL et al., 2014; Ouafae et al., 2020), de modo que estudos no âmbito da DN para áreas como monitoramento industrial, redes de sensores, robótica, processamento de sinais, visão computacional, reconhecimento de padrões, mineração de texto, segurança da informação, diagnóstico e supervisão médica tem contribuído para o desenvolvimento de sistemas inteligentes (PIMENTEL et al., 2014; Oliveira, Moisés A. et al., 2020).

No campo da computação afetiva, os sinais verbais e não verbais naturais das emoções, cognições, percepções e comportamentos humanos são recursos de dados que viabilizam a produção de sistemas de interação eficientes entre o homem e a máquina, onde a face humana destaca-se por oferecer diversas informações acerca do estado afetivo humano. O avanço nos estudos sobre a emoção humana e comunicação não verbal, impulsionam inúmeras aplicações em campos diversos como segurança, medicina e educação (CALVO; D'MELLO, 2010; CHATTERJEE; CHANDRAN, 2016; PUNYANI; GUPTA; KUMAR, 2020; Pantic; Patras, 2006).

Neste trabalho, na fase de pré-processamento, métodos como *Viola-Jones* (VJ) (VIOLA; JONES, 2001), *Kanade-Lucas-Tomasi* (KLT) (CHAI; SHI, 2011; BARNOUTI; AL-MAYYAHI; AL-DABBAGH, 2018) e Análise de Componentes Principais, do inglês *Principal Component Analysis* (PCA) (LIU; KAU, 2017) são usados para extrair características da face e produzir o vetor de atributos.

Na etapa de processamento, Redes Neurais Artificiais, do inglês *Artificial Neural Networks* (ANN) (MARKOU; SINGH, 2003b; PIMENTEL et al., 2014) são utilizadas para

DN e classificação dos estados afetivos da face, possibilitando a produção de um algoritmo compacto com baixo custo computacional que detecta novidades no reconhecimento de expressões faciais em tempo real em fluxo de vídeo.

O algoritmo VJ inicialmente detecta o rosto em um *frame* de vídeo e consecutivamente o algoritmo KLT identifica a região da face com um conjunto de pontos chamados de bons recursos permitindo o rastreamento do rosto durante a execução do vídeo (CHAI; SHI, 2011; BARNOUTI; AL-MAYYAHI; AL-DABBAGH, 2018).

O algoritmo PCA caracterizado como um método estatístico utilizado para extrair os componentes principais de um conjunto de dados (LIU; KAU, 2017) é empregado neste trabalho para extrair atributos dos dados obtidos com os algoritmos VJ e KLT, onde apenas o primeiro componente é utilizado para codificação do rosto humano, possibilitando a produção do vetor atributos para treinamento e teste da rede.

Em aplicações com processamento de vídeo em que o mesmo objeto pode mudar gradualmente durante a operação devido as diferentes condições de iluminação, tempos de exposição e outros motivos, as redes neurais tornam-se uma técnica muito útil (Markou; Singh, 2006), como também tornam-se atrativas por não precisarem de atualizações em relação aos dados de treinamento para detectar novos eventos (MARKOU; SINGH, 2003b).

Na fase de processamento, as redes Perceptron de Multicamadas, do inglês *Multilayer Perceptron* (MLP) e Função de Base Radial, do inglês *Radial Basis Function* (RBF) são empregadas em uma abordagem multi-classes, sendo avaliadas quanto ao seu respectivo desempenho em processamento, assertividade para DN e classificação dos estados afetivos da face.

A base de dados proposta como recurso visual para pesquisa, conta com uma seleção dinâmica e multimodal de expressões faciais e vocais em inglês avaliando a autenticidade emocional de 24 atores profissionais (12 mulheres e 12 homens). As emoções calma, feliz, triste, raiva, medo, surpresa e nojo estão incluídas. Cada expressão é selecionada em dois níveis de excitação emocional (normal e forte) com uma expressão neutra adicional (LIVINGSTONE, 2018; Abdullah; Ahmad; Han, 2020).

Assim, este trabalho investiga as redes MLP e RBF na tarefa de DN no reconhecimento de expressões faciais em fluxo de vídeo para uma abordagem multi-classes. As redes são avaliadas acerca da DN com *frames* de vídeos semelhantes aos usados na fase de treinamento e *frames* de vídeos integralmente novos. Motivado igualmente na integração dos métodos VJ, KLT, PCA e ANN para implementar um algoritmo compacto que detecta novas expressões faciais ou classifica expressões faciais em fluxo de vídeo em tempo real. Esta pesquisa também caracteriza-se como inovadora por fornecer uma nova visão para o sistema de Reconhecimento de Expressões Faciais, do inglês *Facial Expression Recognition* (FER) agregando a abordagem DN. Esta renovação fornece uma nova perspectiva para o

progresso de tecnologias e produtos existentes. Deste modo, este trabalho visa contribuir para o desenvolvimento de sistemas inteligentes no campo da computação afetiva.

1.1 Justificativa

Detectar novos eventos é uma tarefa desafiadora e importante em diversos sistemas de classificação, visto que a habilidade de DN é muito útil para sistemas que necessitam de atualização acerca da aprendizagem, de modo que um sistema multi-classes que necessita de atualização poderá sempre ser treinado com novas possibilidades de aprendizado existentes (Markou; Singh, 2006).

Logo, o desenvolvimento de um algoritmo compacto com baixo custo computacional capacitado para detectar novidades e classificar expressões faciais em fluxo de vídeo possibilitará agregar aos dados normais maior variedade dos estados afetivos da face e consequentemente viabilizará o desenvolvimento de sistemas inteligentes embarcados (Suchitra; Suja P.; Tripathi, 2016).

Deste modo, a pesquisa acerca da DN aplicada ao reconhecimento de expressões faciais em fluxo de vídeo visa contribuir para o progresso do projeto robô assistivo denominado HiBot, que está sendo desenvolvido no Laboratório de Robótica do Departamento de Engenharia Elétrica da Universidade Federal da Bahia para atender crianças com Transtorno do Espectro do Autista (TEA) (Camada; Cerqueira; Lima, 2017; Ismail et al., 2011; Ahmed et al., 2013). Trata-se de um robô constituído por um conjunto de sensores e atuadores afetivos que possibilitará a Interação Humano-Robô, do inglês *Human-Robot Interaction* (HRI). Assim, esta pesquisa pode ser futuramente adaptada para DN no reconhecimento de expressões faciais para crianças com TEA.

1.2 Objetivos

1.2.1 Objetivos Gerais

Este trabalho visa o desenvolvimento de um algoritmo compacto com baixo custo computacional implementado em ambiente MATLAB (versão 2018.a) para DN aplicada ao reconhecimento de expressões faciais em fluxo de vídeo e consecutivamente para classificação dos estados afetivos básicos da face.

1.2.2 Objetivos Específicos

Os objetivos específicos deste trabalho consistem em:

- Detectar novidades no reconhecimento de expressões faciais em fluxo de vídeo usando as redes neurais MLP e RBF.

- Classificar os estados afetivos da face em fluxo de vídeo conforme dados de treinamento usando as redes neurais MLP e RBF para uma abordagem multi-classes.

- Extrair atributos de expressões faciais em fluxo de vídeo utilizando os métodos VJ, KLT e PCA.

- Examinar a performance das redes MLP e RBF para DN com expressões faciais semelhantes as usadas na fase de treinamento e investigar a DN para expressões faciais integralmente novas.

- Avaliar a resposta de tempo real do algoritmo desenvolvido com base na taxa de quadros por segundo, do inglês *Frames Per Second* (FPS) para DN e classificação dos estados afetivos da face com uso das redes MLP e RBF.

- Estimar o custo computacional dos algoritmos propostos com as redes MLP e RBF a partir do tempo de computação e quantidade de memória exigida do computador para as etapas de DN e classificação dos estados afetivos da face.

1.3 Artigos e Participações em Eventos

A seguir são listados os artigos e participações em eventos relacionados a este trabalho:

- Bove, M. S. P., Cerqueira, J. J. F., Simas Filho, E. F. (2020). *Novelty Detection Applied in Recognition of Facial Expressions.* XXIII Congresso Brasileiro de Automática, 1-8.

- Apresentação por vídeo gravado no XXIII Congresso Brasileiro de Automática, no período de 23 à 26 de novembro de 2020.

1.4 Organização do Trabalho

- **Capítulo 2. Fundamentação Teórica**

  Neste capítulo outros trabalhos correlatos relevantes realizados na pesquisa com DN para o processamento de imagens ou vídeos são apresentados para análise. Similarmente, apresenta-se trabalhos afins para as metodologias VJ, KLT, PCA e

para o uso do Banco de Dados Audiovisual Ryerson de Fala Emocional e Canção, do inglês *The Ryerson Audio-Visual Database of Emotional Speech and Song* (RAVDESS) para reconhecimento de expressões faciais com redes neurais. Descreve-se acerca das informações globais da arquitetura do projeto Hibot destinado ao tratamento de crianças com TEA. Especifica-se particularidades do reconhecimento de expressões faciais para uma abordagem automatizada, destaca-se métodos e técnicas para extração de atributos da face com o uso dos algoritmos VJ, KLT e PCA. Por fim, apresenta-se os métodos e técnicas empregados para capacitar as redes MLP e RBF para DN.

- **Capítulo 3. Materiais e Métodos**

  Para este capítulo descreve-se sobre as métricas de avaliação do algoritmo, destaca-se as características técnicas do banco de dados, apresenta-se a organização do conjunto de dados para as etapas de treinamento e teste com as redes MLP e RBF. Apresenta-se também os métodos empregados para treinamento e DN com as redes MLP e RBF.

- **Capítulo 4. Resultados e Discussão**

  Neste capítulo apresenta-se os resultados obtidos para: (i) extração de atributos da face, (ii) reconhecimento das expressões faciais conforme dados de treinamento, (iii) DN com faces semelhantes as faces usadas na fase de treinamento, (iv) DN com faces totalmente novas, (v) avaliação da resposta de tempo real do algoritmo proposto com base na taxa de FPS e (vi) previsão da eficiência computacional dos algoritmos MLP e RBF considerando o tempo de computação e memória requerida para as etapas de classificação e DN.

- **Capítulo 5. Conclusão**

  Para o capítulo de conclusão descreve-se em síntese acerca das contribuições alcançadas nesta pesquisa, como também apresenta-se recomendações para trabalhos futuros.

- **Apêndice**

  Para o apêndice disponibiliza-se o artigo *Novelty Detection Applied in Recognition of Facial Expressions* realizado na pesquisa.

- **Anexos**

  Em anexos disponibiliza-se os códigos desenvolvidos na pesquisa.

2 Fundamentação Teórica

Neste capítulo, apresenta-se pesquisas e resultados que tratam dos métodos empregados neste trabalho. De maneira introdutória apresenta-se as caraterísticas da arquitetura do projeto HiBot e informações acerca da sua aplicação. Aborda-se sobre a fundamentação teórica para o reconhecimento de expressões faciais. Apresenta-se aspectos técnicos sobre os métodos de extração de atributos da face com os algoritmos VJ, KLT e PCA. Por fim, descreve-se a respeito dos métodos empregados para habilitar as redes MLP e RBF para DN.

2.1 Trabalhos Correlatos

Nesta seção apresenta-se pesquisas correlacionadas a DN para o processamento de imagens ou vídeos que são abordadas com o objetivo de apresentar métodos, técnicas e resultados realizados por outros pesquisadores. Da mesma forma, apresenta-se trabalhos para as metodologias VJ, KLT, PCA e para o banco de dados RAVDESS. Descreve-se acerca da proposta do trabalho e diferencial da pesquisa.

2.1.1 Detecção de Novidades com Redes Neurais

O trabalho de Markou e Singh (2006) proposto para DN com ANN para sequência de imagens usa um modelo baseado em uma classe que combinada a rede MLP com um filtro de saída para DN. Este modelo é avaliado com a utilização de quatro pares de vídeos, onde o algoritmo proposto é testado pela performance de reconhecer e classificar imagens como vegetação, céu, estrada, água e pedra em cenas naturais de vídeo, consecutivamente pelo desempenho em detectar novos objetos em cenas naturais de vídeo como pasta, tecido, nuvem, cadeira, caixa de madeira e bola. Os resultados experimentais produzidos apresentam diferentes níveis de desempenho de acordo com a métrica Z para DN. Para abordagem com uso da rede MLP os melhores desempenhos na análise das sequências de vídeo com a métrica Z são 89%, 92%, 94% e 67%.

Na pesquisa de Wildermann e Teich (2008) com processamento de imagens em ambiente dinâmico para adaptação *online*, redes RBF treinadas para reconhecimento de uma classe juntamente com um algoritmo de aprendizagem sequencial são usadas para detectar novos dados recebidos e posteriormente para modificar a topologia da rede adicionando novos neurônios na camada oculta com base na novidade detectada. O treinamento é realizado com a base de dados ORL contendo 400 imagens de faces de 40 pessoas diferentes, todas tiradas em condições de iluminação variadas, com inclinação e

rotação de até 20° e diferentes expressões faciais. Para avaliar as redes, usou-se 80 faces de 8 pessoas diferentes do banco de dados ORL com amostras positivas e novamente a mesma quantidade de amostras negativas são usadas do banco de dados CBCL. O experimento com 80 amostras positivas e 80 amostras negativas confirmam a performance da rede em detectar a face ou não com aproximadamente 100% de acurácia.

Na pesquisa de Kim e Cho (2019) propõe-se a DN com uso da rede Autoencoder para análise de uma classe. O modelo proposto consiste em um codificador que extrai atributos das imagens de entrada, um decodificador que reconstrói as imagens para verificar as características extraídas e um discriminador que é usado para DN. O modelo proposto faz uso do banco de dados de dígitos manuscritos MNIST, do banco de dados de objetos CALTECH-256 e do banco de dados com a circulação de pedestres UCSD Ped1. Para verificar o desempenho da DN usa-se os resultados com a área sobre a curva, do inglês *Area Under the Curve* (AUC) das características operacionais do receptor, do inglês *Receiver Operating Characteristic* (ROC). Para detecção de anomalias com os bancos MNIST e CALTECH-256 obtém-se uma AUC de 87,5% e para base de dados UCSD Ped1 atingi-se uma AUC de 89% na detecção de padrões anômalos em imagens com a circulação de pedestres.

No trabalho de Nantes, Brown e Maire (2013) as ANN são usadas para detectar anomalias em ambiente virtual 3D para estudo de uma classe. As redes MLP e Mapas de Kohonen, do inglês *Self-Organized Map* (SOM) são treinadas para aprender a aparência geométrica e colorida correta de objetos e detectar representações novas ou anômalas que afetam a geometria e cor da imagem. Para o treinamento das redes 1000 imagens livres de *bugs* foram geradas de acordo com a avaliação de um observador humano e exemplos anômalos foram produzidos alterando a geometria e cor das imagens. Os gráficos *boxplots* mostram o desempenho para detecção de anomalias, de modo que os bons resultados obtidos para detecção de anomalias na avaliação da geometria não são conclusivos acerca do parecer de superação entre as redes MLP e SOM. Para avaliação da cor obteve-se uma AUC de 65% com uso da rede MLP e uma AUC de 95% com a utilização da rede SOM.

Abordagens diversas com o uso de ANN são empregadas para DN no processamento de imagens ou vídeos. As pesquisas descritas acima reúnem aplicações com uso das redes MLP, RBF, SOM e Autoencoder baseada em uma única classe. Contudo, o processo realizado para habilitar as ANN para DN reúne métodos diversos de acordo com o modelo de rede, sendo necessário treinamento, ajustes e testes para identificar novos dados com uso de redes neurais (MARKOU; SINGH, 2003b; HODGE; AUSTIN, 2004).

2.1.2 Detecção, Rastreamento e Extração de Atributos Faciais.

A pesquisa de Dang e Sharma (2017) discute e analisa quatro algoritmos básicos que são utilizados para detecção de objetos: (i) *Viola-Jones*, (ii) *SMQT Features and SNOW*

Classifier, (iii) *Neural Network-Based Face Detection* e (iv) *Support Vector Machines-Based face detection*. Os resultados produzidos são estimados com base na métrica *recall* de modo que o algoritmo VJ apresenta o melhor desempenho para detecção de objetos.

Na pesquisa de Chatterjee e Chandran (2016) um estudo comparativo em relação aos algoritmos *Camshift* e KLT é apresentado para detecção do rosto em tempo real com o uso de uma *webcam*. Os resultados do estudo comparativo entre os algoritmos *Camshift* e KLT são apresentados com uso de imagens, que mostram o melhor desempenho do algoritmo KLT para tarefa de rastreamento da face.

No trabalho de Cherabit, Djeradi e Chelali (2020) os algoritmos *Normalized Sum of Squared Differences* (NSSD), *Normalized Cross Correlation* (NCC) e KLT são avaliados em relação ao rastreamento de rostos falantes em movimento. Os resultados são apresentados com base no erro de rastreamento da face durante a execução do vídeo e no tempo de processamento. O algoritmo KLT apresenta o menor erro e também o menor tempo de processamento, cerca de um segundo.

2.1.3 Base de Dados.

Na pesquisa de Abdullah, Ahmad e Han (2020) Redes Neurais Recorrentes, do inglês *Recurrent Neural Networks* (RNN), Redes Neurais de Convolução, do inglês *Convolution Neural Networks* (CNN) juntamente com o banco de dados RAVDESS são usadas no reconhecimento de expressões faciais em fluxo de vídeo. Neste processo 3923 vídeos foram usados na fase de treinamento, 490 vídeos na fase de validação e 491 vídeos foram usados para fins de testes. O resultado disponível para esse conjunto de dados usando apenas a análise visual fornece 61% de precisão para o teste de classificação.

No trabalho de Ghaleb, Popa e Asteriadis (2019) redes neurais *SoundNet*, 3D-CNN e 3D *ConvNet* são organizadas em uma estrutura unificada para reconhecer emoções em recurso de áudio e vídeo. O método proposto avalia dois conjuntos de dados, CREMA-D e RAVDESS. Os resultados do estudo são promissores, alcançando um desempenho satisfatório em ambos os conjuntos de dados e mostrando um impacto significativo da percepção da emoção multimodal e temporal com uma taxa de reconhecimento de 65% para o banco de dados CREMA-D e 67% para o banco de dados RAVDESS.

Na pesquisa de Zhihao et al. (2019) as redes neurais convolucionais *AlexNet*, *GoogleNet* e *ResNet* são utilizadas para realizar o reconhecimento da emoção humana em vídeo. O treinamento e teste das redes neurais foram realizados com o conjunto de dados RAVDESS. Os resultados obtidos para classificação no reconhecimento das emoções em vídeo apresentam uma acurácia de 79.74% para rede *AlexNet*, 75.89% para rede *ResNet* e 62.89% para rede *GoogleNet*.

2.1.4 Proposta do Trabalho.

Esta pesquisa propõe investigar a DN aplicada ao reconhecimento de expressões faciais em fluxo de vídeo com a integração dos métodos VJ, KLT, PCA e ANN. De modo que os algoritmos VJ, KLT e PCA são usados para extrair atributos da face e as redes neurais MLP e RBF são utilizadas para DN e classificação das expressões faciais.

A base de dados da pesquisa utiliza vídeos do banco de dados RAVDESS que são organizados em vídeos de treinamento e vídeos de teste para DN. Os resultados avaliam a performance dos algoritmos com uso das redes MLP e RBF em relação ao desempenho de classificação dos estados afetivos da face, com relação ao percentual de acerto na tarefa de detecção de novos estados afetivos da face, com referência ao percentual de acerto na detecção de novas faces, em relação a resposta de tempo real com base na taxa de FPS, por fim acerca da eficiência computacional dos algoritmos segundo o tempo de computação e quantidade de memória requerida pelo algoritmo para as etapas de DN e classificação.

Assim, o diferencial desta pesquisa comparado aos outros trabalhos apresentados para DN com imagens ou vídeos faz referência a abordagem multi-classes, onde as redes neurais MLP e RBF são treinadas para classificar expressões faciais de quatro estados afetivos diferentes e avaliadas a acerca da taxa de acerto em detectar novos *frames* de vídeo com estados afetivos da face que são similares aos usados na fase de treinamento ou detectar *frames* de vídeos com faces nunca antes vistas.

2.2 Robô Assistivo HiBot

A Robótica Socialmente Assistiva, do inglês *Socially Assistive Robotics* (SAR) é um subcampo de pesquisa da HRI, que visa o desenvolvimento de máquinas capazes de auxiliar os usuários, normalmente em contextos de saúde e educação, por meio de interação social, em vez de física (Matari, 2014; Nikolopoulos et al., 2011; Chen; Chen, 2018). De modo que, robôs assistivos tem sido amplamente pesquisado no campo HRI para terapia de crianças com TEA (Ismail et al., 2011; Ackovska et al., 2017; Askari et al., 2018).

O uso de robôs como tecnologia assistencial deu origem à Terapia Assistida por Robôs, sendo um dos campos de pesquisa em crescimento. Nos últimos quinze anos, houve um grande progresso no tratamento de crianças autistas por meio da interação robótica para fins terapêuticos (Ackovska et al., 2017). Acredita-se também que HRI produz um efeito de alto impacto quando o robô é capaz de comunicar e interagir positivamente com humanos, permitindo aprimorar as habilidades humanas de comunicação e interação social (Ismail et al., 2011).

A ideia parte de que crianças com TEA se sentem seguras em um ambiente previsível com os comportamentos repetitivos de robôs, enquanto tendem a ter medo das aparências

e movimentos de humanos que são complicadas e mutáveis (Lee; Obinata; Aoki, 2014). O projeto do robô assistivo Hibot conta com um módulo de sensores afetivos e um módulo de atuadores afetivos que são destinados a interação do robô com a criança autista, visando auxiliar terapeutas e familiares no tratamento do TEA.

O projeto Hibot reúne pesquisadores diversos do Laboratório de Robótica do Departamento de Engenharia Elétrica da Universidade Federal da Bahia que atuam no desenvolvimento do robô. Esta pesquisa concentra seus esforços no desenvolvimento de um algoritmo compacto com baixo custo computacional com eficiência e eficácia para detectar novidades e reconhecer expressões faciais, pretendendo no futuro a implementação do módulo de reconhecimento de expressões faciais do projeto robô Hibot.

Um algoritmo com resposta em tempo real que classifica e detecta novidades no reconhecimento de expressões faciais vai capacitar o robô HiBot a entender os gestos e emoções (Suchitra; Suja P.; Tripathi, 2016) da criança autista provenientes da face, melhorando sua eficiência na tarefa de HRI. A DN por sua vez leva o projeto HiBot ao um novo patamar de HRI, onde novos gestos e emoções da criança autista poderão ser utilizadas para agregar maiores dados ao módulo de reconhecimento de expressões faciais e consequentemente para contribuir com os terapeutas e familiares na evolução do tratamento. A Figura 1 mostra a arquitetura proposta para o desenvolvimento do projeto Hibot.

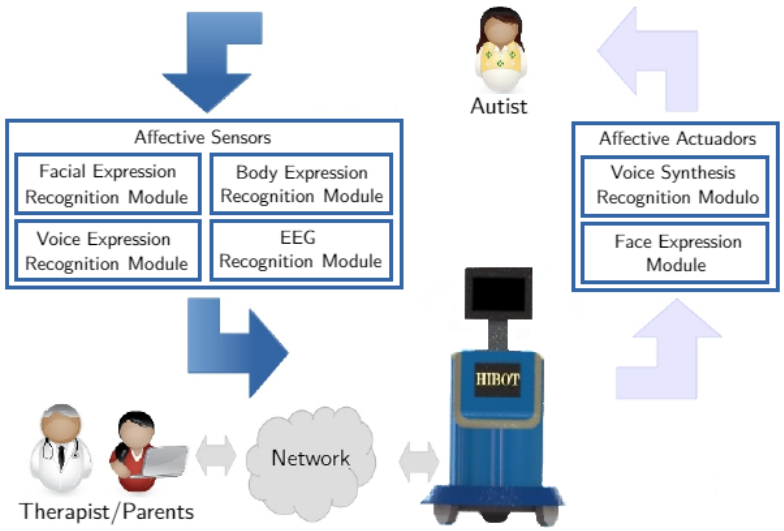

Figura 1 – Arquitetura do projeto robô assistivo HiBot. Fonte: Autor.

2.3 Expressões Faciais

O rosto humano possui uma estrutura complexa e dinâmica (DELAC; GRGIC; GRGIC, 2005), sendo uma das informações mais importantes nas ciências biométricas com base na identificação pessoal (Abbas; Safi; Rijab, 2017). A face humana engloba uma grande

variedade de atividades diferentes, mediando a identificação da pessoa, a atratividade e as pistas comunicativas faciais, de forma que nossos enunciados são acompanhados por expressões faciais adequadas, que esclarecem o que é dito e se é suposto ser importante, engraçado ou sério (Pantic; Patras, 2006).

As expressões faciais são consideradas as mais indicadas para reconhecer o estado psicológico de uma pessoa durante a comunicação (JAMEEL; SINGHAL; BANSAL, 2016), visto que as expressões faciais revelam nosso foco atual de atenção, sincronizam o diálogo, sinalizam compreensão ou desacordo, regulando as nossas interações com o meio ambiente e outras pessoas em nossa vizinhança (Pantic; Patras, 2006).

Nos últimos anos, diferentes abordagens foram propostas para desenvolver métodos de análise totalmente automatizada para medir e descrever a atividade muscular facial (Pantic; Patras, 2006; JAMEEL; SINGHAL; BANSAL, 2016). Dentre as abordagens, o Sistema de Codificação de Ação Facial, do inglês *Facial Action Coding System* (FACS) é o método mais amplamente utilizado na pesquisa psicológica, onde as expressões faciais são organizadas em Unidades de Ação, do inglês *Actios Units* (AUs) (Pantic; Patras, 2006).

O FACS conta com um estudo sobre a existência de emoções universais entre grupos sociais e raciais distintos que padronizam o processo de reconhecimento facial, classificando e rotulando as emoções como básicas, expressas pelos estados afetivos de felicidade, raiva, tristeza, surpresa, desgosto e medo. Este processamento é feito em várias fases incluindo aquisição de imagens, extração de atributos e finalmente classificação de expressões (JAMEEL; SINGHAL; BANSAL, 2016).

Nomeadamente, as alterações na expressão facial são descritas com FACS em termos de 44 unidades de ação diferentes, cada uma das quais está anatomicamente relacionada com a contração de um músculo facial específico ou de um conjunto de músculos faciais (Pantic; Patras, 2006). As Figuras 2 e 3 mostram AUs facial superior e inferior para expressões faciais universais.

Figura 2 – Unidades da ação facial superior. Adaptado de (De la Torre et al., 2015)

Figura 3 – Unidades da ação facial inferior. Adaptado de (De la Torre et al., 2015)

2.4 Atributos Faciais

No domínio do processamento de imagens, o reconhecimento do rosto se tornou um dos fenômenos mais interessantes para aplicações pragmáticas. O processo de reconhecimento facial significa o reconhecimento de uma pessoa individual por fisionomia a partir de imagens faciais capturadas, combinando certos vetores de atributos em um banco de dados predefinido (Das; Akter, 2017).

Métodos empregados para extração de atributos faciais usando imagens ou quadros de vídeos destinados ao reconhecimento facial, usualmente fazem uso de três técnicas: (i) holístico, onde o rosto é processado como um todo, (ii) baseado em características, na qual o rosto é processado em regiões específicas e (iii) híbrido, onde faz-se uso dos dois métodos anteriores (Tong; Liao; Ji, 2007). Estes métodos são empregados visando a produção de um vetor de atributos com alto nível de representação das características da face.

A abordagem holística empregada nesta pesquisa para classificação dos estados afetivos da face e DN leva em consideração informações globais de um determinado conjunto de faces. Essas informações globais são representadas por vetores de atributos que são obtidos diretamente dos *pixels* das imagens faciais. Esses recursos são responsáveis por identificar e representar distintamente as variações entre as diferentes imagens faciais e, portanto, identificar de forma única o indivíduo ou o sujeito (Zafaruddin; Fadewar, 2014).

A proposta principal deste método consiste na segmentação integral do rosto, examinando um conjunto de características que não depende das formas geométricas da face como olhos, nariz, orelhas e boca. Utilizando assim toda a informação da representação facial, onde o tamanho do vetor de atributos resultante é representado pelo número de

pixels provenientes da dimensão (largura x altura) do quadro de vídeo segmentado.

Métodos estatísticos de redução de dimensionalidade são utilizados para mapear dados de entrada para novos dados de saída, podendo ser empregados para eliminar redundâncias e características indesejadas da face segmentada. A técnica PCA usada neste trabalho reduz o grande espaço de dados de dimensionalidade em um espaço de atributos de menor dimensionalidade, necessário para descrever os dados de modo eficiente (Meher; Maben, 2014).

2.5 Reconhecimento de Expressões Faciais

O FER abrange os estágios de localização da face, normalização, extração de atributos e classificação. A localização da face, comumente denomina-se como detecção facial, pretendendo extrair da imagem de entrada somente o conteúdo da imagem facial, porém quando este processo ocorre por meio de vídeo, há ainda mais uma etapa que consiste em rastrear no vídeo a face. A normalização envolve o pré-processamento da imagem facial que consiste em realizar ajustes para melhor descrição da face. Na fase de extração de atributos usa-se comumente métodos baseado em recurso de textura, borda, recursos global e local, geométricos ou baseado em correção. No termino do processo do sistema FER, emprega-se um classificador para identificar as expressões faciais (REVINA; EMMANUEL, 2018; JAMEEL; SINGHAL; BANSAL, 2016).

Sistemas automatizados de HRI aplicado no reconhecimento dos estados afetivos da face humana, inicialmente necessitam detectar a face, consecutivamente rastrear as mudanças dinâmicas provenientes dos movimentos da face e consecutivamente reconhecer as expressões faciais (Ismail et al., 2011). Esta sinergia produzida em tempo real viabiliza a HRI, permitindo aos robôs a tomada de decisão para que assim possam inferir nos estados emocionais dos humanos a partir da interação visual.

Métodos e técnicas para detecção facial, rastreamento facial e reconhecimento de expressões faciais aplicadas em fluxo de vídeo, são temas em ascensão na área de pesquisa de visão computacional, onde avanços tecnológicos a cerca destas investigações contribuem para o sucesso da HRI (Yongmian Zhang; Qiang Ji, 2005). O reconhecimento de expressões faciais em fluxo de vídeo produz uma dinâmica semelhante a dinâmica da HRI, de modo que o algoritmo deve ser capaz de detectar a face, rastrear a face conforme as alterações de posição e reconhecer os estados afetivos durante a produção do vídeo.

2.5.1 Detecção Facial

A detecção do rosto é uma das tecnologias computacionais que está conectada ao processamento de imagens, sendo uma etapa importante para o processo de reconhecimento facial. O objetivo básico dos algoritmos de detecção do rosto são determinar se há algum

rosto em uma imagem ou não. Em outras palavras, a detecção do rosto é uma tarefa em que os rostos mostrados em fotos ou vídeos são procurados automaticamente (Sharifara; Mohd Rahim; Anisi, 2014).

O algoritmo VJ é um dos métodos de detecção facial mais amplamente aplicado no âmbito acadêmico, isso devido a sua robustez em detectar faces em imagens e em consequência da performance de processamento em tempo real com baixo custo computacional (Candra Kirana; Wibawanto; Wahyu Herwanto, 2018). A técnica do algoritmo VJ conta com quatro métodos para realizar a detecção da face: Seleção de Atributos *Haar*, Imagem Integral, Algoritmo *AdaBoost* e Classificadores em Cascata.

- **Seleção de Atributos *Haar*.**

  Os rostos humanos possuem algumas propriedades semelhantes, essas propriedades faciais são comparadas usando os atributos *Haar* também conhecidos como atributos de imagem digital (Dang; Sharma, 2017; Zhu; Chen, 2015). Os atributos *Haar* são obtidos com uso de retângulos que são posicionados dentro de uma janela de detecção.

  A ideia do atributo *Haar* é considerar regiões retangulares adjacentes em um local específico de uma janela de detecção, somar as intensidades de *pixel* em cada região e calcular a diferença entre essas somas. Essa diferença é então usada como uma resposta de atributo para categorizar subseções de uma imagem (Zhu; Chen, 2015). A Figura 4 mostra exemplos de janelas de detecção com retângulos para obtenção de atributos *Haar* com a imagem do ator 1 do banco de dados RAVDESS.

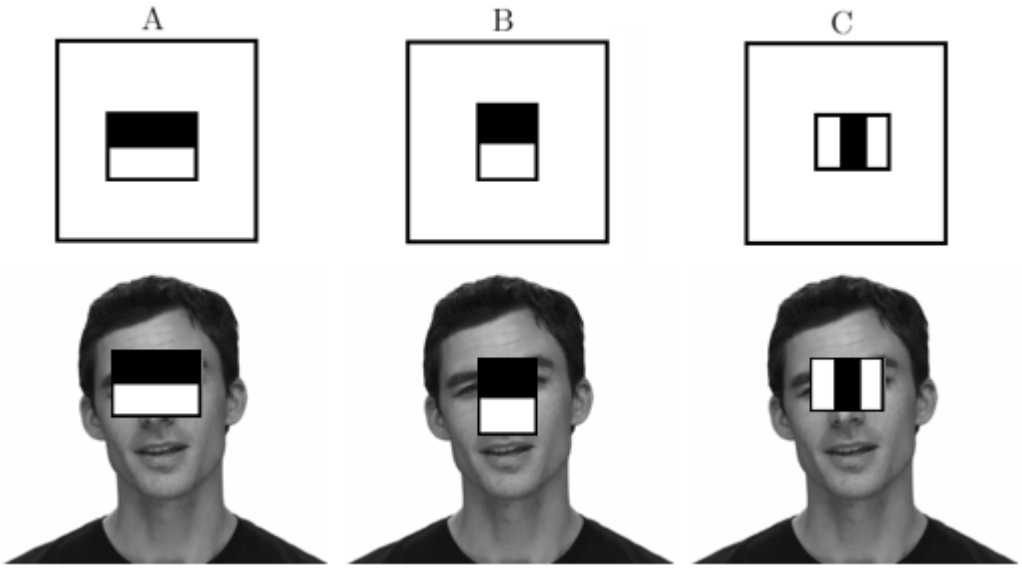

Figura 4 – Exemplos de retângulos posicionados no interior de janelas de detecção para obtenção de atributos *Haar* com a imagem do ator 1 do banco de dados RAVDESS. Fonte: Autor.

A Figura 5 mostra retângulos para obtenção de características de borda, linha e pontos centrais.

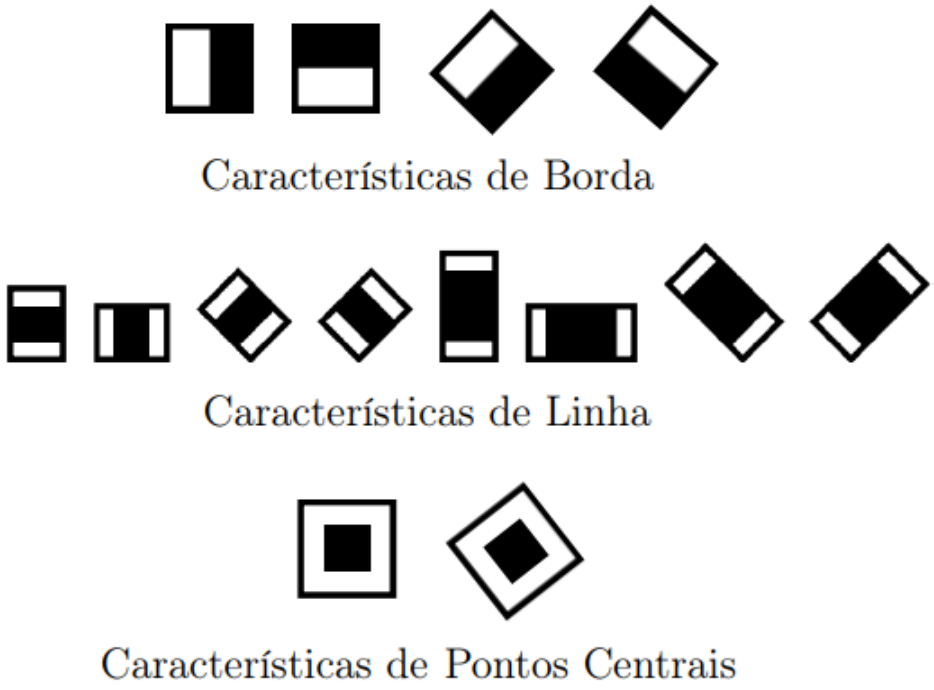

Figura 5 – Características de *Haar* para borda, linha e pontos centrais. Fonte: Autor.

- **Imagem Integral**.

  A imagem integral é empregada no algoritmo VJ para reduzir o número de operações de soma permitindo uma computação com menor custo computacional e maior velocidade de processamento (Dang; Sharma, 2017; Zhu; Chen, 2015).

  No algoritmo VJ uma imagem integral é representada por uma matriz que possui a mesma dimensão da imagem de entrada, sendo conhecida como uma tabela de área somada (Zhu; Chen, 2015). A Equação (2.1) é empregada para obtenção da imagem integral, onde $i(x, y)$ representa um subconjunto retangular de *pixels* da imagem de entrada.

$$ii(x, y) = \sum_{x' \leq x, y' \geq y} i(x', y') \tag{2.1}$$

  Para exemplificar a obtenção do *pixel* $ii(3, 3)$ em ênfase na imagem integral conforme apresentado na Figura 6, destaca-se na imagem de entrada um subconjunto retangular de *pixels* conforme mostrado na Figura 6. Assim, todos os *pixels* da imagem integral são obtidos com a soma à esquerda e acima de uma referência (x, y) a partir da imagem de entrada (Dang; Sharma, 2017; Zhu; Chen, 2015).

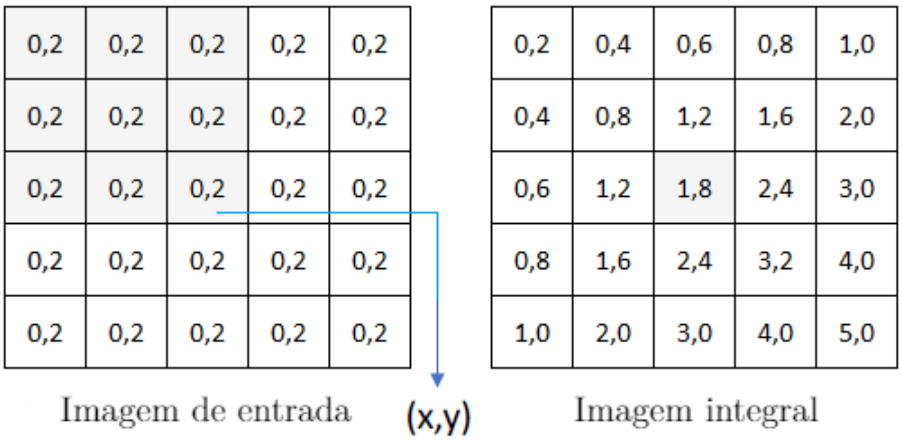

Figura 6 – Matrizes hipotéticas com a representação de uma imagem de entrada e uma imagem integral usadas para demonstrar o processo de obtenção de uma imagem integral. Fonte: Autor.

Para demonstrar o uso da imagem integral, na Figura 7, destaca-se sombreando em cinza na imagem integral os *pixels* respectivos aos valores de referências $ii(A)$, $ii(B)$, $ii(C)$ e $ii(D)$ que são computados com uso da Equação (2.2) para encontrar o valor referente ao somatório de *pixels* sombreados em cinza na imagem de entrada mostrado na Figura 7.

$$\sum_{(x,y)\in ABCD} = ii(D) + ii(A) - ii(B) - ii(C) \tag{2.2}$$

$$= 3,2 + 0,4 - 0,8 - 1,6 = 1,2$$

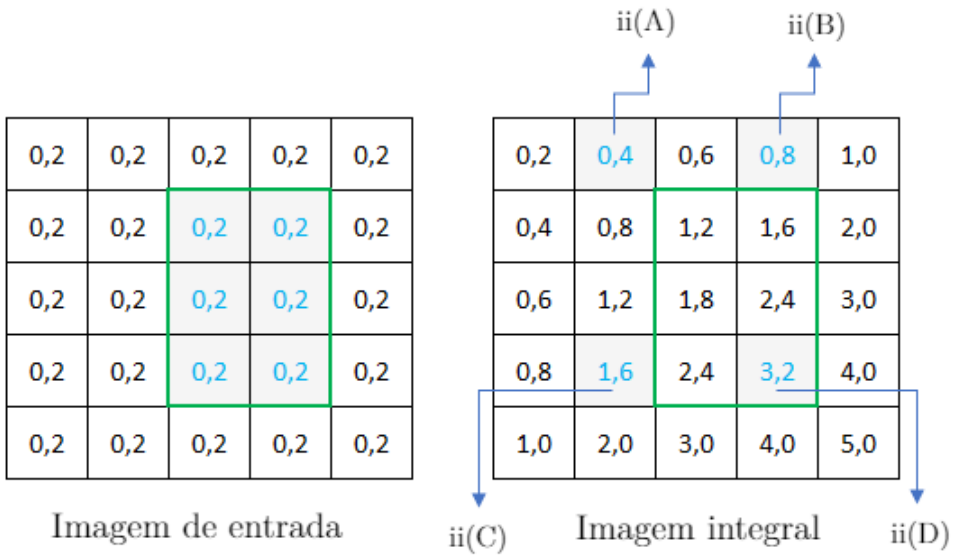

Figura 7 – Matrizes hipotéticas com a representação de uma imagem de entrada e uma imagem integral para exemplificar o uso da imagem integral. Fonte: Autor.

- ***AdaBoost***.

  O *AdaBoost* é um algoritmo de aprendizado de máquina que pode ser aplicado para classificação ou regressão, suas variantes têm sido amplamente utilizadas na detecção de objetos nos últimos anos, e alcançaram um sucesso surpreendente na detecção da face humana (Wu; Nagahashi, 2014).

  No algoritmo VJ, o algoritmo *AdaBoost* é usado para criar uma cascata de classificadores para detecção da face (Peleshko; Soroka, 2013), trata-se de um método iterativo que seleciona classificadores fracos para compor uma classificador forte (Xue; Mao; Zhang, 2006; Tavallali; Yazdi; Khosravi, 2017). Os classificadores fracos são representados por atributos *Haar* que são significativos para detecção da face.

  A Equação (2.3) mostra a descrição para um classificador fraco, onde $t = 1, 2, ..., T$ representa as iterações, x_i uma sub-janela da imagem para um conjunto de N exemplos de treinamento $i = 1, 2..., N$, f_t representa o valor do atributo, p_t a paridade que indica a direção do sinal de desigualdade e θ_t um limiar que decide se x_i deve ser classificado como positivo (uma face) ou negativo (não face) (Chatrath et al., 2014).

$$h_t(x_i, f_t, p_t, \theta_t) = \begin{cases} 1 \text{ se } p_t f_t(x_i) < p_t \theta_t \\ \\ 0 \text{ caso contrário} \end{cases} \tag{2.3}$$

  A Equação (2.4) mostra a descrição para um classificador forte com a combinação linear de classificadores fracos (Nehru; Padmavathi, 2017), onde $\alpha_1, \alpha_2, ..., \alpha_t$ representam os pesos atribuídos aos classificadores fracos.

$$C(x_i) = \alpha_1 h_1 + \alpha_2 h_2 + ... + \alpha_t h_t \tag{2.4}$$

- **Classificadores em Cascata**.

  No último estágio o classificador em cascata combina classificadores de forma a processar eficientemente regiões da imagem em busca de um padrão. Cada estágio da cascata contém um classificador *AdaBoost* mais específico e complexo do que seu anterior, de modo que o algoritmo rejeite rapidamente regiões que sejam muito diferentes das características procuradas, concentrando-se nas áreas importantes da imagem que contêm o objeto de interesse, como uma região facial (Dang; Sharma, 2017). Assim, o classificador em cascata é treinado para detectar faces em tempo real com alto nível de precisão e velocidade de processamento. A Figura 8 mostra o processo de um classificador em cascata utilizado para aceitar ou descartar uma dada entrada.

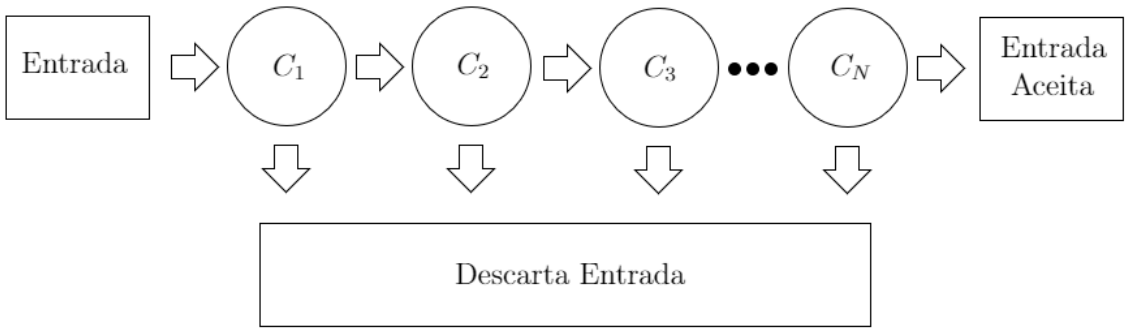

Figura 8 – Classificador em cascata. Fonte: Autor.

2.5.2 Rastreamento Facial

A detecção facial e o rastreamento de faces são tarefas muito importantes para análise em vídeo, de modo que o algoritmo deve ser capaz de lidar com as variações de aparência que podem incluir mudanças de iluminação, movimento da câmera, oclusões e pose onde o ambiente muda com o tempo. Recentemente, a tecnologia biométrica facial é amplamente utilizada em muitos campos, como segurança bancária, passagens de fronteira, *check-in* em aeroportos, monitoramento residencial, reunião remota em escritórios, prisões e fábricas (BARNOUTI; AL-MAYYAHI; AL-DABBAGH, 2018).

O método KLT é uma abordagem de rastreamento robusta que permite rastrear um conjunto de pontos de recursos em quadros de vídeo. Sua eficiência computacional e robustez para mudanças de escala são aspectos relevantes que tornam seu uso amplamente viável no desenvolvimento de sistemas de visão computacional para aplicações em tempo real (BARNOUTI; AL-MAYYAHI; AL-DABBAGH, 2018; CHATTERJEE; CHANDRAN, 2016; Cherabit; Djeradi; Chelali, 2020; Putro; Jo, 2018).

A estrutura do algoritmo KLT é composta por duas fases de operação, extração de recursos e rastreamento (CHAI; SHI, 2011). Na fase de extração de recursos, inicialmente o algoritmo VJ detecta o rosto no vídeo e os pontos de recursos são identificados ao redor do rosto (BARNOUTI; AL-MAYYAHI; AL-DABBAGH, 2018; CHATTERJEE; CHANDRAN, 2016). Com os pontos de recurso identificados na imagem, a tarefa de rastreamento de recurso é rastreá-los quadro a quadro (CHAI; SHI, 2011).

Para cada ponto de recurso do quadro anterior, o rastreador de ponto tenta encontrar um ponto correspondente no quadro atual. O deslocamento dos pares de pontos correspondentes pode ser calculado como vetores de movimento. Assim, o processo de rastreamento da região facial depende do movimento dos centros das feições em dois quadros de vídeo sucessivos (MSTAFA; ELLEITHY, 2016), possibilitando estimar mudanças de translação, rotação e escala entre pontos antigos e novos.

As Equações (2.5), (2.6) e (2.7) descrevem a técnica para computar o rastreamento do rosto através dos quadros de vídeo. Onde R_t e R_{t-1} representam as áreas da face em dois quadros de vídeos adjacentes, C_t e C_{t-1} são os centros de posição dos atributos em dois quadros consecutivos, por fim f_t e f_{t-1} são os pontos característicos nos quadros atual e anterior (MSTAFA; ELLEITHY, 2016).

$$R_t = R_{t-1} + (C_t - C_{t-1}) \tag{2.5}$$

$$C_t = \frac{1}{\mid f_t \mid} \sum_i f_t(i) \tag{2.6}$$

$$C_{t-1} = \frac{1}{\mid f_{t-1} \mid} \sum_i f_{t-1}(i) \tag{2.7}$$

2.5.3 Extração de Atributos Faciais

Métodos de análise e compreensão de imagens tem ganhado destaque nos últimos anos com sucesso em aplicações de reconhecimento facial (DELAC; GRGIC; GRGIC, 2005). A técnica PCA é um procedimento estatístico amplamente utilizado na análise exploratória de dados, que permite a identificação de padrões nos dados, mantendo seu *status* de identificação e reduzindo efetivamente as dimensões em imagens de rostos humanos (Abbas; Safi; Rijab, 2017).

A redução dos dados elimina informações irrelevantes ou redundantes para chegar a uma taxa de compressão mais alta para o primeiro componente (LIU; KAU, 2017) e consecutivamente para o segundo componente, produzindo uma representação de baixa dimensão dos dados de entrada sem perda significativa dos dados originais. A fórmula matemática da PCA é baseada nos autovetores e autovalores da matriz de covariância dos dados, sendo considerada uma técnica robusta com um processo simples, rápido e que funciona bem em ambiente restrito para reconhecimento facial (Abbas; Safi; Rijab, 2017). Deste modo, a técnica de PCA visa encontrar um conjunto de vetores ortogonais v_i referente a matriz de covariância que melhor representa a distribuição dos dados de entrada.

Considerando uma matriz de dados $\mathbf{X} \in \Re^{\,mxn}$ representando as características de uma imagem conforme Equação (2.8).

$$\mathbf{X} = \begin{vmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} & x_{m2} & \dots & x_{mn} \end{vmatrix} \tag{2.8}$$

A matriz normalizada da imagem de entrada $\mathbf{X}$ com vetores linhas centrados em zero é obtida por meio da Equação (2.9), sendo $\mathbf{M}$ a média amostral dos vetores linhas descrito pela Equação (2.10).

$$\mathbf{N} = \mathbf{X} - \mathbf{M} \tag{2.9}$$

$$\mathbf{M} = \frac{1}{n-1} \begin{vmatrix} x_{11} + & x_{12} + & ... & + x_{1n} \\ x_{21} + & x_{22} + & ... & + x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m1} + & x_{m2} + & ... & + x_{mn} \end{vmatrix} \tag{2.10}$$

Deste modo, a matriz de covariância $\mathbf{C} \in \Re^{\,mxn}$ é computada conforme a Equação (2.11).

$$\mathbf{C} = \frac{1}{n-1}\mathbf{N}^T\mathbf{N} \tag{2.11}$$

Logo o primeiro e segundo componente principal da imagem original são computados conforme as Equações (2.12) e (2.13), onde $v_1 = [v_{11}, v_{21}, ..., v_{m1}]^T$ e $v_2 = [v_{12}, v_{22}, ..., v_{m2}]^T$ são denominados como vetores singulares à direita, sendo obtidos da decomposição da matriz de covariância $\mathbf{C}$ pelo método de Decomposição de Valores Singulares, do inglês *Singular Value Decomposition* (SVD).

$$Y_1 = v_{11} \begin{vmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{m1} \end{vmatrix} + v_{21} \begin{vmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{m2} \end{vmatrix} + ... + v_{m1} \begin{vmatrix} x_{1n} \\ x_{2n} \\ \vdots \\ x_{mn} \end{vmatrix} \tag{2.12}$$

$$Y_2 = v_{12} \begin{vmatrix} x_{11} \\ x_{21} \\ \vdots \\ x_{m1} \end{vmatrix} + v_{22} \begin{vmatrix} x_{12} \\ x_{22} \\ \vdots \\ x_{m2} \end{vmatrix} + ... + v_{m2} \begin{vmatrix} x_{1n} \\ x_{2n} \\ \vdots \\ x_{mn} \end{vmatrix} \tag{2.13}$$

O teorema de SVD descrito pela Equação (2.14) é um método de fatoração que decompõe uma matriz $\mathbf{A}$ no produto de três matrizes conforme ilustrado na Figura 9, sendo proposto para resolver o problema com o elevado custo computacional no cálculo de autovalores e autovetores da matriz de covariância proveniente da imagem de treino (Li et al., 2014).

$$\mathbf{A} = \mathbf{U}\mathbf{S}\mathbf{V}^T \tag{2.14}$$

De modo que

- $\mathbf{U} \in \Re^{mxm}$ é uma matriz ortogonal, i.e. $\mathbf{U}^T\mathbf{U} = \mathbf{I}$, e as colunas de $\mathbf{U}$ são autovetores de $\mathbf{A}\mathbf{A}^T$, chamados de vetores singulares à esquerda.

- $\mathbf{V} \in \Re^{nxn}$ é uma matriz ortogonal, i.e. $\mathbf{V}^T\mathbf{V} = \mathbf{I}$, e as colunas de $\mathbf{V}$ são autovetores de $\mathbf{A}^T\mathbf{A}$, chamados de vetores singulares à direita.

- $\mathbf{S} \in \Re^{mxm}$ é uma matriz diagonal, de modo que os elementos $d_1 \geqslant d_2 \geqslant ... \geqslant d_n \geqslant 0$, e esses valores são a raiz quadrada dos autovetores de $\mathbf{A}^T\mathbf{A}$, chamados de valores singulares.

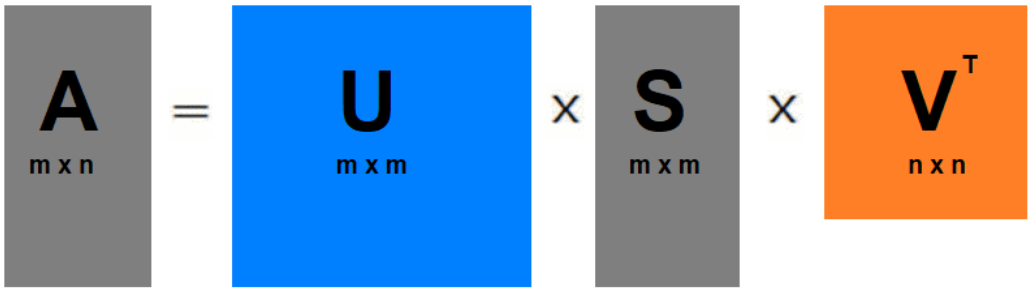

Figura 9 – Interpretação visual para decomposição em valores singulares. Fonte: Autor.

2.6 Detecção de Novidades

A pesquisa no campo da DN tem ganhado destaque nos últimos anos devido a força motriz de aplicações nas áreas de monitoramento de sistemas críticos de segurança e detecção de novos objetos em sequências de imagens (Yuhua Li, 2008). De forma que, termos como detecção de novidade (*novelty detection*), detecção de anomalia (*anomaly detection*) e detecção de observações discrepantes (*outlier detection*) são originados de diferentes domínios de aplicação, no qual todos buscam encontrar modelos que diferem dos modelos normais, sendo as definições anomalia e *outlier* atribuídas a uma situação indesejada e o termo novidade a uma condição nova que precisa ser incorporada ao padrão normal (PIMENTEL et al., 2014).

A DN depende de modelos construídos em dados de treinamento coletados de classes conhecidas, assim quando um detector de novidades é aplicado para classificar um vetor de dados não visto, ele é capaz de decidir se o vetor de dados está associado a classes conhecidas ou resultou de novas classes (Yuhua Li, 2008), de modo que a DN é geralmente investigada com base na abordagem estatística ou baseada em redes neurais (MARKOU; SINGH, 2003a; MARKOU; SINGH, 2003b).

A abordagem estatística visa estimar características na distribuição dos dados usando técnicas de modelagem estatística. Essas propriedades estimadas para condição de normalidade são então usadas para detectar quando uma amostra de teste vem da mesma distribuição ou não (Yuhua Li, 2008).

Técnicas de DN estatísticas especificamente avançadas são baseadas em uma estimativa de densidade de probabilidade que é usada para calcular a probabilidade de uma amostra de teste, de forma que um limite é então aplicado à probabilidade para dar uma indicação de novidade (Yuhua Li, 2008; PIMENTEL et al., 2014), onde a densidade de probabilidade pode ser estimada usando métodos paramétricos ou não paramétricos (MARKOU; SINGH, 2003a; PIMENTEL et al., 2014).

Métodos paramétricos assumem que os dados são provenientes de um conjunto de distribuições conhecidas com modelos especificados por parâmetros. No entanto, as formas de distribuição subjacentes de dados do mundo real geralmente são desconhecidas até que tenhamos conhecimento suficiente sobre o problema (Yuhua Li, 2008; MARKOU; SINGH, 2003a).

Métodos não paramétricos são preferidos para aplicações no mundo real, uma vez que não fazem suposições sobre as propriedades estatísticas dos dados. A desvantagem da estimativa de densidade é que ela requer um grande número de amostras para treinamento do modelo padrão, tornando-se uma tarefa muito desafiadora, especialmente quando a dimensionalidade dos dados é alta (Yuhua Li, 2008).

A abordagem baseada em rede neural é fundamentada principalmente na construção de limites de decisão fechados em torno dos dados de treinamento, onde um vetor de teste é considerado uma amostra nova se estiver fora dos limites fechados. Assim, a propriedade próxima dos limites é essencial para o sucesso da DN (Yuhua Li, 2008; PIMENTEL et al., 2014).

2.6.1 Redes Neurais Artificiais

As redes neurais são conceituados como sistemas computacionais adaptativos, baseado no comportamento das redes neurais biológicas, sendo formadas por um conjunto de neurônios artificiais que interagem entre si, com capacidade para aprender com dados representativos do problema usando treinamento para ajustar seus pesos sinápticos (HAYKIN, 2009). Sua capacidade de aprender limites complexos para identificar classes e modelar dados implícitos de forma autônoma tornam as redes neurais um método amplamente usado para DN (MARKOU; SINGH, 2003b).

O treinamento das redes podem ser realizados pelos métodos de aprendizado supervisionado, aprendizado não-supervisionado e aprendizado por reforço.

- **Aprendizado Supervisionado.**

  Um agente externo com conhecimento do ambiente apresenta dados de entrada-saída para que a rede através do algoritmo de treinamento possa criar uma representação do sistema em análise.

- **Aprendizado Não-supervisionado ou Auto-organizado.**

  Não há um agente externo, desta forma é fornecido a rede dados com parâmetros livres, que são agrupados de acordo com as similaridades, sendo essa técnica intitulada de clusterização.

- **Aprendizado por Reforço.**

  Assim como no aprendizado não-supervisionado não existe um agente externo, deste modo o mapeamento de entrada-saída é realizada pela interação contínua do sistema, através de recompensas positivas ou negativas, que emergem da atuação do sistema durante o processo de aprendizagem.

Na prática, as redes neurais não detectam novidades automaticamente, pois atuam essencialmente como discriminadoras, em vez de atuar como uma detectora de novos eventos (MARKOU; SINGH, 2003b; Yuhua Li, 2008). Deste modo, o processo realizado para habilitar as redes neurais para tarefa de DN considera as etapas de treinamento, ajuste e teste para determinar limites para realizar a detecção de novos dados (HODGE; AUSTIN, 2004), onde os métodos utilizados para definição dos limites para DN variam dependendo do tipo de rede neural e dos algoritmos associados (Yuhua Li, 2008; PIMENTEL et al., 2014; MARKOU; SINGH, 2003b).

De modo geral, a DN utilizando redes neurais de aprendizado supervisionado considera a avaliação dos resultados apresentados na camada de saída da rede comparando com um valor limite que permite identificar quando um vetor de atributos apresentado na entrada da rede é diferente do vetor de atributos utilizado na fase de treinamento (AUGUSTEIJN; FOLKERT, 2002; SAMEER; MARKOU, 2004).

A metodologia baseada em redes neurais é apropriada para DN com dados estáticos ou dinâmicos (HODGE; AUSTIN, 2004) e também destaca-se por não precisar de atualização dos dados como nos métodos estatísticos de detecção de novos eventos (MARKOU; SINGH, 2003b). Em diversas aplicações, as redes neurais de aprendizado supervisionado mais utilizadas para DN em uma abordagem multi-classe são as redes MLP e RBF (MARKOU; SINGH, 2003b; BARRETO; FROTA, 2012).

2.6.2 Rede MLP

Redes MLP têm sido amplamente aplicadas com sucesso para resolver problemas complexos não-lineares em áreas diversas da engenharia e pesquisa científica. São redes de aprendizado global, tipo *feedforward* de treinamento supervisionado e eficiente para aplicações onde o vetor de entrada possui grandes dimensões (HAYKIN, 2009; PIMENTEL et al., 2014; MARKOU; SINGH, 2003b).

A sua arquitetura é constituída de três conjuntos principais, referindo-se a camada de entrada (*input layer)*, camadas ocultas (*hidden layers)* e camada de saída (*output layer*). A camada de entrada é formada pelas unidades sensoriais podendo ser configurada conforme a dimensão do vetor de atributos. A camada oculta pode ser parametrizada para trabalhar com uma ou mais camadas intermediárias. Por fim, a camada de saída pode ser configurada conforme a necessidade de classes.

A configuração da rede MLP em relação a camada oculta, função de ativação e algoritmo de treinamento são elementos importantes para habilitar a rede MLP para DN.

- **Camada Oculta**

  Em aplicações com DN as redes MLP são parametrizadas para trabalhar com apenas uma única camada oculta, de modo que esta configuração melhora sua performance para DN, consequentemente permitindo detectar limites de classes arbitrariamente complexas (HODGE; AUSTIN, 2004).

- **Função de Ativação.**

  A parametrização da única camada oculta com a função de ativação gaussiana, conduz a uma outra estratégia para amplificar o desempenho da rede MLP em detectar novidades, isso ocorre porque o campo receptivo dos neurônios são forçados a ser mais seletivo, sendo ativado apenas para uma região restrita do espaço de entrada, otimizando a performance da rede para DN (BARRETO; FROTA, 2012).

- **Algoritmo de Treinamento.**

  O algoritmo Gradiente Conjugado em Escala, do inglês *Scaled Conjugate Gradient* (SCG) é baseado no procedimento de gradiente de segunda ordem, demonstrando ter um bom desempenho em uma ampla variedade de problemas, especialmente para redes com um grande número de pesos sinápticos. O algoritmo SCG é projetado para evitar a demorada pesquisa de linha, este algoritmo combina a abordagem da região de confiança do modelo usado no algoritmo de *Levenberg-Marquardt* com a abordagem do gradiente conjugado (Sivasankari; Thanushkodi; Kalaivanan, 2013; ZAKARIA; ISA; SUANDI, 2010).

2.6.3 Rede RBF

As redes RBF possuem uma grande variedade de aplicações em contextos diversos como aproximação de funções, regularização, interpolação ruidosa, previsão de densidade, classificação de padrões, reconhecimento de fala, diagnóstico médico, reconhecimento de caligrafia, processamento de imagem, diagnóstico de falhas, entre outras. Nessas aplicações, as redes RBF são frequentemente usadas com a regra de saída *"Winner Takes All"* (WTA) (KAYHAN; OZDEMIR; EMINOGLU, 2012; HAYKIN, 2009; PONT; JONES, 2002).

São redes do tipo *feedforward* multicamada de aprendizado local, eficientes para aplicações onde a dimensão do vetor de entrada é reduzida. Sua única camada oculta utiliza funções de base radial que permite modelar espaços de grande dimensão com maior desempenho para velocidade de aprendizagem, requisitos de memória e generalização em comparação com MLP (VASCONCELOS; FAIRHURST, 1995). Estratégias de aprendizagem para rede RBF podem ser do gênero supervisionada, auto-organizada, empírica ou híbrida (HAYKIN, 2009).

A parametrização da rede RBF acerca da função de ativação e algoritmo de treinamento também são elementos importantes para habilitar a rede RBF para DN.

- **Função de Ativação.**

  A única camada oculta da rede RBF pode operar com funções de base radial como gaussiana, multi-quadrática ou multi-quadrática inversa. A função de ativação gaussiana possibilita transformar um determinado conjunto de dados x não linearmente separáveis em um novo conjunto, no qual a probabilidade dos padrões tornam-se linearmente separáveis (HAYKIN, 2009) consecutivamente possibilitando a DN. Cada neurônio presente na camada oculta da rede RBF corresponde ao um sino gaussiano que pode ser ajustado conforme a disposição dos dados gerando um espaço bidimensional (KRIESEL, 2007).

- **Algoritmo de Treinamento.**

  O uso do algoritmo de treinamento de regularização Bayesiana auto-organiza a rede RBF para formar um classificador Bayesiano ampliando consecutivamente o desempenho da rede para DN (PIMENTEL et al., 2014; ALBRECHT et al., 2000; FORESEE; HAGAN, 2017).

  A regularização bayesiana minimiza uma combinação linear de erros quadráticos e pesos, sendo capaz de produzir redes com excelentes performances de generalização. Esta regularização Bayesiana ocorre dentro do algoritmo *Levenberg-Marquardt* onde calcula-se a matriz Jacobiana (FORESEE; HAGAN, 2017).

2.6.4 Arquitetura das Redes MLP e RBF

A Figura 10 apresenta a topologia das redes MLP e RBF configurada para DN, onde $\mathbf{x} = [x_1, x_2, x_i, ...x_n]^T$ representa o vetor de atributos, $\phi = [\phi_1, \phi_2, \phi_i, ...\phi_l]^T$ o conjunto de neurônios da camada oculta, $\mathbf{y} = [y_1, y_2, y_i, ...y_n]^T$ o conjunto de neurônios da camada de saída, w_{ln} pesos sinápticos conectados a camada de entrada e camada oculta e w_{kl} pesos sinápticos conectados a camada oculta e camada de saída.

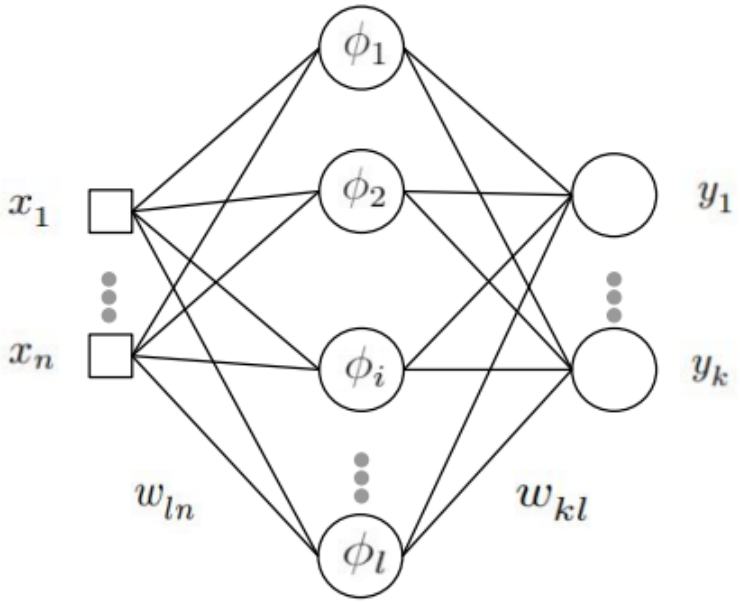

Figura 10 – Topologia das redes MLP e RBF. Fonte: Autor.

Para rede MLP o campo local induzido dos neurônios ocultos é representado pela Equação (2.15), a função de ativação gaussiana dos neurônios ocultos é descrita pela Equação (2.16) e a função dos neurônios de saída é obtida pela Equação (2.17). O campo local induzido do neurônio $v_i(n)$ ajusta a altura da função gaussiana ϕ_i e a constante γ_i ajusta o raio da função.

$$v_i(n) = \sum_{i=1}^{n} w_{ji}(n)x_i(n) \tag{2.15}$$

$$\phi_i(\mathbf{x},\mathbf{w}) = exp\left[-\frac{v_i(n)^2}{\gamma_i^2} \right] \tag{2.16}$$

$$y_k(\mathbf{x},\mathbf{w}) = \sum_{j=1}^{l} w_{kj}\phi_i(\mathbf{x},\mathbf{w}) \tag{2.17}$$

Para rede RBF o campo local induzido dos neurônios ocultos é representado pela Equação (2.18), a função de ativação gaussiana dos neurônios ocultos é descrita pela Equação (2.19) e a função dos neurônios de saída é obtida pela Equação (2.20). O campo

local induzido do neurônio $v_i(n)$ ajusta a função gaussiana no centro dos dados observados. Este processo ocorre pelo cálculo da distância euclidiana $\parallel x_i(n) - c_i(n) \parallel$, considerando o vetor de entrada $\mathbf{x} = [x_1, x_2, x_i, ...x_n]^T$ e o centro dos dados observados $\mathbf{c} = [c_1, c_2, c_i, ...c_n]^T$. O desvio padrão σ_i ajusta o raio da função gaussiana.

$$v_i(n) = \parallel x_i(n) - c_i(n) \parallel \tag{2.18}$$

$$\phi_i(\mathbf{x},\mathbf{c}) = exp\left[- \frac{v_i(n)^2}{2\sigma_i^2} \right] \tag{2.19}$$

$$y_k(\mathbf{x},\mathbf{c}) = \sum_{j=1}^{l} w_{kj}\phi_i(\mathbf{x},\mathbf{c}) \tag{2.20}$$

A utilização da função *softmax* descrita pela Equação (2.21) geralmente é empregada para normalizar os neurônios de saída das redes em uma distribuição de probabilidades para um problema de multi-classes, no entanto a função *softmax* pode ser utilizada como uma estratégia para DN. A visualização dos valores de ativação dos neurônios de saída da rede de forma probabilística possibilita distinguir com maior precisão o valor de referência para DN e consequentemente para criar limiares confiáveis (SAMEER; MARKOU, 2004).

$$softmax(y_i) = \frac{e^{y_i}}{\sum_k e^{y_k}} \tag{2.21}$$

Uma alternativa para DN pode ser alcançada avaliando a distância do neurônio vencedor com um valor de referência definido pelo usuário no teste com os vetores normais (AUGUSTEIJN; FOLKERT, 2002). A Equação (2.22) representa uma nova saída usada para DN, onde $\delta(n) = [\delta_1, \delta_2, \delta_i, ...\delta_k]^T$ corresponde ao vetor de limiar para DN de cada classe. De forma que uma novidade é identificada quando todas as saídas $z(y_i)$ simultaneamente possuem valores menores que zero.

$$z(y_i) = softmax(y_i) - \delta_i \tag{2.22}$$

O valor de referência δ_i predefinido pelo usuário provém do processo de teste da rede treinada com os vetores utilizados na fase de treinamento juntamente com o teste de vetores nunca antes vistos pela rede. Este processo de teste revela o valor de ativação de referência para os neurônios de saída para padrões normais e também mostra os diferentes níveis de ativação dos neurônios de saída para vetores desconhecidos.

As Equações (2.23), (2.24), (2.25) e (2.26) são computadas para classificar os estados afetivos da face e também para detectar novos eventos com uso das redes neurais MLP e RBF.

$$z_1(y_1) = softmax(y_1) - \delta_1 \tag{2.23}$$

$$z_2(y_2) = softmax(y_2) - \delta_2 \tag{2.24}$$

$$z_3(y_3) = softmax(y_3) - \delta_3 \tag{2.25}$$

$$z_4(y_4) = softmax(y_4) - \delta_4 \tag{2.26}$$

3 Materiais e Métodos

Neste capítulo apresenta-se as métricas utilizadas para avaliação de desempenho do algoritmo de DN aplicada no reconhecimento de expressões faciais em fluxo de vídeo. Descreve-se a respeito dos recursos de *software e hardware* utilizados no trabalho. Apresenta-se características técnicas do banco de dados e sua respectiva organização. Por fim, descreve-se acerca do processo realizado para produção do algoritmo proposto com a integração dos algoritmos VJ, KLT, PCA e ANN.

3.1 Avaliação de Desempenho do Algoritmo

As métricas usadas para avaliação de desempenho do algoritmo são a Acurácia de Classificação (ACC), Erro Quadrático Médio, do inglês *Mean Square Error* (MSE), taxa de DN (TDN) e taxa de FPS.

- **Acurácia de Classificação (ACC)**

  A acurácia de classificação geral proveniente da matriz de confusão é a forma padrão de relatar a precisão de um sistema de classificação, de modo que uma matriz de confusão possibilita a inspeção visual completa da alocação dos acertos e erros. As células diagonais mostram o total de acertos e as células fora da diagonal contêm o total de erros ou confusões, onde o total de exemplos é representado pelo total de acertos com total de erros (Ariza-Lopez; Rodriguez-Avi; Alba-Fernandez, 2018). A ACC (%) revela a taxa de sucesso para tarefa de classificação dos estados afetivos da face em fluxo de vídeo. A Equação (3.1) é empregada no cálculo da acurácia de classificação, onde TA corresponde ao total de acertos e TE total de exemplos.

$$ACC(\%) = \frac{TA}{TE} \tag{3.1}$$

- ***Mean Square Error* (MSE)**

  No processo de aprendizado das ANN geralmente usa-se a função do MSE como função de custo (Sai; Jinxia; Zhongxia, 2009), nesta função verifica-se o quão próximas as estimativas ou previsões estão dos valores pretendidos.

  O MSE é usado juntamente com ACC (%) para identificar as redes MLP e RBF com maior potencial para DN em fluxo de vídeo. O menor MSE proporciona uma precisão relativa para os níveis de ativação dos neurônios da camada de saída das redes MLP e RBF, viabilizando a construção de limiares com maior precisão para DN.

- **Taxa de Detecção de Novidades (TDN)**

  A Equação (3.2) possibilita avaliar o algoritmo proposto acerca da taxa de acertos na tarefa de DN, onde TFN representa o total de *frames* identificados como novos e TF o total de *frames*.

$$TDN(\%) = \frac{TFN}{TF} \tag{3.2}$$

- **Taxa de *Frames Por Second* (FPS)**

  A taxa de FPS é empregada para avaliar as redes MLP e RBF no conjunto de dados de treinamento e teste para DN. Estes resultados revelam a rede que apresenta o melhor desempenho para resposta em tempo real.

3.2 Recursos do Trabalho

No desenvolvimento deste trabalho os experimentos foram realizados com o uso do *software* MATLAB 2018.a e um *notebook* Lenovo com Processador: Intel Core i5-7200U CPU @ 2.70 GHz com 8 GB de RAM, Placa de vídeo: Intel HD *Graphics* 620 e Sistema Operacional: 64 bits.

3.3 Base de Dados do Trabalho

Nesta seção descreve-se sobre o conjunto de vídeos da base de dados Livingstone (2018) e apresenta-se detalhes da base de dados utilizada no trabalho para as etapas de treinamento e teste para DN com as redes MLP e RBF.

3.3.1 Base de Dados RAVDESS

A classificação das emoções a partir de expressões faciais avançou rapidamente na última década, impulsionada por tecnologias inteligentes de baixo custo e amplo interesse de pesquisadores em neurociência, psicologia, psiquiatria, audiologia e ciência da computação. Para atender a essas necessidades, a base de dados *The Ryerson Audio-Visual and Song* (RAVDESS) (LIVINGSTONE, 2018) usada nesta pesquisa, conta com uma seleção dinâmica e multimodal de expressões faciais e vocais em inglês da América do Norte avaliando a autenticidade emocional de 24 atores profissionais (12 mulheres e 12 homens). As emoções calma, felicidade, tristeza, raiva, medo, surpresa e nojo estão incluídas. Cada expressão é selecionada em dois níveis de excitação emocional (normal e forte) com uma expressão neutra adicional (LIVINGSTONE, 2018; Abdullah; Ahmad; Han, 2020).

As gravações foram realizadas em estúdio profissional de forma individual, com câmera *Sony Handycam* HDR-SR11 de 1080i com resolução de 1920x1080 *pixels* a 30 fps.

Os atores foram iluminados por lâmpadas fluorescentes de teto e três lâmpadas 28W 5200k CRI 82, instaladas em refletores de 10” com guarda-chuvas parabólicos brancos de 38”. Esta composição promoveu iluminação de espectro total, minimizando sombras faciais (LIVINGSTONE, 2018).

As gravações de voz foram capturadas por um microfone condensador de tubo a vácuo Rode NTK, equipado com um filtro pop *Stedman proscreen* XL, colocado a 20 cm do ator. A saída do microfone foi gravada usando *Pro Tools* 8 e uma estação de mixagem *Digidesign* 003, a uma taxa de amostragem de 48 kHz, 16 bits (LIVINGSTONE, 2018).

A base de dados RAVDESS contém um total de 7356 gravações produzidas por 24 atores, estes arquivos são disponibilizados gratuitamente a partir de uma licença *Creative Commons* e estão disponíveis para serem baixados em três formatos de dados: somente áudio (16 bits, 48 kHz, tipo .wav), áudio-vídeo (1280x720 *pixels*, taxa de 30 fps, AAC 48 kHz, tipo .mp4) e somente vídeo (sem som)(LIVINGSTONE, 2018).

3.3.2 Base de Dados do Trabalho

A base de dados do trabalho é constituída por um conjunto de 40 vídeos, sendo 16 vídeos destinados para fase de treinamento e 24 vídeos reservados para etapa de DN. Os vídeos da fase de treinamento são organizados em 12 vídeos de treino e 04 vídeos para o teste de validação cruzada *Holdout*. Os vídeos para DN são particionados igualmente em vídeos com expressões faciais semelhantes as usadas na fase de treinamento e vídeos com expressões faciais de novos atores. A Figura 11 apresenta o particionamento da base de dados com vídeos de treinamento e vídeos para DN.

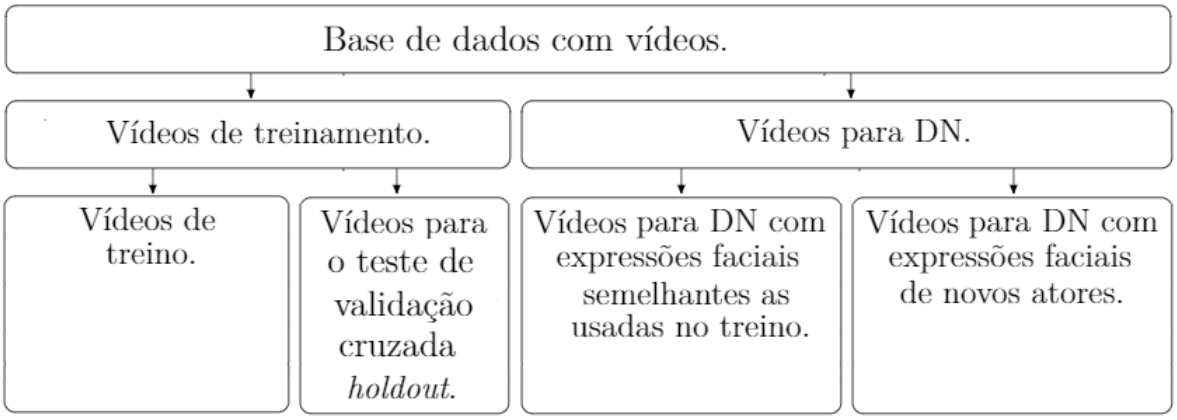

Figura 11 – Base de dados com vídeos de treinamento e vídeos para DN. Fonte: Autor.

3.3.3 Vídeos de Treinamento

A Figura 12 mostra a seleção de trechos de vídeos do ator 1 usados no treinamento das redes MLP e RBF com os estados afetivos feliz, triste, raiva e neutro da base de dados Livingstone (2018).

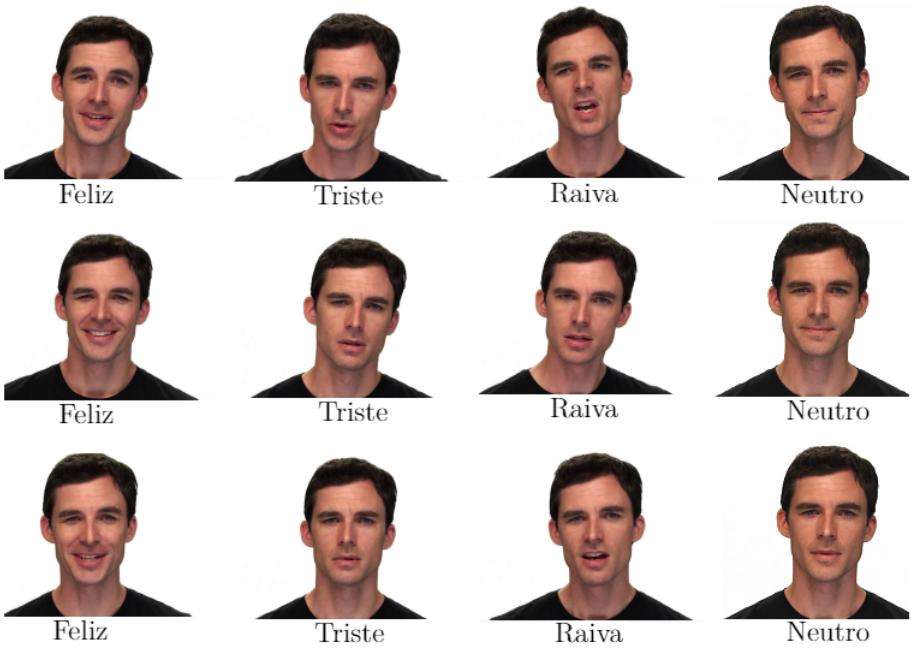

Figura 12 – Trechos de vídeos do ator 1 com os estados afetivos feliz, triste, raiva e neutro usados na fase de treinamento das rede MLP e RBF. Fonte: Autor.

A Tabela 1 mostra a seleção dos recursos de vídeos do ator 1 da base de dados Livingstone (2018) utilizados na fase de treinamento das redes MLP e RBF descrevendo o estado afetivo e total de *frames*.

Tabela 1 – Recursos de vídeos do ator 1 para uma taxa de 30 fps usados na fase de treinamento e teste de validação cruzada *Holdout* das redes MLP e RBF.

Vídeos	Treino ou Teste	Estado Afetivo	Total *Frames*
Vídeo 1	Treino	Feliz	103
Vídeo 2	Treino	Triste	132
Vídeo 3	Treino	Raiva	142
Vídeo 4	Treino	Neutro	97
Vídeo 5	Treino	Feliz	103
Vídeo 6	Treino	Triste	106
Vídeo 7	Treino	Raiva	104
Vídeo 8	Treino	Neutro	99
Vídeo 9	Treino	Feliz	105
Vídeo 10	Treino	Triste	105
Vídeo 11	Treino	Raiva	117
Vídeo 12	Treino	Neutro	98
Vídeo 13	**Teste**	Feliz	133
Vídeo 14	**Teste**	Triste	110
Vídeo 15	**Teste**	Raiva	115
Vídeo 16	**Teste**	Neutro	98

3.3.4 Vídeos para DN

A Figura 13 mostra a seleção de trechos de vídeos do ator 1 usados na fase de teste das redes MLP e RBF para DN com expressões faciais semelhantes as usadas na fase de treinamento com os estados afetivos calmo, medo, surpreso e nojo da base de dados Livingstone (2018).

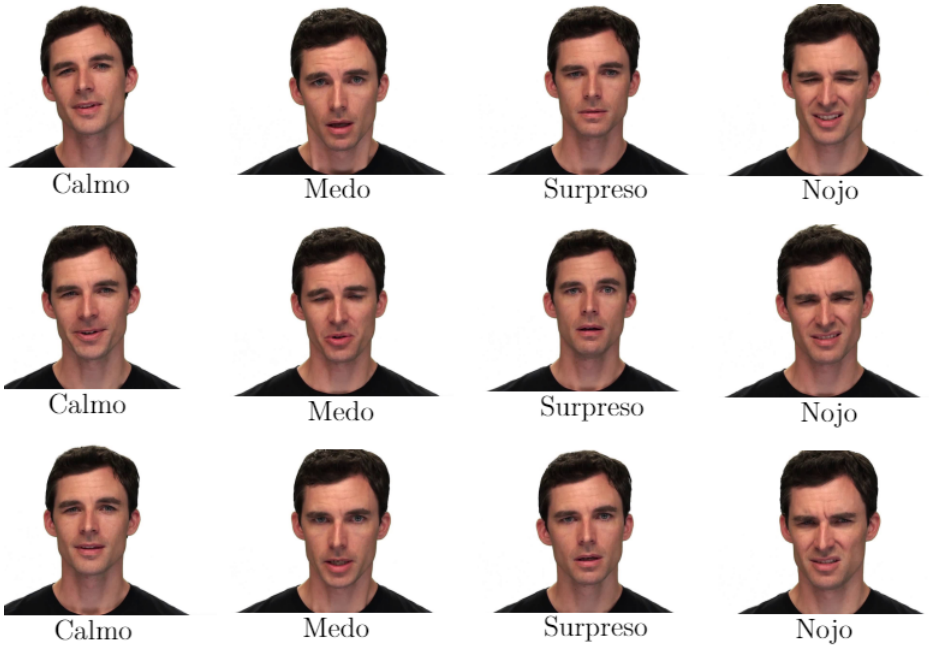

Figura 13 – Trechos de vídeos do ator 1 com os estados afetivos calmo, medo, surpreso e nojo usados na fase de teste para DN. Fonte: Autor.

A Tabela 2 mostra a seleção dos recursos de vídeos do ator 1 da base de dados Livingstone (2018) usados na fase de teste para DN com as redes MLP e RBF descrevendo o estado afetivo e total de *frames*.

Tabela 2 – Recursos de vídeos do ator 1 para uma taxa de 30 fps usados na fase de teste das redes MLP e RBF para DN.

Vídeos Novidade	Estado Afetivo	Total *Frames*
Vídeo 1	Calmo	105
Vídeo 2	Calmo	107
Vídeo 3	Calmo	104
Vídeo 4	Medo	109
Vídeo 5	Medo	108
Vídeo 6	Medo	102
Vídeo 7	Surpreso	97
Vídeo 8	Surpreso	101
Vídeo 9	Surpreso	97
Vídeo 10	Nojo	115
Vídeo 11	Nojo	116
Vídeo 12	Nojo	116

A Figura 14 mostra a seleção de trechos de vídeos usados na fase de teste das redes MLP e RBF para DN com expressões faciais inteiramente novas com a atriz 2 feliz, ator 5 triste, atriz 6 calma e atriz 10 com raiva da base de dados Livingstone (2018).

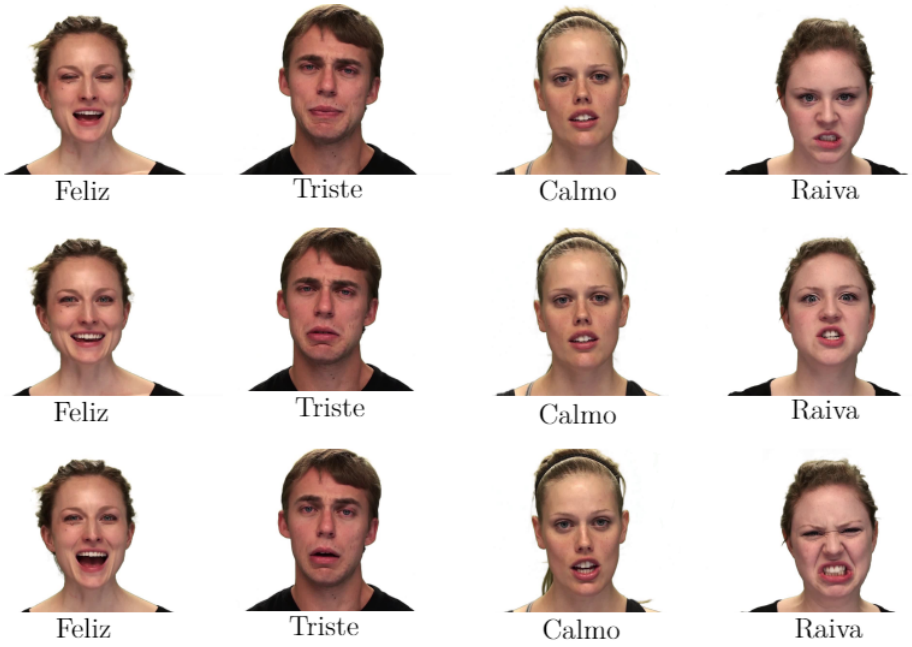

Figura 14 – Trechos de vídeos de novos atores usados na fase de teste para DN com as expressões faciais feliz, triste, calmo e raiva. Fonte: Autor.

A Tabela 3 mostra a seleção dos recursos de vídeos de novos atores da base de dados Livingstone (2018) usados na fase de teste para DN com as redes MLP e RBF descrevendo o estado afetivo e total de *frames*.

Tabela 3 – Recursos de vídeos de novos atores para uma taxa de 30 fps usados na fase de teste das redes MLP e RBF para DN.

Vídeos Novidade	Ator e Estado Afetivo	Total *Frames*
Vídeo 13	2 Feliz	134
Vídeo 14	2 Feliz	138
Vídeo 15	2 Feliz	138
Vídeo 16	5 Triste	129
Vídeo 17	5 Triste	123
Vídeo 18	5 Triste	148
Vídeo 19	6 Calmo	139
Vídeo 20	6 Calmo	139
Vídeo 21	6 Calmo	138
Vídeo 22	10 Raiva	144
Vídeo 23	10 Raiva	136
Vídeo 24	10 Raiva	136

3.4 Modelo Proposto para DN.

A Figura 15 apresenta o modelo aplicado para detecção de novidades no reconhecimento de expressões faciais em fluxo de vídeo.

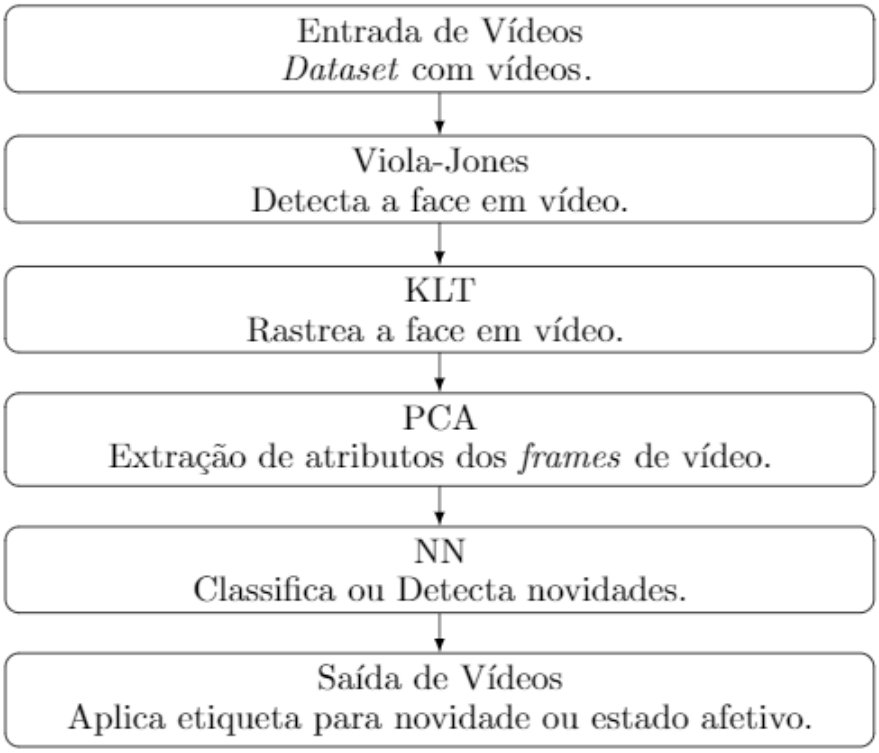

Figura 15 – Modelo aplicado para detecção de novidades no reconhecimento de expressões faciais em fluxo de vídeo. Fonte: Autor.

O modelo conta com a seguinte estrutura:

- **Entrada de Vídeos.**

  Vídeos de treinamento são utilizados para o teste de classificação dos estados afetivos. Vídeos para DN com expressões faciais similares são empregadas na fase de teste para verificar a taxa de acertos das redes MLP e RBF em identificar novidades quando o vetor de atributos produz pequenos distúrbios nas entradas das redes. Vídeos integralmente novos são utilizados na fase de teste para avaliar as redes MLP e RBF na tarefa de DN quando os vetores de atributos produzem distúrbios grosseiros.

- **Pré-processamento.**

  A etapa de pré-processamento é constituída da integração e sinergia dos algorítimos VJ, KLT e PCA. Os vídeos presentes na base de dados possuem dimensão de 1280x720 *pixels*, durante a execução do algoritmo o vídeo é redimensionado para

dimensão de 360x640 *pixels*. O algoritmo VJ detecta a face no primeiro *frame* de vídeo. O algoritmo KLT rastreia a face ao longo dos *frames* com base nos bons recursos que são delimitados dentro da área segmentada. A face é segmentada a cada *frame* com uma janela de dimensão de 80x80 *pixels*, sendo esta janela adequada para segmentação integral da face para um vídeo com exibição de 360x640 *pixels*. Para todos os *frames* do vídeo em análise calcula-se os coeficientes do primeiro componente do PCA com base SVD da matriz de covariância da imagem segmentada da face. Por fim, a imagem segmentada e compactada com PCA a cada *frame* é transformada em um vetor com 6400 atributos e posteriormente o vetor de atributos é apresentado para entrada da rede.

- **Processamento.**

  Na fase de processamento as redes neurais MLP e RBF já treinadas para DN e para classificar os estados afetivos do ator 1 (feliz, triste, raiva e neutro) recebem a cada *frame* o vetor com 6400 atributos já compactos pelo PCA.

- **Classificação**

  Para condição de classificação dos estados afetivos feliz, triste, raiva e neutro do ator 1, apenas uma das saídas da rede $z_1 = [softmax(y_1) - \delta_1]$, $z_2 = [softmax(y_2) - \delta_2]$, $z_3 = [softmax(y_3) - \delta_3]$ e $z_4 = [softmax(y_4) - \delta_4]$ apresenta o valor de $z > 0$, onde $\delta = [\delta_1, \delta_2, \delta_3$ e $\delta_4]$ faz referência ao limiar de decisão para DN de cada classe em estudo.

- **Detecção de Novidade**

  Para condição de DN todas as saídas da rede $z_1 = [softmax(y_1) - \delta_1]$, $z_2 = [softmax(y_2) - \delta_2]$, $z_3 = [softmax(y_3) - \delta_3]$ e $z_4 = [softmax(y_4) - \delta_4]$ apresentam o valor de $z < 0$.

- **Saída de Vídeo**

  Durante a execução do vídeo uma etiqueta com o respectivo estado afetivo da face ou uma etiqueta de novidade é aplicada a cada *frame*, tendo como base a avaliação das saídas $z = [z_1, z_2, z_3$ e $z_4]$.

3.5 Extração de Atributos de Vídeos

Na etapa de extração de atributos um conjunto de 4686 vetores provenientes do total de *frames* respectivos aos vídeos de treinamento e detecção de novidades são obtidos com a integração dos algoritmos VJ, KLT e PCA. A Figura 16 mostra o método empregado na produção dos 4686 vetores de atributos.

Figura 16 – Modelo empregado para extração dos atributos faciais de cada *frame* em fluxo de vídeo com a integração dos algoritmos VJ, KLT e PCA. Fonte: Autor.

O processo realizado para produção dos vetores de atributos com recursos de vídeos conta com as etapas de extração de atributos de treinamento e extração de atributos para teste de DN, onde cada vídeo é apresentado individualmente para o modelo proposto na Figura 16.

- **Extração de Atributos de Treinamento**

  O conjunto de vetores para treino e teste de validação cruzada *Holdout* das redes MLP e RBF são representados por duas matrizes com dimensão de 6400x1311 e 6400x456 referente aos estados afetivos do ator 1 conforme descrito na Tabela 1.

- **Extração de Atributos para Teste de DN**

  O conjunto de vetores usados para teste das redes MLP e RBF na tarefa de DN com faces similares são representados por uma matriz de dimensão 6400x1277 relativo aos estados afetivos do ator 1 conforme descrito na Tabela 2.

  O conjunto vetores usados para teste das redes MLP e RBF na tarefa de DN com faces integralmente novas são representados por uma matriz de dimensão 6400x1642 referente aos estados afetivos de novos atores conforme descrito na Tabela 3.

3.6 Treinamento Rede MLP

No treinamento da rede MLP a camada de entrada possui 6400 nós, a configuração da única camada oculta faz uso da função de ativação gaussiana e a camada de saída possui 4 neurônios. Os pesos sinápticos da rede MLP são ajustados pelo algoritmo de treinamento SCG com o valor da taxa de aprendizagem parametrizado manualmente. A técnica de validação cruzada *Holdout* é empregada com base no total de *frames* descritos

na Tabela 1, onde 74,2% dos *frames* são destinados para treinamento e 25,8% para teste. O treinamento da rede é executado uma única vez com parada programada para o número de épocas, de forma que a rede é parametrizada com 70% dos dados destinados para treinamento, 15% reservados para teste e 15% designados para validação.

A rede MLP é investigada com N experimentações alterando a quantidade de neurônios da camada oculta, neste processo avalia-se o desempenho da rede para ACC (%) e MSE para as etapas de treino e teste de validação cruzada *Holdout*. Na etapa de treino a ACC (%) revela a arquitetura de rede MLP com melhor desempenho para classificação dos estados afetivos da face em fluxo de vídeo e o MSE possibilita a seleção da rede MLP com maior potencial para construção dos limiares para DN. Na fase do teste de validação cruzada *Holdout* a ACC (%) possibilita a identificação da arquitetura de rede com maior desempenho para generalização, afim de limitar problemas com *overfitting* (Yong Liu, 2006; Zhang, 2000).

3.7 Treinamento Rede RBF

No treinamento da rede RBF a camada de entrada possui 6400 nós, a única camada oculta faz uso da função de ativação gaussiana com raio σ_i ajustado manualmente, a camada de saída possui 4 neurônios e os pesos sinápticos da rede são ajustados pelo algoritmo de treinamento de Regularização Bayesiana com a taxa de aprendizagem parametrizada manualmente. A técnica de validação cruzada *Holdout* é executada com base no total de *frames* descritos na Tabela 1, onde 74,2% dos *frames* são destinados para treinamento e 25,8% para teste. O treinamento da rede RBF é executado uma única vez com parada programada para o número de neurônios presentes na camada oculta com meta para MSE igual a zero.

Considerando as métricas de ACC (%) e MSE provenientes das N experimentações alterando a quantidade de neurônios da camada oculta, a seleção da arquitetura de rede RBF considera a melhor projeção para classificação dos estados afetivos da face em fluxo de vídeo, o melhor potencial para construção dos limiares para DN e a maior performance para generalização.

3.8 Limiar de Detecção de Novidades

Os limiares para DN $\delta = [\delta_1, \delta_2, \delta_3$ e $\delta_4]$ são construídos com as redes MLP e RBF que apresentam o máximo desempenho para ACC (%) de treino, o menor MSE de treino, a menor quantidade de neurônios na camada oculta e a maior performance de generalização. As redes com menor quantidade de neurônios ocultos são menos suscetível a apresentar problemas de *overfitting* (Zhang, 2000) e também produzem melhores resultados para

resposta em tempo real, ou seja, maiores taxas de FPS.

A Figura 17 ilustra o processo realizado para obtenção dos limiares para DN com as redes MLP e RBF. Neste processo os limiares para DN são obtidos apresentando para as redes MLP e RBF já treinadas os vetores de atributos referentes aos estados afetivos feliz, triste, raiva e neutro. O limiar de DN para cada estado afetivo é definido identificando dentro do conjunto de neurônios vencedores de cada classe o menor nível de ativação.

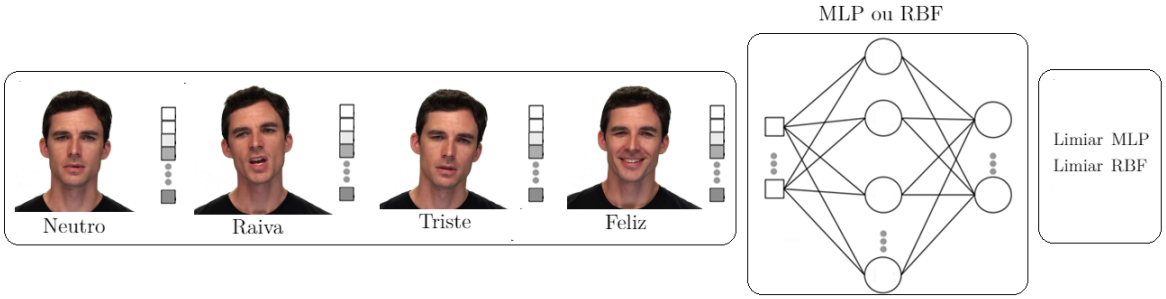

Figura 17 – Processo realizado para definição dos limiares para DN com as redes MLP e RBF. Fonte: Autor

3.9 Teste de Detecção de Novidades

O teste para DN conta com a base de dados descrita nas Tabelas 2 e 3. O processo realizado para DN consiste em apresentar para as redes MLP e RBF já treinadas todos os vetores obtidos com os vídeos da base de dados para DN e posteriormente em verificar a quantidade de saídas da rede onde todas as classes apresentam o valor de $z < 0$, sendo $z_1 = [softmax(y_1) - \delta_1]$, $z_2 = [softmax(y_2) - \delta_2]$, $z_3 = [softmax(y_3) - \delta_3]$ e $z_4 = [softmax(y_4) - \delta_4]$. Em um segundo momento, o teste de DN é realizado apresentando para o algoritmo proposto da Figura 15 os vídeos descritos nas Tabelas 2 e 3.

4 Resultados e Discussão

Neste capítulo apresenta-se resultados para detecção de novidades aplicada ao reconhecimento de expressões faciais em fluxo de vídeo. Destaca-se resultados com a extração de atributos faciais, treinamento das redes MLP e RBF, obtenção dos limiares para DN, resposta em tempo real segundo a taxa de FPS e por último a previsão do custo computacional com base no tempo de computação e quantidade de memória requerida pelos algoritmos com uso das redes MLP e RBF.

4.1 Pré-processamento

Na Figura 18, os quadros de vídeos das expressões faciais em escala de RGB são procedentes da segmentação da face em fluxo de vídeo usando os algoritmos VJ e KLT pelo método holístico. As expressões faciais em escala de RGB são segmentadas na dimensão de 80x80 *pixels* para vídeos executados na resolução de 360x640 *pixels*. Os quadros de vídeos das expressões faciais em escala de cinza são obtidos com o cálculo do primeiro eixo do componente principal a partir dos quadros de vídeos em escala de RGB. Os resultados do primeiro componente do PCA produz imagens com uma compressão média de 33% sem perdas significativas dos atributos faciais, consecutivamente possibilitando a produção de um vetor de atributos com uma representação significativa do conjunto de dados originais. Para apresentar os resultados obtidos na fase de pré-processamento com a execução dos algoritmos VJ, KLT e PCA disponibiliza-se o link: <https://youtu.be/Zem6fwn2R1k>.

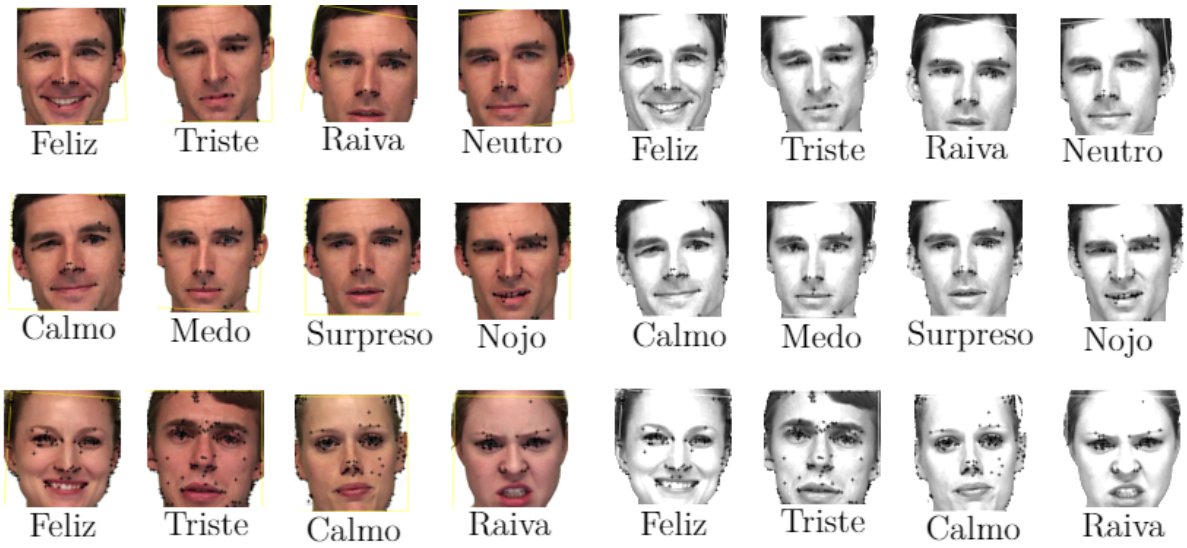

Figura 18 – Quadros de vídeos com expressões faciais em escala de RGB e quadros de vídeos com expressões faciais em escala de cinza. Fonte: Autor.

4.2 Processamento

Na etapa de processamento apresenta-se resultados com a identificação das redes MLP e RBF com maior potencial para construção dos limiares de DN. Expõe-se através das matrizes de confusão detalhes acerca da etapa de classificação. Apresenta-se gráficos justificando a definição dos limiares de DN. Exibe-se trechos de vídeos com os resultados para as etapas de classificação e DN. Apresenta-se gráficos para contabilizar a assertividade das redes MLP e RBF na tarefa de DN. Expõe-se através de tabelas a taxa de FPS para avaliação da base de treino e teste para DN. Por fim, apresenta-se gráficos e tabelas para estimar o custo computacional do algoritmo com uso das redes MLP e RBF para as etapas de DN e classificação.

- **Rede MLP**

A Tabela 4 apresenta 10 experimentações, investigando a rede MLP com até 30 neurônios na camada oculta. O método de validação cruzada *Holdout* é realizado com base nos *frames* da Tabela 1, sendo 74,2% dos *frames* destinados para treino da rede MLP e 25,8% dos *frames* reservados para o teste de validação cruzada. Os resultados de acurácia de treino são obtidos com a rede parametrizada com 70% para dados de treinamento, 15% para dados de validação, 15% para dados de teste, função de ativação gaussiana, algoritmo de treinamento SCG, taxa de aprendizagem com valor padrão de 0.005 e parada programada para 1000 épocas.

Tabela 4 – Identificação da rede MLP com melhor performance para construção dos limites para DN com base no total de neurônios da camada oculta, acurácia de classificação e erro quadrático médio para as etapas de treino e teste.

Neurônios Ocultos	ACC (%) Treino	MSE Treino	Tempo (seg) Treino	ACC (%) Teste	MSE Teste
5	79,5	0,05	5,5	30,1	0,21
7	99,5	0,002	7,2	43,6	0,23
10	99,9	0,002	7,4	44,7	0,23
12	**99,8**	**0,001**	**10,1**	**52,9**	**0,19**
15	99,9	0,001	10,1	47,5	0,21
18	99,8	0,002	10,2	48,2	0,21
20	99,9	0,003	10,2	45,4	0,23
22	99,9	0,001	10,4	51,9	0,21
26	99,9	0,001	10,5	52,4	0,19
30	99,9	0,001	11,4	52,1	0,19

Os resultados descritos na Tabela 4 para rede MLP relacionando ACC (%) vs MSE para as fases de treino e teste segundo a quantidade de neurônios da camada oculta viabilizam a distinção da arquitetura de rede MLP que combina a melhor performance

para construção dos limiares para DN. Parametrizações da taxa de aprendizagem dentro da faixa de 0.1 até 0.0001 não alteram o desempenho do treino para ACC, MSE e Tempo. Neste caso, a escolha da arquitetura com 12 neurônios na camada oculta fundamenta-se: (i) na combinação dos maiores valores de ACC (%), de modo que estes resultados de acurácia revelam a melhor performance para classificação dos estados afetivos da face em fluxo de vídeo e o maior desempenho de generalização; (ii) no menor valor MSE de treino que indica a qualidade de convergência dos valores reais com os valores pretendidos, proporcionando níveis de ativação com menor margem de flutuação na camada de saída da rede e consecutivamente permitindo a definição de limiares consistentes para DN; (iii) na menor quantidade de neurônios presentes na camada oculta que produz maiores taxas de FPS.

A Figura 19 apresenta os resultados da etapa de classificação com a rede MLP configurada com 12 neurônios na camada oculta para classificação dos estados afetivos feliz, triste, raiva e neutro do ator 1 da base de dados Livingstone (2018).

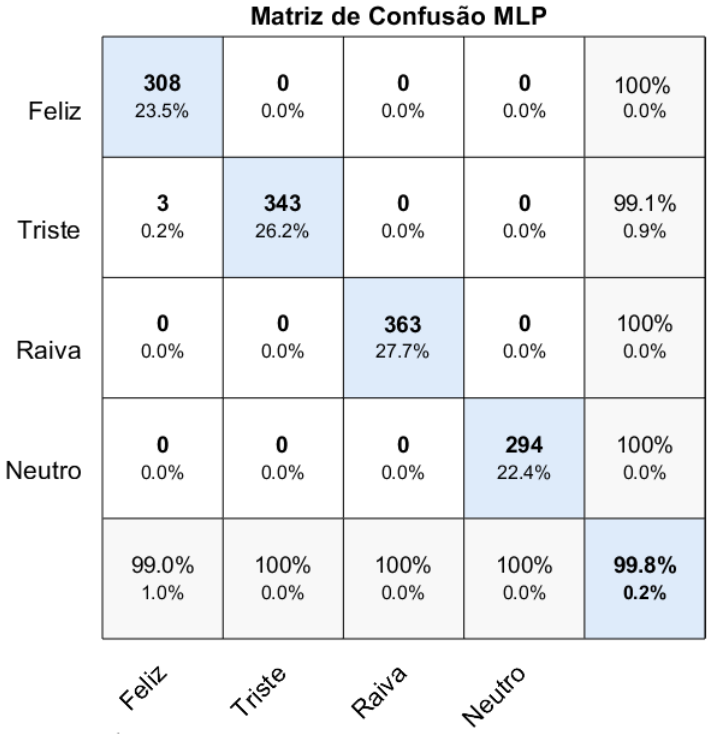

Figura 19 – Matriz de confusão MLP para classificação dos estados afetivos feliz, triste, raiva e neutro do ator 1. Fonte: Autor.

De 1311 *frames*, 1308 representado 99,8% são a proporção correta e 3 representado 0,2% são a proporção incorreta. De 311 casos de feliz, 99,0% são previstos como corretos e 1,0% previsto como incorreto. As confusões ocorrem entre os estados afetivos feliz e triste devido à semelhança de algumas expressões faciais. O resultado alcançado com a matriz de confusão MLP revela o comportamento da rede no processo de classificação dos estados afetivos da face do ator 1 em fluxo de vídeo.

As Figuras 20, 21, 22 e 23 fazem referência aos gráficos gerados para definição dos limiares de DN com a rede MLP. Cada gráfico é obtido apresentando para rede MLP treinada os vetores de atributos respectivos aos estados afetivos feliz, triste, raiva e neutro do ator 1. Os níveis de ativação produzidos pelas saídas da rede MLP para cada estado afetivo em análise expõe o valor de limiar para detecção de novos estados afetivos ou para detecção de novas faces.

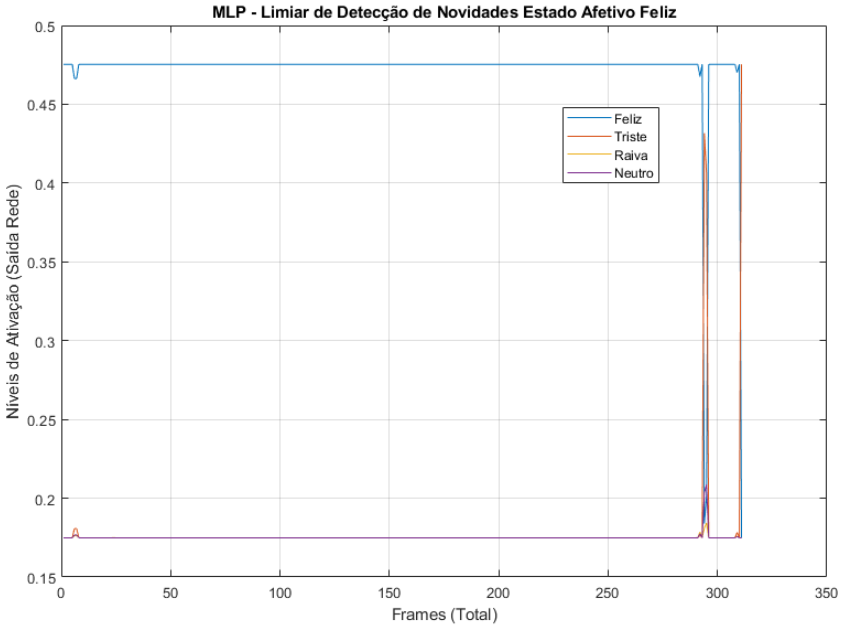

Figura 20 – Níveis de ativação produzidos pelas saídas da rede MLP para definição do limiar de DN do estado afetivo feliz. Fonte: Autor.

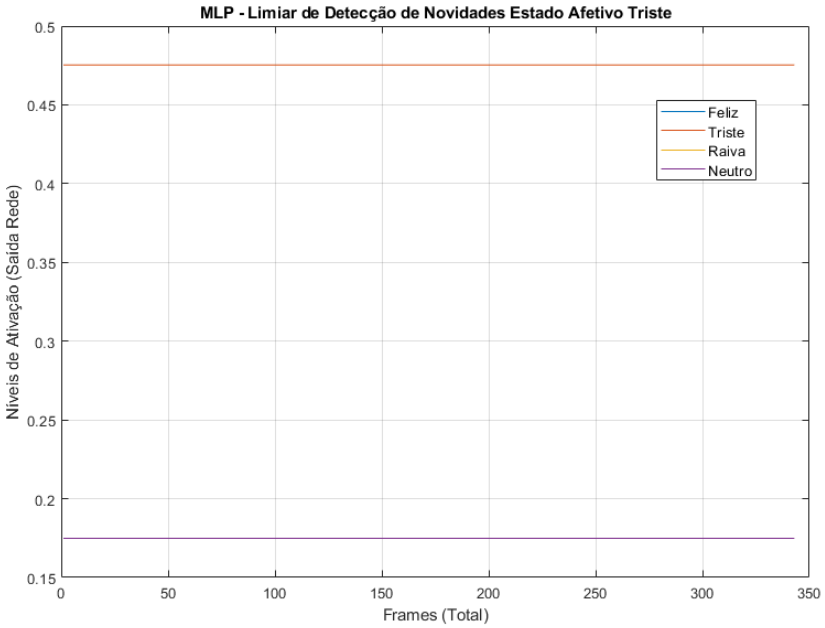

Figura 21 – Níveis de ativação produzidos pelas saídas da rede MLP para definição do limiar de DN do estado afetivo triste. Fonte: Autor.

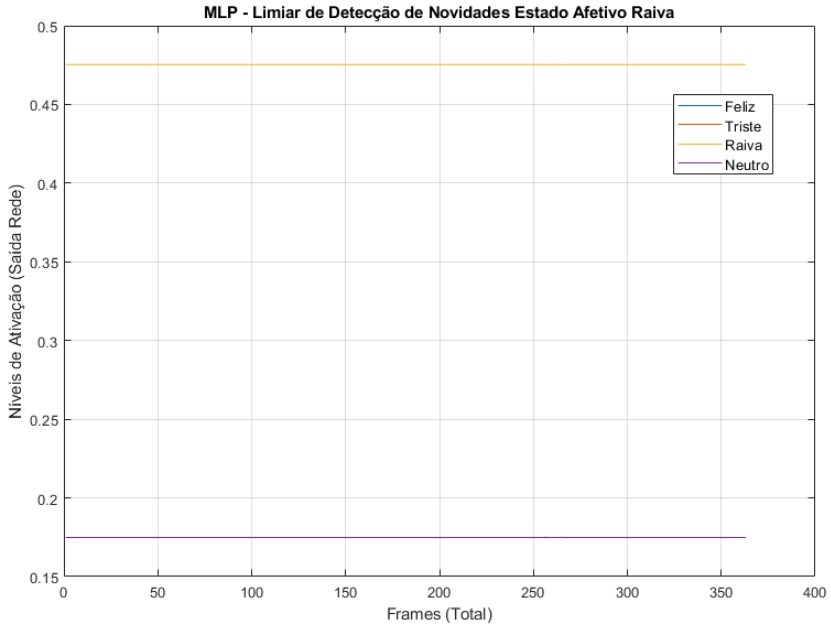

Figura 22 – Níveis de ativação produzidos pelas saídas da rede MLP para definição do limiar de DN do estado afetivo raiva. Fonte: Autor.

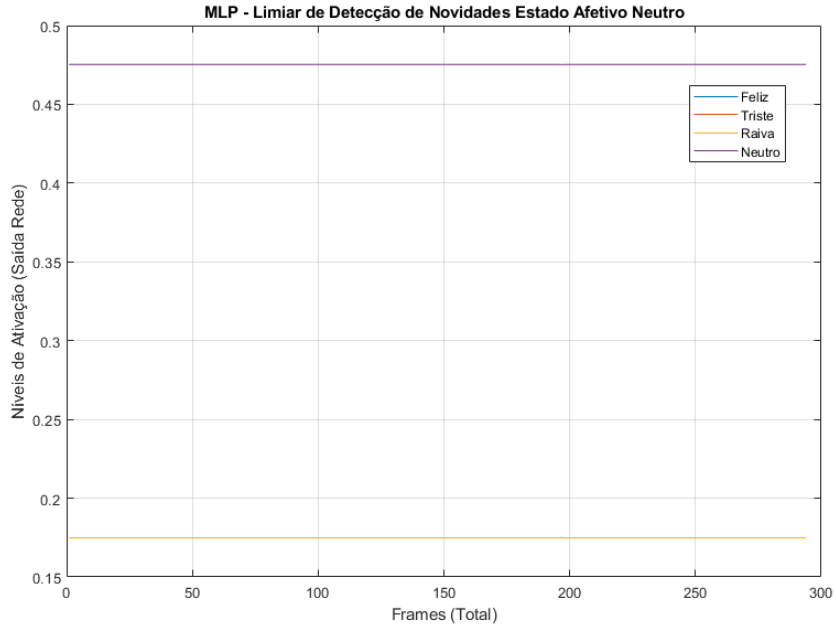

Figura 23 – Níveis de ativação produzidos pelas saídas da rede MLP para definição do limiar de DN do estado afetivo neutro. Fonte: Autor.

A rede MLP apresenta níveis de ativação estáveis para o neurônio vencedor em 98,8% dos dados, de modo que em 0,2% dos dados a rede apresenta descontinuidades abruptas no nível de ativação em função da confusão entre os estados afetivos feliz e triste. Os limiares $\delta = [\delta_1, \delta_2, \delta_3$ e $\delta_4]$ da rede MLP são definidos com valor de 0,46 para δ_1 e com valor de 0,47 para δ_2, δ_3 e δ_4 estes valores permitem a DN para novos vetores de entrada e asseguram a classificação correta dos estados afetivos em fluxo de vídeo.

- **Rede RBF**

Na Tabela 5 os resultados são descritos para 10 experimentações, investigando a arquitetura da rede RBF com até 180 neurônios na camada oculta. O método de validação cruzada *Holdout* é executado com uso no total de *frames* especificado na Tabela 1, sendo 74,2% dos *frames* destinados para treino e 25,8% dos *frames* reservados para teste. Os resultados são obtidos com a rede RBF parametrizada com a função de ativação gaussiana de raio σ_i ajustado com valor de 80, algoritmo de treinamento Regularização Bayesiana, taxa de aprendizagem com valor de 0.1, parada programada para o número de neurônios presentes na camada oculta e meta MSE igual zero.

Tabela 5 – Identificação da rede RBF com melhor performance para construção dos limites para DN com base no total de neurônios da camada oculta, acurácia de classificação e erro quadrático médio.

Neurônios Ocultos	ACC (%) Treino	MSE Treino	Tempo (seg) Treino	ACC (%) Teste	MSE Teste
5	56,6	0,13	62,9	35,1	0,42
15	90,8	0,07	68,7	43,4	0,30
30	98,4	0,02	78,7	42,1	0,27
60	99,9	0,008	139,8	48,7	0,25
90	99,9	0,005	254,9	50,4	0,21
100	**99,9**	**0,002**	**281,2**	**52,2**	**0,16**
120	99,9	0,001	379,5	52,2	0,17
130	99,9	0,001	420,9	51,1	0,19
150	99,9	0,001	490,2	51,1	0,20
180	99,9	0,0009	768,8	51,1	0,25

Na Tabela 5 verifica-se que a rede RBF possui uma performance semelhante a rede MLP para ACC (%), porém difere acerca do desempenho do MSE e também em relação a quantidade de neurônios da camada oculta. Configurações da taxa de aprendizagem dentro da faixa de 0.1 até 0.0001 não alteram o desempenho do treino para ACC e MSE, no entanto a maior taxa de aprendizagem dentro da faixa especificada reduz o tempo de treinamento em média 7 segundos. Para este caso, a rede RBF que combina o melhor resultado para construção dos limiares para DN possui 100 neurônios na camada oculta. Variações significativas no nível de ativação do neurônio vencedor impossibilita a definição

de limiares consistentes para DN, deste modo esta escolha ampara-se principalmente no valor do MSE que assegura níveis de ativação do neurônio vencedor em uma faixa aceitável para definição do limiar para DN.

Os resultados descritos nas Tabelas 4 e 5, validam que o algoritmo SCG acelera a convergência do aprendizado da rede MLP comparado com a rede RBF que utiliza o algoritmo Regulador Bayesiano. O algoritmo SCG juntamente com o algoritmo *Levenberg-Marquardt* são considerados os métodos mais robustos em termos de precisão e velocidade para treinamento da rede MLP (Mishra; Prusty; Hota, 2015), de modo que o método *Levenberg-Marquardt* não treina redes com muitos parâmetros de ajuste (SHINDE; SAYYAD, 2016).

A Figura 24 apresenta os resultados para etapa de classificação com a rede RBF configurada para 100 neurônios na camada oculta para classificação dos estados afetivos feliz, triste, raiva e neutro do ator 1 da base de dados Livingstone (2018).

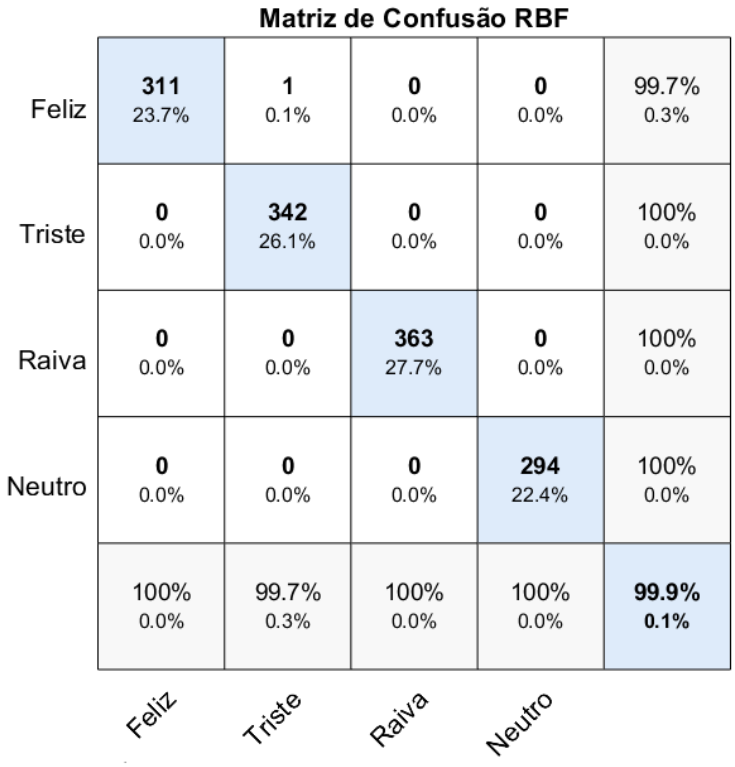

Figura 24 – Matriz de confusão para classificação dos estados afetivos feliz, triste, raiva e neutro do ator 1 com a rede RBF. Fonte: Autor.

De 1311 *frames*, 1310 representado 99,9% são a proporção correta e 1 representado 0,1% é proporção incorreta. De 343 casos de triste, 99,7% são previstos como corretos e 0,3% previsto como incorreto. Uma única confusão ocorre entre os estados afetivos triste e feliz devido à semelhança das expressões faciais. O resultado alcançado com a matriz de confusão RBF revela o comportamento da rede para classificação dos estados afetivos da face do ator 1 em fluxo de vídeo.

As Figuras 25, 26, 27 e 28 fazem referência aos gráficos produzidos para definição dos limiares de DN com a rede RBF. Os gráficos são obtidos apresentando para rede RBF treinada os vetores de atributos respectivos aos estados afetivos feliz, triste, raiva e neutro do ator 1. Os níveis de ativação gerados pelas saídas da rede RBF para cada estado afetivo em estudo expõe o valor de limiar para DN.

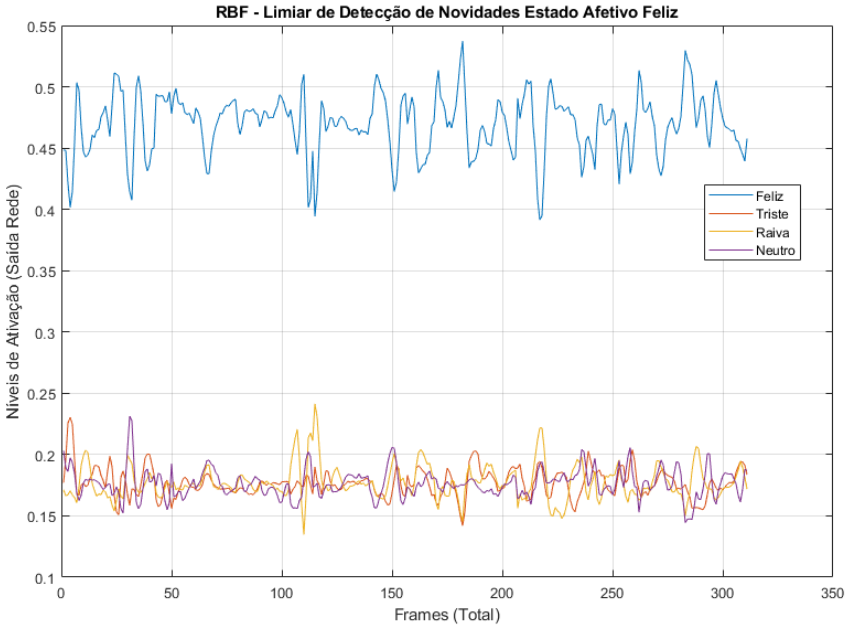

Figura 25 – Níveis de ativação produzidos pelas saídas da rede RBF para definição do limiar de DN do estado afetivo feliz. Fonte: Autor.

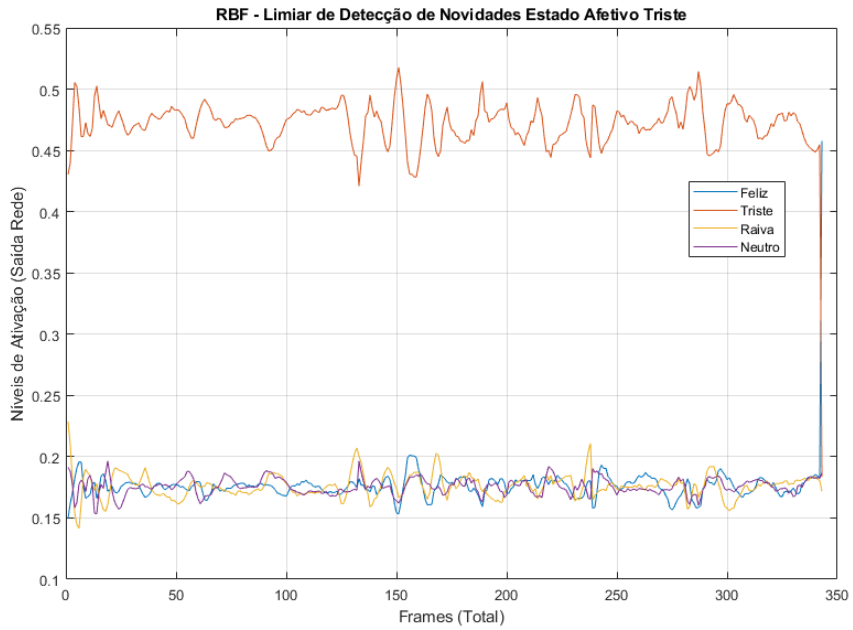

Figura 26 – Níveis de ativação produzidos pelas saídas da rede RBF para definição do limiar de DN do estado afetivo triste. Fonte: Autor.

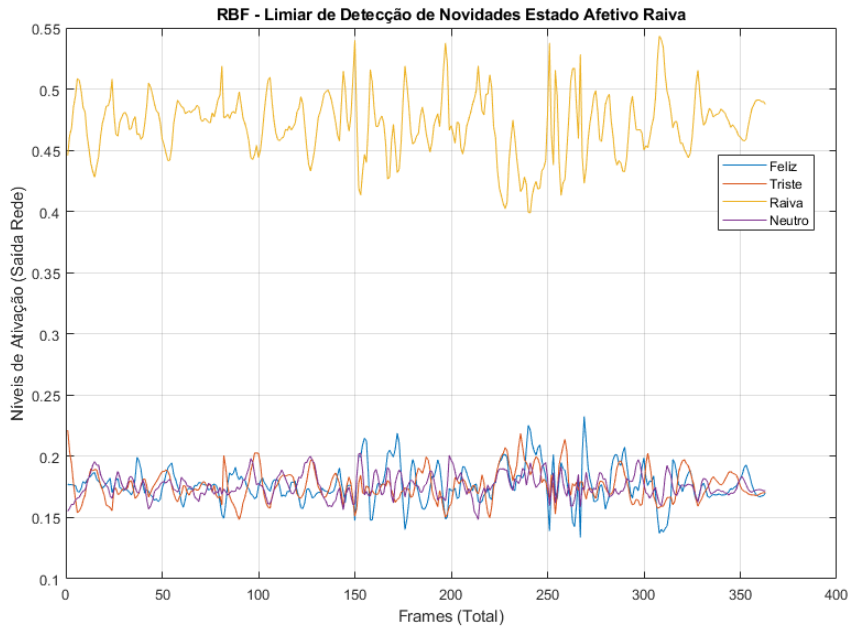

Figura 27 – Níveis de ativação produzidos pelas saídas da rede RBF para definição do limiar de DN do estado afetivo raiva. Fonte: Autor.

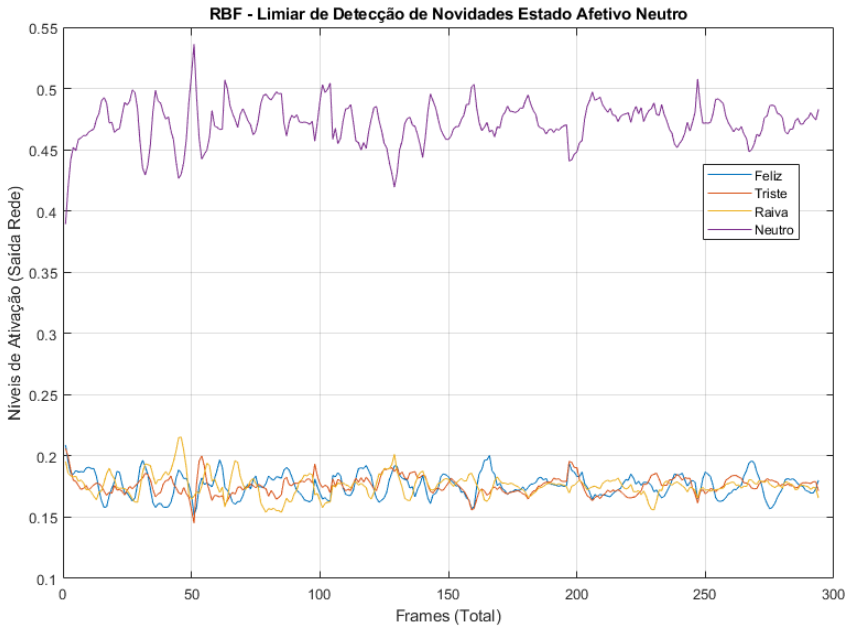

Figura 28 – Níveis de ativação produzidos pelas saídas da rede RBF para definição do limiar de DN do estado afetivo neutro. Fonte: Autor.

A rede RBF apresenta níveis de ativação estáveis para o neurônio vencedor em uma faixa de 0,37 até 0,55 em referência aos estados afetivos feliz, triste, raiva e neutro, de forma que o estado afetivo triste mostra uma descontinuidade abrupta no nível de ativação devido a confusão com o estado afetivo feliz. Os limiares $\delta = [\delta_1, \delta_2, \delta_3$ e $\delta_4]$ da rede RBF são definidos com valor de 0,37 para δ_1 e δ_4, 0,38 para δ_3 e 0,42 para δ_2 estes valores viabilizam a DN para novos padrões de entrada e possibilitam a classificação correta dos estados afetivos em fluxo de vídeo.

- **Classificação dos Estados Afetivos da Face em Fluxo de Vídeo**

As Figuras 29 e 30 apresentam trechos de vídeos com os resultados obtidos para o teste de classificação em fluxo de vídeo com a integração dos algoritmos VJ, KLT, PCA e ANN (MLP ou RBF).

Figura 29 – Trechos de vídeos com a classificação correta para os estados afetivos neutro, feliz, raiva e triste do ator 1 em fluxo de vídeo. Fonte: Autor.

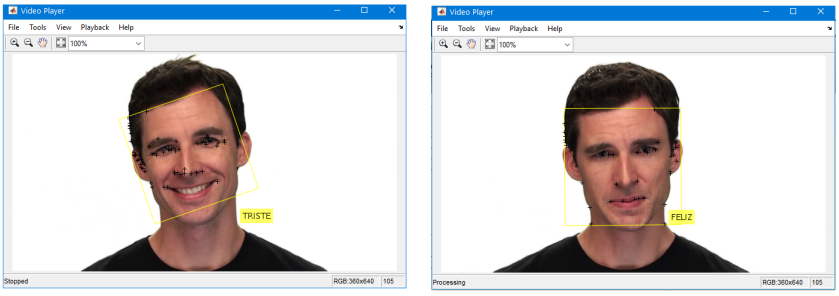

Figura 30 – Trechos de vídeos com a classificação incorreta para os estados afetivos feliz e triste do ator 1 em fluxo de vídeo. Fonte: Autor.

- **Detecção de Novidades com Rede MLP**

  Os resultados apresentados na Figura 31 para DN são obtidos usando a rede MLP com base nos dados descritos nas Tabelas 2 e 3.

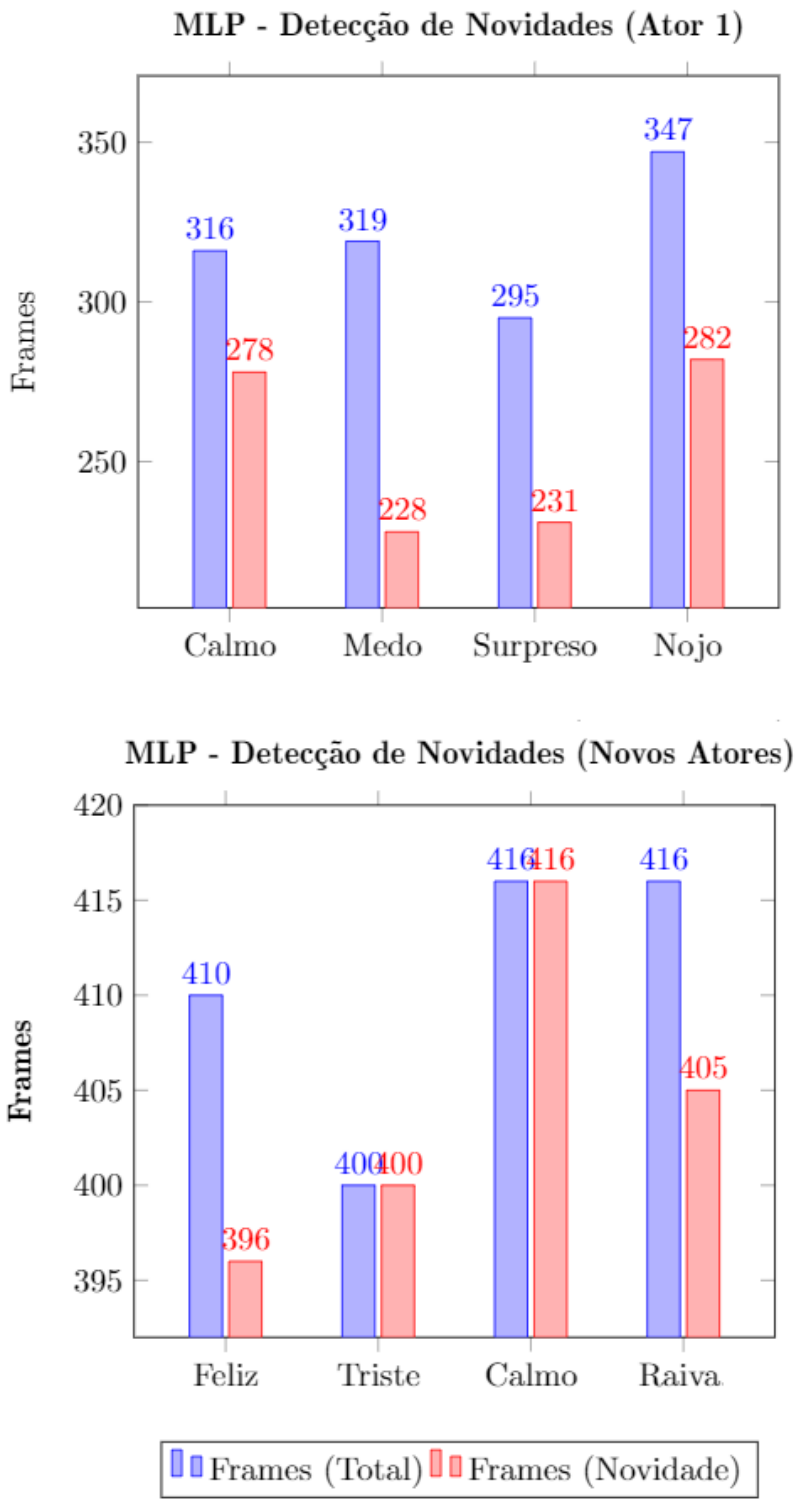

Figura 31 – Detecção de Novidades com uso da Rede MLP. Fonte: Autor.

- **Detecção Novidades com Rede RBF**

  Na Figura 32 apresenta-se os resultados para DN com a rede RBF usando os dados das Tabelas 2 e 3.

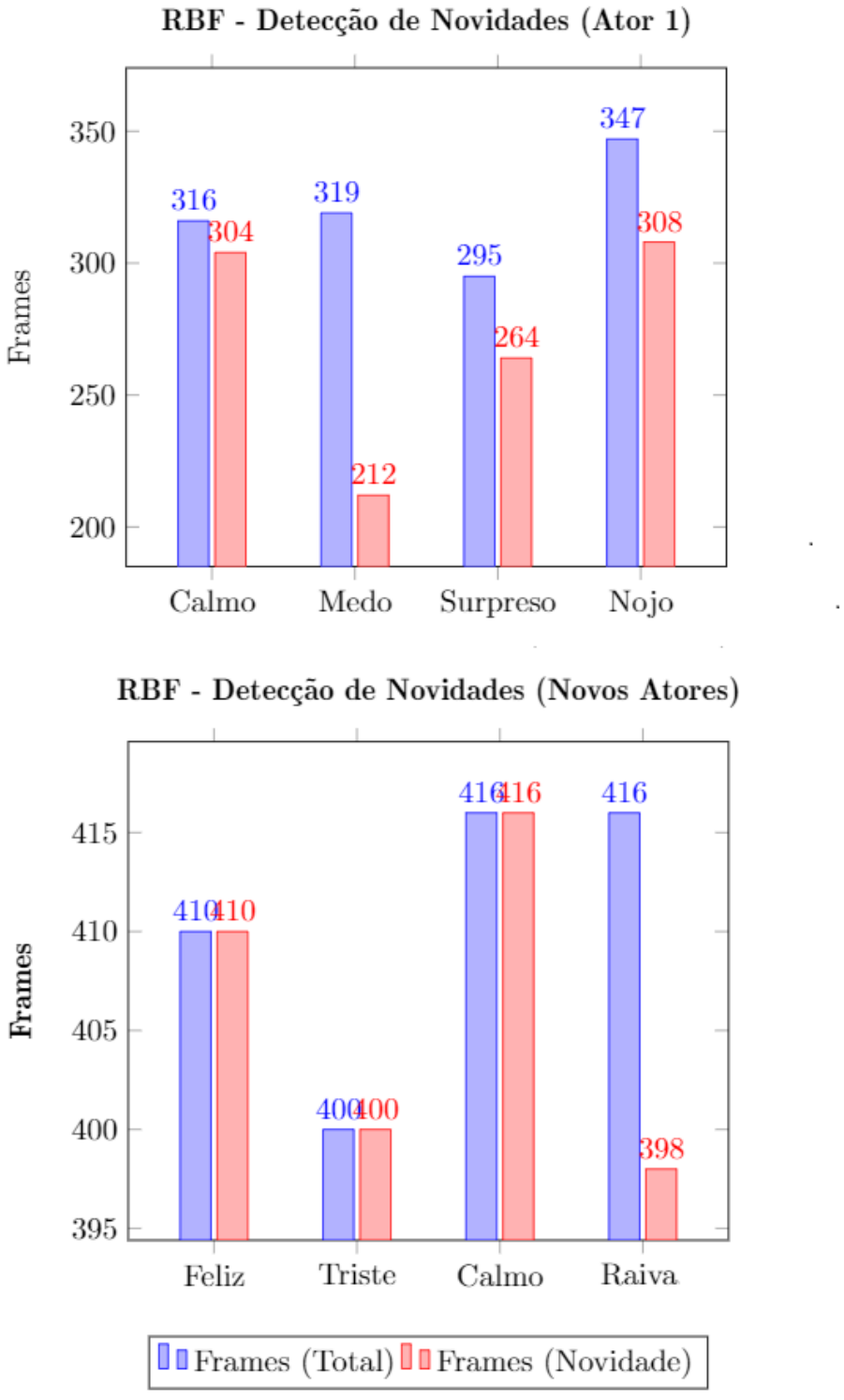

Figura 32 – Detecção de novidades com uso da rede RBF. Fonte: Autor.

- **Detecção Novidades em Fluxo de Vídeo.**

  A Figura 33 apresenta trechos de vídeos com os resultados obtidos para identificação correta no teste de DN em tempo real com a integração dos algoritmos VJ, KLT, PCA e ANN (MLP ou RBF) considerando os estados afetivos do ator 1 e novos atores conforme descrito nas Tabelas 2 e 3. As etiquetas são aplicadas a cada *frame* e a taxa de sucesso na identificação da novidade está condicionada aos resultados apresentados nas Figuras 31 e 32.

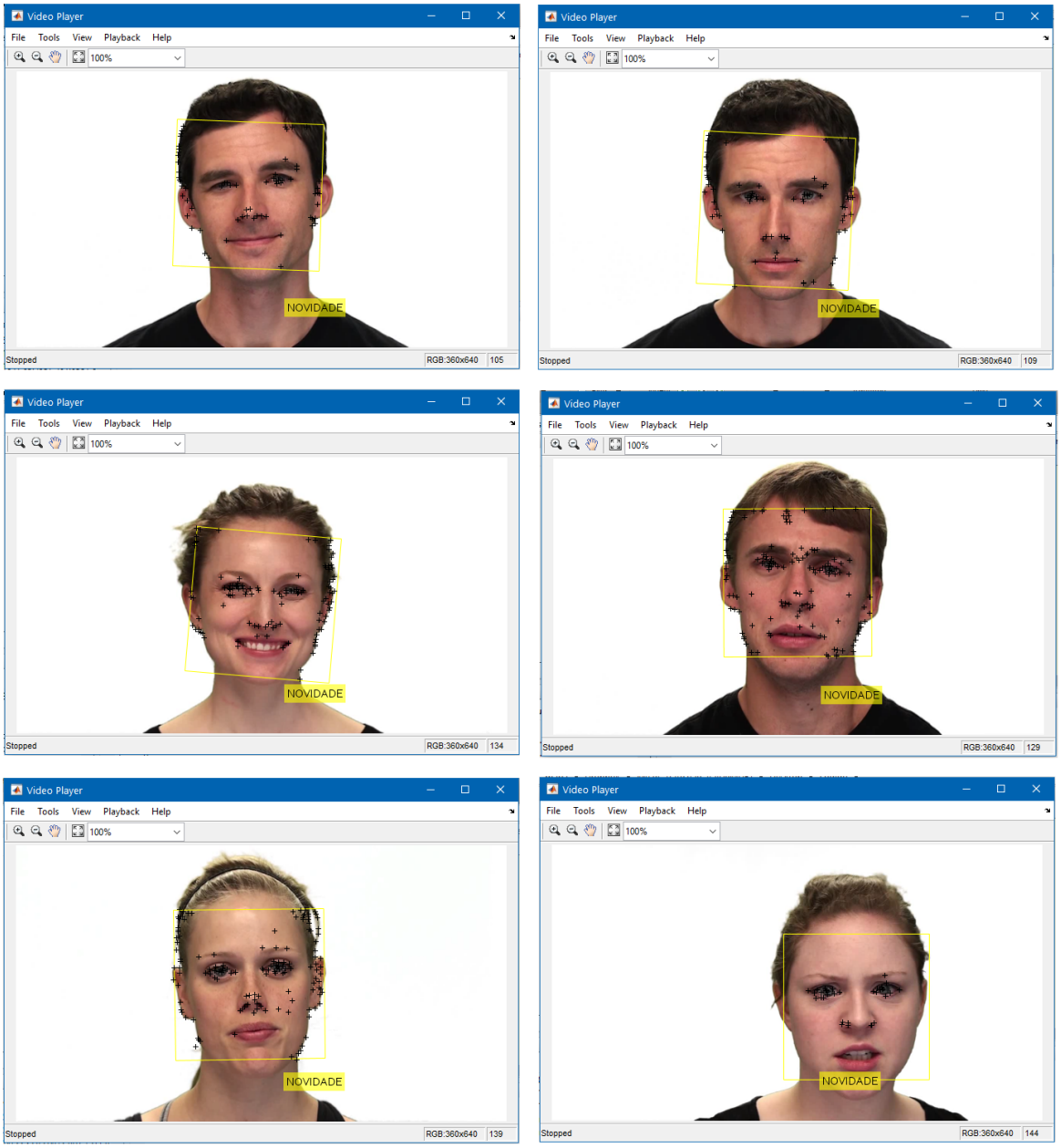

Figura 33 – Trechos de vídeos com o teste de DN em fluxo de vídeo para expressões faciais similares as expressões faciais usadas na fase de treinamento e para expressões faciais de novos atores. Fonte: Autor.

As Figuras 34, 35, 36, 37, 38 e 39 mostram trechos de vídeos com os resultados obtidos para classificação incorreta no teste de DN em fluxo de vídeo com a integração dos algoritmos VJ, KLT, PCA e ANN (MLP ou RBF). As expressões faciais que não são detectadas como novas são classificadas com uma expressão facial da fase de treinamento.

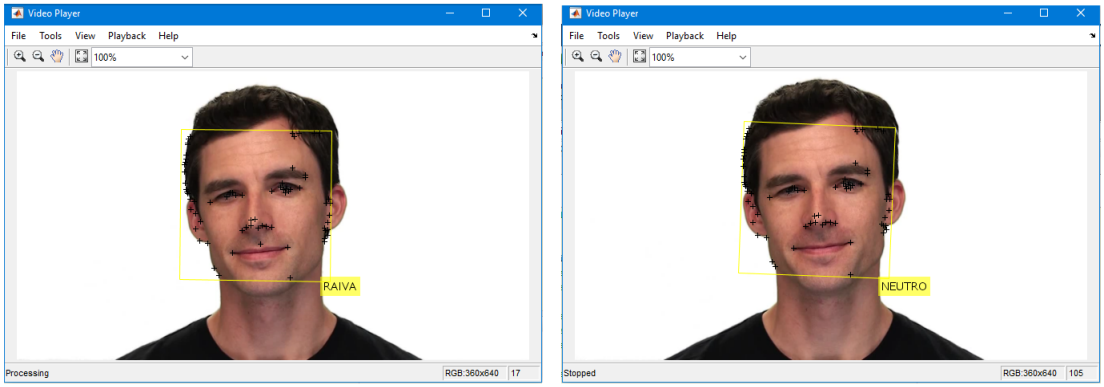

Figura 34 – Trechos de vídeos onde a expressão facial calma é classificada de forma incorreta como raiva ou neutro no teste de DN.

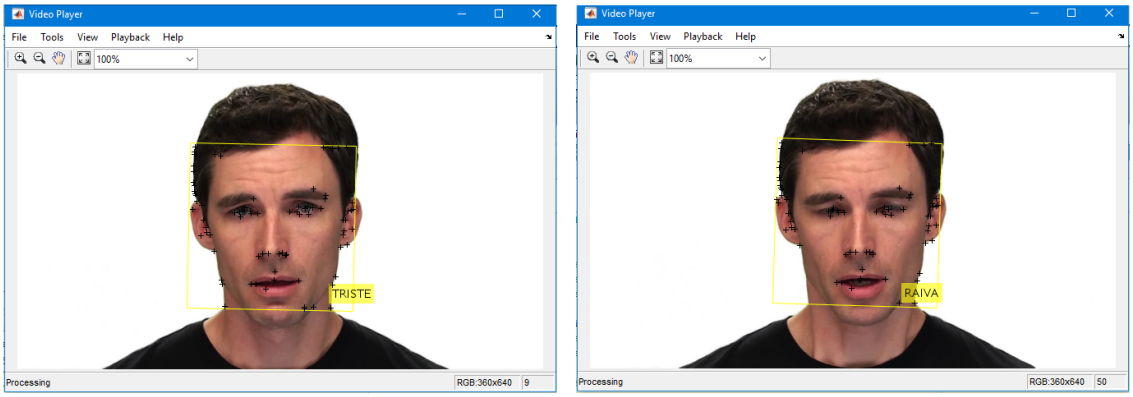

Figura 35 – Trechos de vídeos onde a expressão facial medo é classificada de forma incorreta como triste ou raiva no teste de DN.

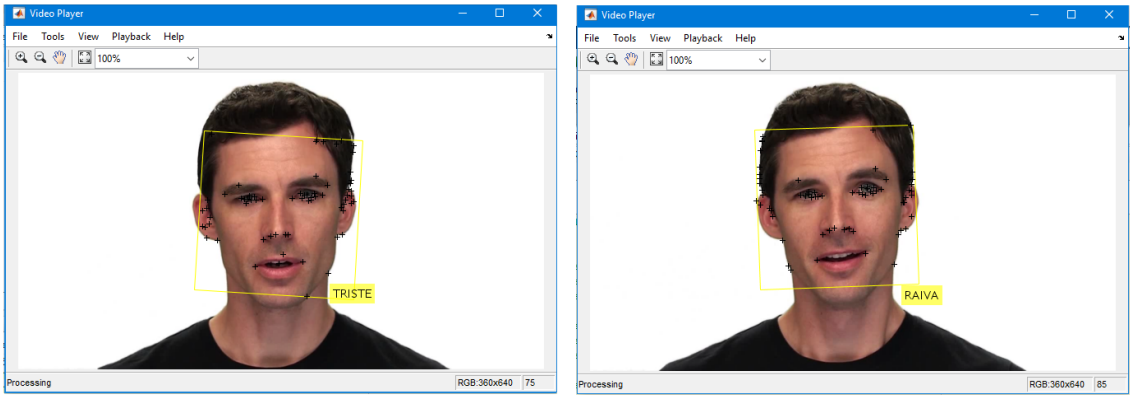

Figura 36 – Trechos de vídeos onde a expressão facial surpreso é classificada de forma incorreta como triste ou raiva no teste de DN.

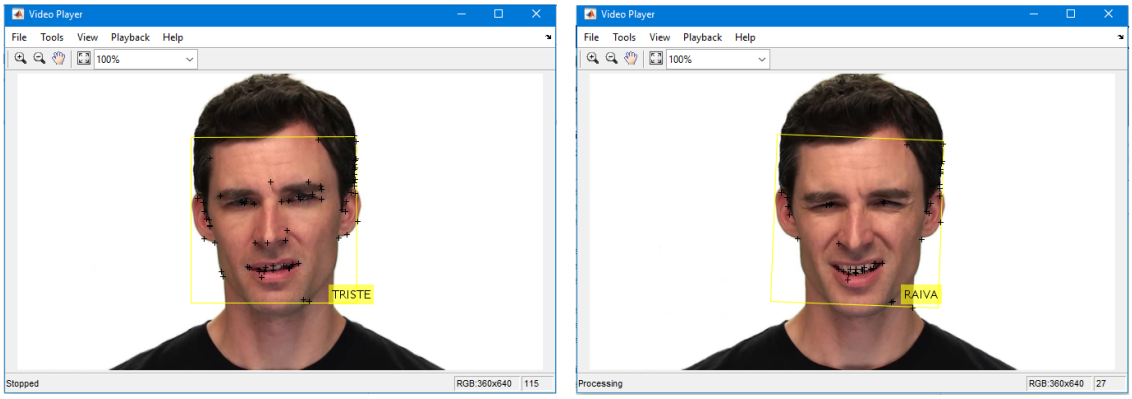

Figura 37 – Trechos de vídeos onde a expressão facial surpreso é classificada de forma incorreta como triste ou raiva no teste de DN.

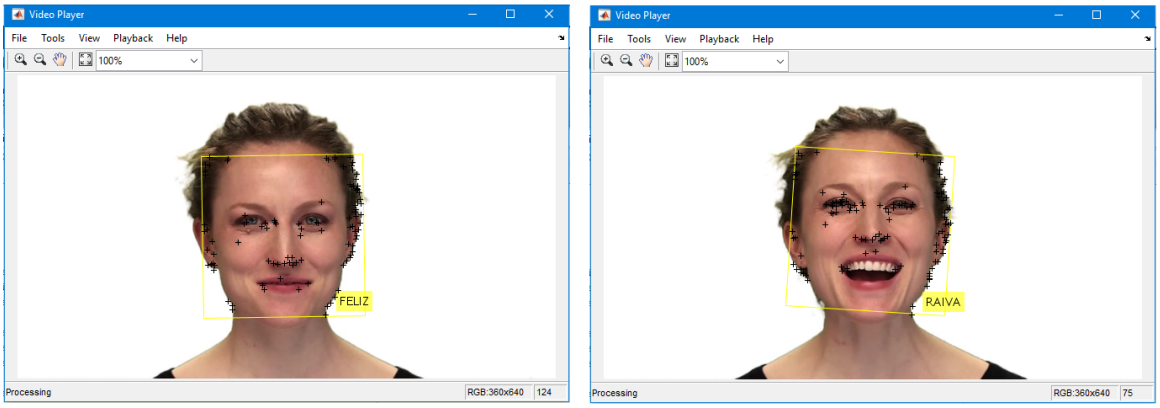

Figura 38 – Trechos de vídeos onde o novo ator é classificado de forma incorreta com a expressão facial feliz e raiva no teste de DN.

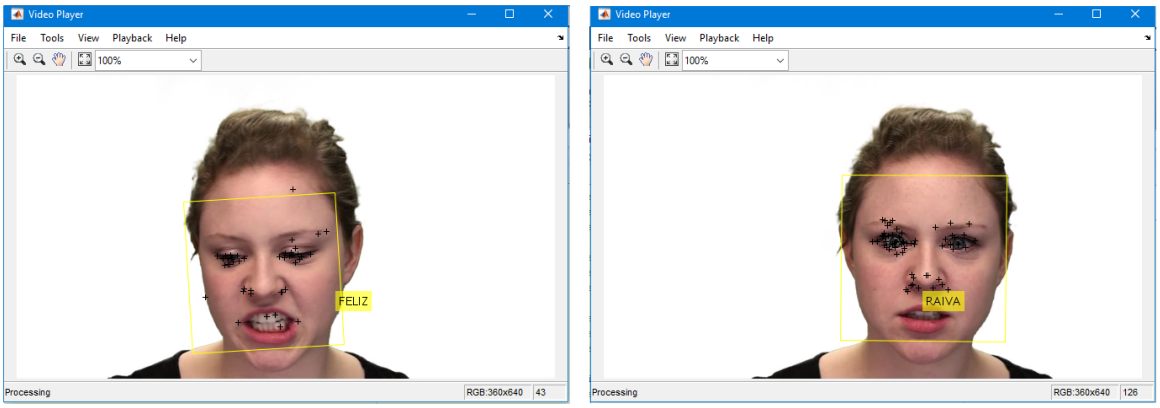

Figura 39 – Trechos de vídeos onde o novo ator é classificado de forma incorreta com a expressão facial feliz e raiva no teste de DN.

No teste de classificação dos estados afetivos da face em fluxo de vídeo as etiquetas são aplicadas a cada *frame* e a taxa de sucesso na identificação do estado afetivo da face do ator 1 está condicionada aos resultados apresentados nas matrizes de confusão mostradas nas Figuras 19 e 24. Logo, estes resultados indicam que as redes MLP e RBF são uma solução adequada para classificar os estados afetivos da face em fluxo de vídeo mesmo com a mudança gradual da face durante a execução do vídeo.

No teste de DN com expressões faciais similares as redes MLP e RBF apresentaram erros para identificar novos *frames* em todo conjunto de vídeos, de modo que as classificações indevidas ocorrem com maior frequência para os estados afetivos triste e raiva. No teste de DN com novos atores as redes MLP e RBF classificam equivocadamente com maior frequência os estados afetivos feliz e raiva.

O algoritmo com uso da rede MLP produz resultados com uma taxa de acerto de 98,5% para detecção de novos atores. No entanto, ao avaliar os vídeos do ator 1 com expressões faciais semelhantes a taxa de acerto da rede MLP reduz para 79,8%. No algoritmo com a rede RBF obtêm-se uma taxa de acerto de 98,9% para a detecção de novos atores, mas ao avaliar os vídeos do ator 1 com expressões faciais similares esta taxa de acerto reduz para 85,2%.

Os resultados alcançados para os testes de DN qualificam as redes MLP e RBF para tarefa de DN com a abordagem multi-classes. Logo, a performance apresentada pelas redes MLP e RBF para DN com faces integralmente novas demonstram que um conjunto de dados que possui pouco semelhança com os dados de treinamento produz erros significativamente maiores elevando consecutivamente a taxa de acerto para DN, no entanto dados que são similares aos dados utilizados na fase de treinamento produzem níveis mais baixos de erros tornando a tarefa de DN mais complexa (Chen-Chia Chuang; Shun-Feng Su; Chin-Ching Hsiao, 2000). Para apresentar os resultados obtidos para DN com as redes MLP e RBF disponibiliza-se os links: <https://youtu.be/9M29DsCcQhQ> e <https://youtu.be/m2APObOwvMQ>

- **Taxa de *Frames Por Second* (FPS) para as redes MLP e RBF**

A taxa de FPS é um requisito importante para aplicações que trabalham com processamento de vídeos e requerem uma resposta em tempo real. O processamento em tempo real trabalha com fluxos de dados que são capturados em tempo real e processados com requisitos de latência mínima da ordem de milissegundos ou segundos visando produzir respostas automatizadas. As Tabelas 6, 7 e 8 apresentam os resultados obtidos para taxa de FPS com a avaliação dos vídeos da etapa de classificação e DN com uso das redes MLP e RBF. As características técnicas de *hardware* e *software* usadas no teste são descritas no Capítulo 3 na Seção 3.2.

Tabela 6 – Taxa de FPS para etapa de classificação.

Vídeo Treino	Estado Afetivo	Total *Frames*	FPS (MLP)	FPS (RBF)
Vídeo 1	Feliz	103	12,6	8,7
Vídeo 2	Triste	132	12,1	8,4
Vídeo 3	Raiva	142	12,4	8,3
Vídeo 4	Neutro	97	12,6	8,9
Vídeo 5	Feliz	103	12,5	8,5
Vídeo 6	Triste	106	12,8	8,6
Vídeo 7	Raiva	104	12,7	8,7
Vídeo 8	Neutro	99	12,8	8,9
Vídeo 9	Feliz	105	12,9	8,5
Vídeo 10	Triste	105	12,6	8,5
Vídeo 11	Raiva	117	12,5	8,4
Vídeo 12	Neutro	98	12,9	8,8

Tabela 7 – Taxa de FPS para DN com expressões faciais do ator 1.

Vídeo Novidade	Estado Afetivo	Total *Frames*	FPS (MLP)	FPS (RBF)
Vídeo 1	Calmo	105	12,7	8,2
Vídeo 2	Calmo	107	12,3	8,5
Vídeo 3	Calmo	104	12,1	8,6
Vídeo 4	Medo	109	12,6	8,3
Vídeo 5	Medo	108	12,7	8,4
Vídeo 6	Medo	102	12,6	8,4
Vídeo 7	Surpreso	97	12,2	8,1
Vídeo 8	Surpreso	101	12,3	8,3
Vídeo 9	Surpreso	97	12,1	8,4
Vídeo 10	Nojo	115	12,4	8,2
Vídeo 11	Nojo	116	12,4	8,3
Vídeo 12	Nojo	116	12,3	8,2

Tabela 8 – Taxa de FPS para DN com expressões faciais de novos atores.

Vídeo Novidade	Estado Afetivo	Total *Frames*	FPS (MLP)	FPS (RBF)
Vídeo 13	2 Feliz	134	12,2	8,1
Vídeo 14	2 Feliz	138	12,2	8,1
Vídeo 15	2 Feliz	138	12,2	8,1
Vídeo 16	5 Triste	129	12,5	8,3
Vídeo 17	5 Triste	123	12,3	8,3
Vídeo 18	5 Triste	148	12,1	8,2
Vídeo 19	6 Calmo	139	12,3	8,1
Vídeo 20	6 Calmo	139	12,2	8,1
Vídeo 21	6 Calmo	138	12,3	8,3
Vídeo 22	10 Raiva	144	12,1	8,1
Vídeo 23	10 Raiva	136	12,3	8,2
Vídeo 24	10 Raiva	136	12,4	8,2

Os resultados obtidos para taxa de FPS na avaliação da base de dados de treino e DN são compatíveis para aplicações de tempo real, de forma que os estados afetivos da face são classificados ou detectados como novos a cada *frame* com fluidez contínua ao longo do vídeo. O algoritmo proposto através da rede MLP apresenta uma taxa de FPS superior em todo conjunto de vídeos, estes resultados confirmam que a rede MLP processa dados em tempo real com maior rapidez comparado a rede RBF. Esta performance superior apresentada através do algoritmo com uso da rede MLP justifica-se devido ao algoritmo de treinamento SGC que acelera a convergência do aprendizado da rede MLP resultando em uma rede com apenas 12 neurônios na camada oculta que processa dados em fluxo de vídeo com menor custo computacional.

- **Custo Computacional**

No processo de análise do custo computacional critérios relacionados ao tempo de processamento e quantidade de memória requerida pelo algoritmo em uma simulação tornam-se aspectos relevantes para o desenvolvimento de sistemas embarcados. De modo que, projetos de sistemas embarcados impõem exigências rígidas, demandando tempos de processamento reduzidos, pequena exigência de memória e baixo consumo de energia (ALJAHDALI; SHETA; DEBNATH, 2015; LUO et al., 2020). Os Gráficos 40, 41 e 42 apresentam os resultados alcançados para o tempo processamento e as Tabelas 9, 10 e 11 para quantidade de memória requerida. Recursos de *hardware* e *software* são descritos no Capítulo 3 na Seção 3.2.

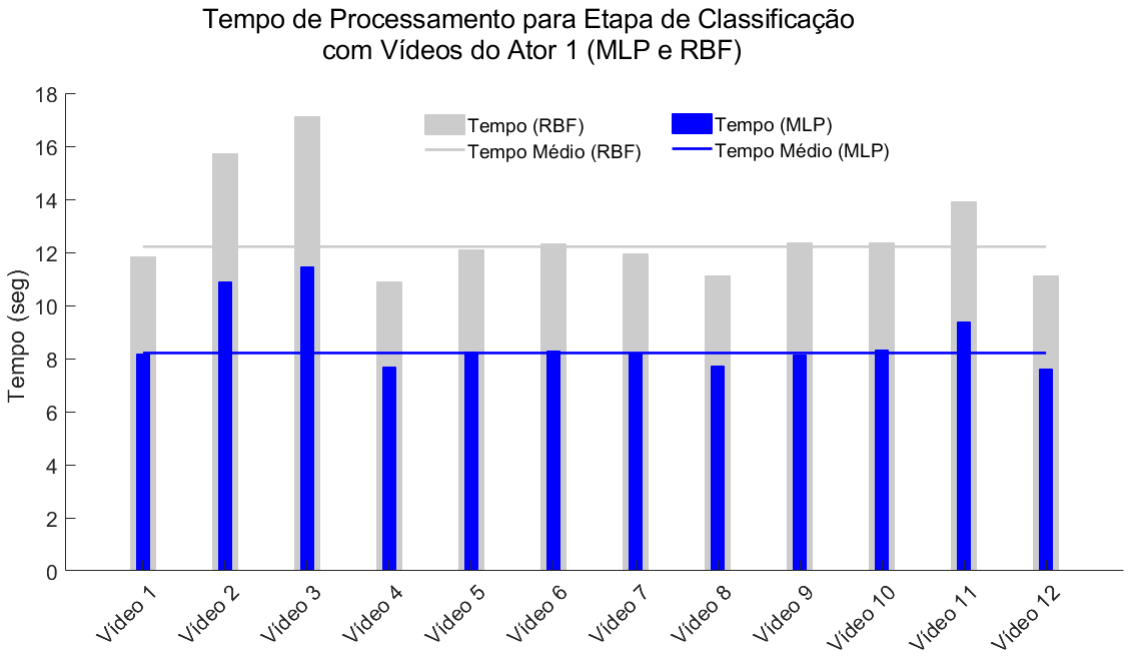

Figura 40 – Tempo de processamento para vídeos da etapa de classificação (Ator 1) com uso dos algoritmos MLP e RBF. Fonte: Autor.

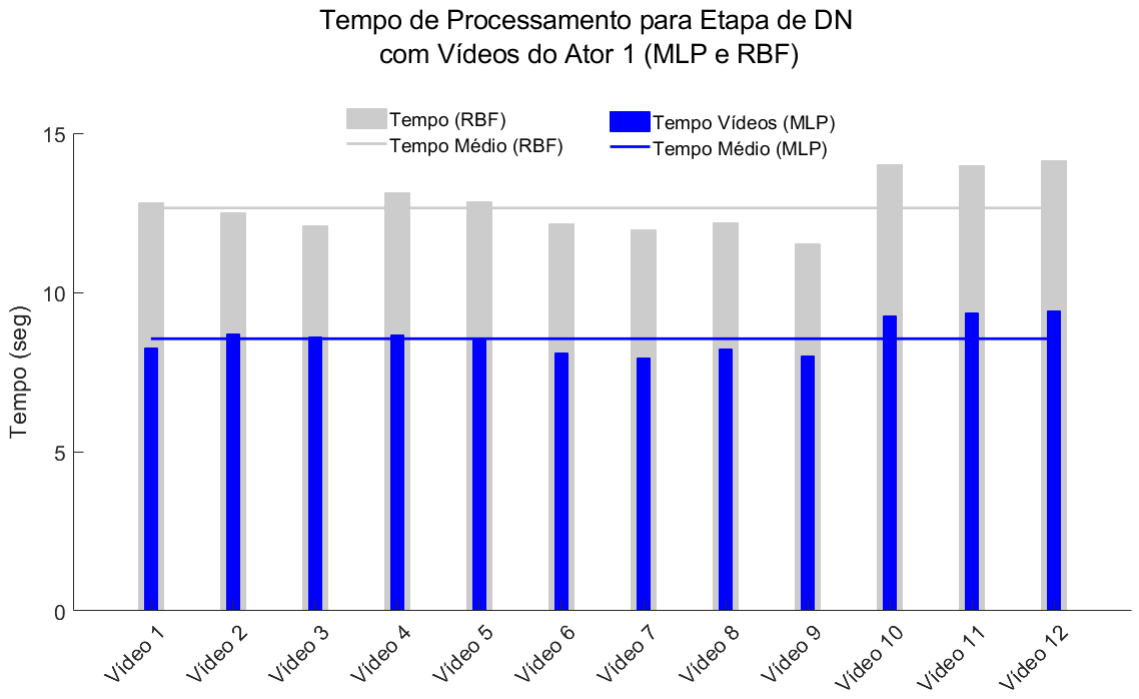

Figura 41 – Tempo de processamento para vídeos da etapa de DN (Ator 1) com uso dos algoritmos MLP e RBF. Fonte: Autor.

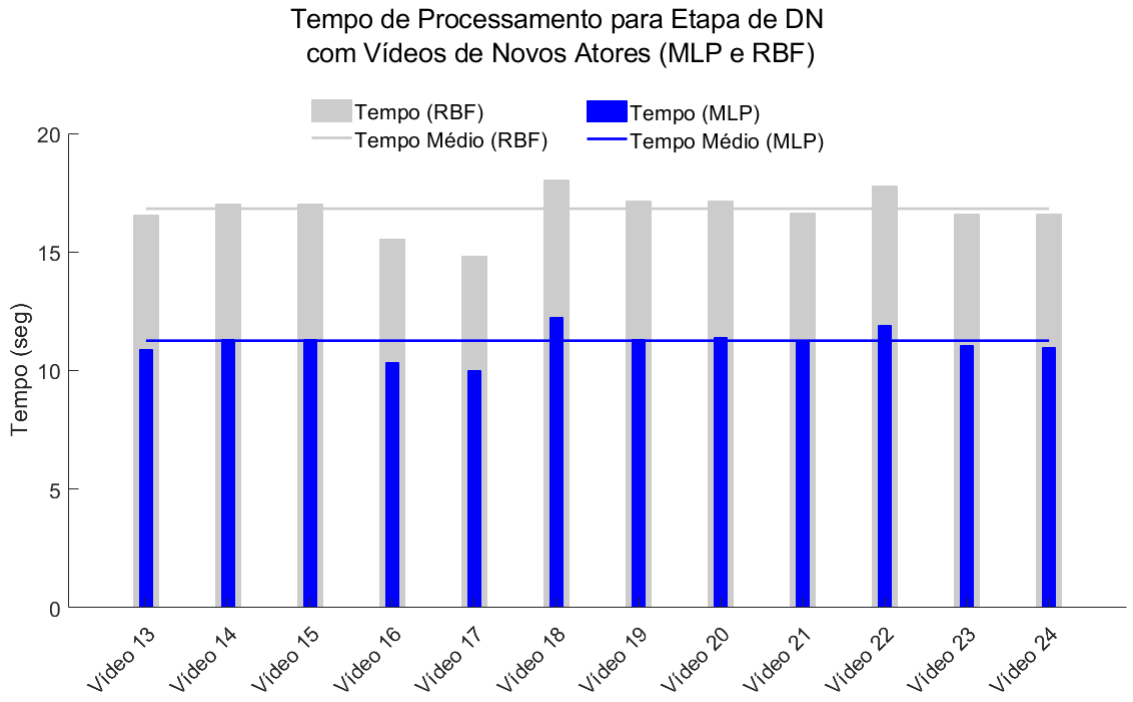

Figura 42 – Tempo de processamento para vídeos da etapa de DN (Novos Atores) com uso dos algoritmos MLP e RBF. Fonte: Autor.

Tabela 9 – Memória requerida para cada vídeo do ator 1 da etapa de classificação.

Vídeo Treino	Estado Afetivo	Total *Frames*	Memória (MLP)	Memória (RBF)
Vídeo 1	Feliz	103	886 MB	901 MB
Vídeo 2	Triste	132	890 MB	907 MB
Vídeo 3	Raiva	142	899 MB	902 MB
Vídeo 4	Neutro	97	876 MB	902 MB
Vídeo 5	Feliz	103	889 MB	902 MB
Vídeo 6	Triste	106	886 MB	904 MB
Vídeo 7	Raiva	104	882 MB	904 MB
Vídeo 8	Neutro	99	883 MB	904 MB
Vídeo 9	Feliz	105	879 MB	907 MB
Vídeo 10	Triste	105	878 MB	906 MB
Vídeo 11	Raiva	117	889 MB	911 MB
Vídeo 12	Neutro	98	871 MB	910 MB

Tabela 10 – Memória requerida para cada vídeo do ator 1 da etapa de DN.

Vídeo Novidade	Estado Afetivo	Total *Frames*	Memória (MLP)	Memória (RBF)
Vídeo 1	Calmo	105	896 MB	910 MB
Vídeo 2	Calmo	107	868 MB	911 MB
Vídeo 3	Calmo	104	871 MB	903 MB
Vídeo 4	Medo	109	881 MB	907 MB
Vídeo 5	Medo	108	879 MB	908 MB
Vídeo 6	Medo	102	877 MB	908 MB
Vídeo 7	Surpreso	97	878 MB	905 MB
Vídeo 8	Surpreso	101	885 MB	909 MB
Vídeo 9	Surpreso	97	884 MB	907 MB
Vídeo 10	Nojo	115	886 MB	911 MB
Vídeo 11	Nojo	116	887 MB	914 MB
Vídeo 12	Nojo	116	887 MB	914 MB

Tabela 11 – Memória requerida para cada vídeo com novos atores da etapa de DN.

Vídeo Novidade	Estado Afetivo	Total *Frames*	Memória (MLP)	Memória (RBF)
Vídeo 13	2 Feliz	134	887 MB	904 MB
Vídeo 14	2 Feliz	138	886 MB	905 MB
Vídeo 15	2 Feliz	138	888 MB	904 MB
Vídeo 16	5 Triste	129	891 MB	905 MB
Vídeo 17	5 Triste	123	886 MB	905 MB
Vídeo 18	5 Triste	148	890 MB	901 MB
Vídeo 19	6 Calmo	139	890 MB	906 MB
Vídeo 20	6 Calmo	139	891 MB	905 MB
Vídeo 21	6 Calmo	138	888 MB	904 MB
Vídeo 22	10 Raiva	144	893 MB	906 MB
Vídeo 23	10 Raiva	136	891 MB	907 MB
Vídeo 24	10 Raiva	136	893 MB	907 MB

O desempenho obtido com os algoritmos MLP e RBF para o tempo de processamento e memória requerida em uma simulação são adequados para o desenvolvimento de sistemas embarcados com a utilização de mini computadores. Constata-se também que o algoritmo MLP apresenta o menor tempo de computação para as etapas de classificação e DN. De maneira que, o menor e o maior tempo de computação em segundos apresentados pelos algoritmos MLP e RBF considerando todos os vídeos das etapas de classificação e DN são de 7,59 e 12,23 versus 10,9 e 18,1. Em relação ao uso da memória o algoritmo MLP produz resultados com maior eficiência para toda etapa de classificação e DN.

- **Comparação dos Resultados para DN**

A Tabela 12 apresenta resultados para processamento de imagens e vídeos para fins comparativos com aplicações de DN. Os resultados da Tabela 12 são descritos com maiores detalhes no Capítulo 2 na Seção 2.1.

Tabela 12 – Resultados para DN em recursos de imagens e vídeos.

Autores, Ano	Aplicações	Rede Neural	Desempenho
Markou e Singh (2006)	DN e classificação de objetos em cenas naturais de quatro pares de vídeos.	MLP	Z de 89%, 92%, 94% e 67%.
Wildermann e Teich (2008)	DN para processamento de imagens de faces em ambiente dinâmico para adaptação *online*.	RBF	ACC de 100%.
Kim e Cho (2019)	Detecção de padrões anômalos em imagens com circulação de pedestres, objetos e dígitos manuscritos.	Autoencoder	AUC de 89%, 87,5% e 87,5%.
Nantes, Brown e Maire (2013)	Detecção de anomalias que afetam a cor e geometria de imagens em ambiente virtual 3D.	MLP e SOM	AUC de 65% e 95%.

Os resultados descritos na Tabela 12 para DN com imagens e vídeos apresentam níveis de desempenho em uma faixa de 67% até 100%. Neste trabalho as taxas de acerto para DN com uso das redes neurais MLP e RBF são de 79,8% e 85,2% para DN com expressões faciais semelhantes as usadas na fase de treinamento e 98,5% e 98,9% para DN com expressões faciais de novos atores. Essas respostas de performance são compatíveis com os resultados produzidos pelos pesquisadores descrito na Tabela 12 confirmando que as redes MLP e RBF podem ser adaptadas e ajustadas para serem utilizadas como uma alternativa eficaz para realizar a DN em fluxo de vídeo.

5 Conclusões

5.1 Conclusões

A DN aplicada no reconhecimento de expressões faciais em fluxo de vídeo foi investigada nesta pesquisa com base nos métodos VJ, KLT, PCA e ANN visando o desenvolvimento de um algoritmo compacto com baixo custo computacional. As redes MLP e RBF são habilitadas para DN mediante as técnicas empregadas nas etapas de treinamento e teste. A validação para DN é realizada com vídeos de expressões faciais semelhantes as usadas na fase de treinamento e vídeos com expressões faciais de novos atores.

A performance das redes MLP e RBF para DN em fluxo de vídeo tem como base a taxa de acertos para detecção de novos *frames*. Os resultados obtidos com a rede RBF confirmam o desempenho de 85,2% para detecção de novos *frames* com atributos próximos aos *frames* usados na fase de treinamento e uma taxa de acerto de 98,9% para *frames* com atributos de novos atores. Os resultados da rede MLP são validados com a taxa de acerto de 98,5% para detecção de *frames* totalmente novos. No entanto, quando quadros de vídeos semelhantes são apresentados para a rede MLP sua performance reduz a uma taxa de sucesso de 79,8% para DN.

Deste modo, conclui-se que as técnicas empregadas para habilitar as redes MLP e RBF para tarefa de DN são ambas consistentes para identificar vetores de atributos totalmente novos. Entretanto, quando avalia-se vetores de atributos semelhantes a taxa de sucesso das redes MLP e RBF são limitadas a um percentual de acerto, onde a rede RBF mostra-se mais adequada para tarefa de DN.

Na fase de extração de atributos a integração dos algoritmos VJ, KLT e PCA viabilizam a produção de vetores de atributos com uma alta fidelidade das características provenientes das expressões faciais captadas ao longo do vídeo. A robustez dos algoritmos VJ e KLT em detectar e rastrear a face com baixo custo computacional ao longo de todo vídeo são aspectos relevantes para lidar com as variações produzidas para as mudanças da face durante todo vídeo. Logo, a técnica PCA permite reduzir a dimensionalidade da face segmentada pelo método holístico eliminando redundâncias e características indesejadas da face sem perda significativa do dados originais.

Na etapa de treinamento as redes MLP e RBF apresentam uma acurácia de classificação de 99,8% e 99,9%. As confusões ocorrem entre os estados afetivos feliz e triste devido a semelhança entre as algumas expressões faciais. Os resultados obtidos para investigação com ACC (%) vs MSE em função da quantidade de neurônios da camada

oculta produz métricas fundamentais para seleção das redes com maior potencial para criação dos limiares para DN em fluxo de vídeo.

Os resultados provenientes da taxa de FPS, tempo de computação e quantidade de memória requerida confirmam que o algoritmo proposto com base na integração dos métodos VJ, KLT, PCA e ANN produz resultados compatíveis para o desenvolvimento de sistemas embarcados inteligentes com resposta de tempo real. De modo, que a integração dos métodos VJ, KLT, PCA e MLP supera a sinergia dos algoritmos VJ, KLT, PCA e RBF em todo conjunto de dados de classificação e DN.

Contudo, conclui-se que a DN aplicada no reconhecimento de expressões faciais em fluxo de vídeo pode ser alcançada com uma taxa de sucesso satisfatória com a integração dos algoritmos VJ, KLT, PCA e ANN para uma abordagem multi-classes. Consequentemente, que esta pesquisa fornece uma concepção inovadora para o desenvolvimento de novas tecnologias com o reconhecimento de expressões faciais possibilitando capacitar um sistema computacional para identificar novos estados afetivos da face que são diferentes dos aprendidos na fase de treinamento da máquina.

5.2 Trabalhos Futuros

Os avanços recentes com uso de redes CNN estabelecem novos padrões de precisão em tarefas de visão computacional, de modo que a DN com uso de redes CNN tem-se mostrado uma alternativa confiável para aplicações com processamento de imagens (Wendl; Marcos; Tuia, 2019; Amorim et al., 2019; Jintawatsakoon; Charoenruengkit, 2020; Ghaffari; Raie, 2017). Para trabalhos futuros, propõe-se um estudo para DN no reconhecimento de expressões faciais em fluxo de vídeo com uso de redes CNN, afim de comparar a taxa de sucesso para DN e taxa de FPS.

Em um segundo momento, em embarcar o algoritmo proposto em um mini computador de baixo custo com capacidade para processamento de vídeo em tempo real, visando a implementação do módulo de reconhecimento de expressões faciais do projeto robô assistivo Hibot.

Referências

Abbas, E. I.; Safi, M. E.; Rijab, K. S. Face recognition rate using different classifier methods based on pca. In: *2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT)*. [S.l.: s.n.], 2017. p. 37–38. Citado 2 vezes nas páginas 11 e 20.

Abdullah, M.; Ahmad, M.; Han, D. Facial expression recognition in videos: An cnn-lstm based model for video classification. In: *2020 International Conference on Electronics, Information, and Communication (ICEIC)*. [S.l.: s.n.], 2020. p. 1–3. Citado 3 vezes nas páginas 2, 9 e 32.

Ackovska, N. et al. Robot - assisted therapy for autistic children. In: *SoutheastCon 2017*. [S.l.: s.n.], 2017. p. 1–2. Citado na página 10.

Ahmed, W. et al. Assisting the autistic with improved facial expression recognition from mixed expressions. In: *2013 Fourth National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics (NCVPRIPG)*. [S.l.: s.n.], 2013. p. 1–4. Citado na página 3.

ALBRECHT, S. et al. Generalized radial basis function networks for classification and novelty detection: self-organization of optimal bayesian decision. *Neural Networks*, v. 13, p. 1075–1076, 2000. Citado na página 26.

ALJAHDALI, S.; SHETA, A. F.; DEBNATH, N. C. Estimating software effort and function point using regression, support vector machine and artificial neural networks models. In: *2015 IEEE/ACS 12th International Conference of Computer Systems and Applications (AICCSA)*. [S.l.: s.n.], 2015. p. 1. Citado na página 60.

Amorim, M. et al. Novelty detection in social media by fusing text and image into a single structure. *IEEE Access*, v. 7, p. 132788, 2019. Citado 2 vezes nas páginas 1 e 66.

Ariza-Lopez, F. J.; Rodriguez-Avi, J.; Alba-Fernandez, M. V. Complete control of an observed confusion matrix. In: *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*. [S.l.: s.n.], 2018. p. 1222. Citado na página 31.

Askari, F. et al. A pilot study on facial expression recognition ability of autistic children using ryan, a rear-projected humanoid robot. In: *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. [S.l.: s.n.], 2018. p. 790–795. Citado na página 10.

AUGUSTEIJN, M. F.; FOLKERT, B. A. Neural network classification and novelty detection. *International Journal of Remote Sensing*, v. 23, p. 2891–2893, 2002. Citado 2 vezes nas páginas 24 e 28.

BARNOUTI, N. H.; AL-MAYYAHI, M. H. N.; AL-DABBAGH, S. S. Real-time face tracking and recognition system using kanade-lucas-tomasi and two-dimensional principal component analysis. *2018 International Conference on Advanced Science and Engineering (ICOASE), Duhok*, p. 24–28, 2018. Citado 3 vezes nas páginas 1, 2 e 19.

BARRETO, A.; FROTA, A. A unifying methodology for the evaluation of neural network models on novelty detection tasks. *Pattern Analysis and Applications*, v. 16, p. 85–89, 2012. Citado 2 vezes nas páginas 24 e 25.

CALVO, R.; D'MELLO, S. Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing*, v. 1, p. 1, 2010. Citado na página 1.

Camada, M. Y. O.; Cerqueira, J. J. F.; Lima, A. M. N. Stereotyped gesture recognition: An analysis between hmm and svm. In: *2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*. [S.l.: s.n.], 2017. p. 328–330. Citado na página 3.

Candra Kirana, K.; Wibawanto, S.; Wahyu Herwanto, H. Facial emotion recognition based on viola-jones algorithm in the learning environment. In: *2018 International Seminar on Application for Technology of Information and Communication*. [S.l.: s.n.], 2018. p. 406. Citado na página 15.

CHAI, Z.; SHI, J. Improving KLT in embedded systems by processing oversampling video sequence in real-time. *2011 International Conference on Reconfigurable Computing and FPGAs*, p. 297–298, 2011. Citado 3 vezes nas páginas 1, 2 e 19.

Chatrath, J. et al. Real time human face detection and tracking. In: *2014 International Conference on Signal Processing and Integrated Networks (SPIN)*. [S.l.: s.n.], 2014. p. 705–706. Citado na página 18.

CHATTERJEE, D.; CHANDRAN, S. Comparative study of camshift and KLT algorithms for real time face detection and tracking applications. *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata*, p. 63–64, 2016. Citado 3 vezes nas páginas 1, 9 e 19.

Chen-Chia Chuang; Shun-Feng Su; Chin-Ching Hsiao. The annealing robust backpropagation (arbp) learning algorithm. *IEEE Transactions on Neural Networks*, v. 11, n. 5, p. 1067–1077, 2000. Citado na página 58.

Chen, Z.; Chen, X. Study on interactive robots with contingent responses. In: *2018 2nd IEEE Advanced Information Management,Communicates,Electronic and Automation Control Conference (IMCEC)*. [S.l.: s.n.], 2018. p. 1001–1005. Citado na página 10.

Cherabit, N.; Djeradi, A.; Chelali, F. z. Facial motion analysis using template matching. In: *020 1st International Conference on Communications, Control Systems and Signal Processing (CCSSP)*. [S.l.: s.n.], 2020. p. 151–156. Citado 2 vezes nas páginas 9 e 19.

Dang, K.; Sharma, S. Review and comparison of face detection algorithms. In: *2017 7th International Conference on Cloud Computing, Data Science Engineering - Confluence*. [S.l.: s.n.], 2017. p. 629–633. Citado 4 vezes nas páginas 8, 15, 16 e 18.

Das, S. K.; Akter, L. Analysis of statistical features for face recognition based on holistic approach. In: *2017 International Conference on Electrical, Computer and Communication Engineering (ECCE)*. [S.l.: s.n.], 2017. p. 75–78. Citado na página 13.

De la Torre, F. et al. Intraface. In: *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. [S.l.: s.n.], 2015. v. 1, p. 4. Citado 2 vezes nas páginas 13 e 12.

DELAC, K.; GRGIC, M.; GRGIC, S. Independent comparative study of pca, ica, and lda on the feret data set. *International Journal of Imaging Systems and Technology*, v. 15, p. 252–254, 2005. Citado 2 vezes nas páginas 11 e 20.

DOMINGUES, R. et al. A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition*, v. 74, p. 407–408, 2018. Citado na página 1.

FARIA, E. R. et al. Novelty detection in data streams. *Artificial Intelligence Review*, v. 45, n. 2, p. 235–269, 2016. Citado na página 1.

FORESEE, F. D.; HAGAN, M. T. Gauss-newton approximation to bayesian learning. *Proceedings of International Conference on Neural Networks (ICNN'97)*, v. 3, p. 1930–1935, 2017. Citado na página 26.

Ghaffari, S.; Raie, A. A. Pedestrian detection using improved proposal boxes and grayscale convolutional learning. In: *2017 10th Iranian Conference on Machine Vision and Image Processing (MVIP)*. [S.l.: s.n.], 2017. p. 70–75. Citado na página 66.

Ghaleb, E.; Popa, M.; Asteriadis, S. Multimodal and temporal perception of audio-visual cues for emotion recognition. In: *2019 8th International Conference on Affective Computing and Intelligent Interaction (ACII)*. [S.l.: s.n.], 2019. p. 552–558. Citado na página 9.

HAYKIN, S. *Neural Networks and Learning Machines*. [S.l.]: Pearson Education, Inc, 2009. (3rd ed). ISBN 10:0-13-147139-2. Citado 3 vezes nas páginas 23, 25 e 26.

HODGE, V. J.; AUSTIN, J. A survey of outlier detection methodologies. *Artificial Intelligence Review*, v. 22, p. 85–122, 2004. Citado 3 vezes nas páginas 8, 24 e 25.

Ismail, L. et al. Face detection technique of humanoid robot nao for application in robotic assistive therapy. In: *2011 IEEE International Conference on Control System, Computing and Engineering*. [S.l.: s.n.], 2011. p. 517–521. Citado 3 vezes nas páginas 3, 10 e 14.

JAMEEL, R.; SINGHAL, A.; BANSAL, A. A comprehensive study on facial expressions recognition techniques. In: *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*. [S.l.: s.n.], 2016. p. 478–479. Citado 2 vezes nas páginas 12 e 14.

Jintawatsakoon, S.; Charoenruengkit, W. Novelty detection of beverage bottle images based on transfer learning. In: *2020 - 5th International Conference on Information Technology (InCIT)*. [S.l.: s.n.], 2020. p. 87. Citado na página 66.

KAYHAN, G.; OZDEMIR, A. E.; EMINOGLU, I. Reviewing and designing pre-processing units for rbf networks: initial structure identification and coarse-tuning of free parameters. *Springer*, v. 1, n. 1, p. 1655–1657, 2012. Citado na página 26.

Kim, J.; Cho, S. Unsupervised novelty detection in video with adversarial autoencoder based on non-euclidean space. In: *2019 15th International Conference on Signal-Image Technology Internet-Based Systems (SITIS)*. [S.l.: s.n.], 2019. p. 22–28. Citado 2 vezes nas páginas 8 e 63.

KRIESEL, D. *A Brief Introduction to Neural Networks*. [s.n.], 2007. Disponível em: <availableathttp://www.dkriesel.com>. Citado na página 26.

Lee, J.; Obinata, G.; Aoki, H. A pilot study of using touch sensing and robotic feedback for children with autism. In: *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. [S.l.: s.n.], 2014. p. 222. Citado na página 11.

Li, C. et al. Matrix reduction based on generalized pca method in face recognition. In: *2014 5th International Conference on Digital Home*. [S.l.: s.n.], 2014. p. 35–37. Citado na página 21.

LIU, Y.; KAU, L. Scalable face image compression based on principal component analysis and arithmetic coding. *2017 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW)*, p. 256, 2017. Citado 3 vezes nas páginas 1, 2 e 20.

LIVINGSTONE, R. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. , May 2018. Disponível em: <https://doi.org/10.1371/journal.pone.0196391>. Citado 8 vezes nas páginas 2, 32, 33, 34, 35, 36, 45 e 49.

LUO, H. et al. Embedded object detection system based on deep neural network. In: *2020 13th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*. [S.l.: s.n.], 2020. p. 383–386. Citado na página 60.

MARKOU, M.; SINGH, S. Novelty detection: a review—part 1: statistical approaches. *Signal Processing*, v. 83, p. 2481, 2003. Citado 2 vezes nas páginas 22 e 23.

MARKOU, M.; SINGH, S. Novelty detection: a review—part 2: neural network based approaches. *Signal Processing*, v. 83, p. 2499–2503, 2003. Citado 7 vezes nas páginas 1, 2, 8, 22, 23, 24 e 25.

Markou, M.; Singh, S. A neural network-based novelty detector for image sequence analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 28, n. 10, p. 1664–1675, 2006. Citado 4 vezes nas páginas 2, 3, 7 e 63.

Matari, M. Socially assistive robotics: Human-robot interaction methods for creating robots that care. In: *2014 9th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. [S.l.: s.n.], 2014. p. 333. Citado na página 10.

Meher, S. S.; Maben, P. Face recognition and facial expression identification using pca. In: *2014 IEEE International Advance Computing Conference (IACC)*. [S.l.: s.n.], 2014. p. 1093. Citado na página 14.

Mishra, S.; Prusty, R.; Hota, P. K. Analysis of levenberg-marquardt and scaled conjugate gradient training algorithms for artificial neural network based ls and mmse estimated channel equalizers. In: *2015 International Conference on Man and Machine Interfacing (MAMI)*. [S.l.: s.n.], 2015. p. 1–7. Citado na página 49.

MSTAFA, R. J.; ELLEITHY, K. M. A video steganography algorithm based on kanade-lucas-tomasi tracking algorithm and error correcting codes. *Multimedia Tools and Applications*, v. 75, n. 17, p. 10311–10319, 2016. Citado 2 vezes nas páginas 19 e 20.

NANTES, A.; BROWN, R.; MAIRE, F. Neural network-based detection of virtual environment anomalies. *Neural Computing and Applications*, v. 23, n. 6, p. 1711–1723, 2013. Citado 2 vezes nas páginas 8 e 63.

Nehru, M.; Padmavathi, S. Illumination invariant face detection using viola jones algorithm. In: *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*. [S.l.: s.n.], 2017. p. 2. Citado na página 18.

Nikolopoulos, C. et al. Robotic agents used to help teach social skills to children with autism: The third generation. In: *2011 RO-MAN*. [S.l.: s.n.], 2011. p. 253–254. Citado na página 10.

Oliveira, Moisés A. et al. Ultrasound-based identification of damage in wind turbine blades using novelty detection. *Ultrasonics*, p. 1–2, 2020. Citado na página 1.

Ouafae, B. et al. Novelty detection review state of art and discussion of new innovations in the main application domains. In: *2020 1st International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*. [S.l.: s.n.], 2020. p. 1–2. Citado na página 1.

Pantic, M.; Patras, I. Dynamics of facial expression: recognition of facial actions and their temporal segments from face profile image sequences. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, v. 36, n. 2, p. 433, 2006. Citado 2 vezes nas páginas 1 e 12.

Peleshko, D.; Soroka, K. Research of usage of haar-like features and adaboost algorithm in viola-jones method of object detection. In: *2013 12th International Conference on the Experience of Designing and Application of CAD Systems in Microelectronics (CADSM)*. [S.l.: s.n.], 2013. p. 284–286. Citado na página 18.

PIMENTEL, A. et al. A review of novelty detection. *Signal Processing*, v. 99, p. 217–232, 2014. Citado 6 vezes nas páginas 1, 22, 23, 24, 25 e 26.

PONT, L. Y.; JONES, N. B. Improving the performance of radial basis function classifiers in condition monitoring and fault diagnosis applications where 'unknown' faults may occur. *Pattern Recognition Letters*, v. 23, p. 569–572, 2002. Citado na página 26.

PUNYANI, P.; GUPTA, R.; KUMAR, A. Neural networks for facial age estimation: a survey on recent advances. *Artificial Intelligence Review*, v. 53, n. 5, p. 3299–3300, 2020. Citado na página 1.

Putro, M. D.; Jo, K. Real-time face tracking for human-robot interaction. In: *2018 International Conference on Information and Communication Technology Robotics (ICT-ROBOT)*. [S.l.: s.n.], 2018. p. 1–4. Citado na página 19.

REVINA, I. M.; EMMANUEL, W. R. S. A survey on human face expression recognition techniques. *Journal of King Saud University - Computer and Information Sciences*, v. 1, n. 1, p. 1–5, 2018. Citado na página 14.

Sai, Y.; Jinxia, R.; Zhongxia, L. Learning of neural networks based on weighted mean squares error function. In: *2009 Second International Symposium on Computational Intelligence and Design*. [S.l.: s.n.], 2009. v. 1, p. 241. Citado na página 31.

SAMEER, S.; MARKOU, M. An approach to novelty detection applied to the classification of image regions. *IEEE Transactions on Knowledge and Data Engineering*, v. 16, p. 396–397, 2004. Citado 3 vezes nas páginas 1, 24 e 28.

Sharifara, A.; Mohd Rahim, M. S.; Anisi, Y. A general review of human face detection including a study of neural networks and haar feature-based cascade classifier in face detection. In: *2014 International Symposium on Biometrics and Security Technologies (ISBAST)*. [S.l.: s.n.], 2014. p. 73–78. Citado na página 15.

SHINDE, S. B.; SAYYAD, S. S. Cost sensitive improved levenberg marquardt algorithm for imbalanced data. In: *2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*. [S.l.: s.n.], 2016. p. 1. Citado na página 49.

Sivasankari, K.; Thanushkodi, K.; Kalaivanan, K. Automated seizure detection using multilayer feed forward network trained using scaled conjugate gradient method. In: *2013 International Conference on Current Trends in Engineering and Technology (ICCTET)*. [S.l.: s.n.], 2013. p. 195–198. Citado na página 25.

Suchitra; Suja P.; Tripathi, S. Real-time emotion recognition from facial images using raspberry pi ii. In: *2016 3rd International Conference on Signal Processing and Integrated Networks (SPIN)*. [S.l.: s.n.], 2016. p. 666. Citado 2 vezes nas páginas 3 e 11.

Tavallali, P.; Yazdi, M.; Khosravi, M. R. An efficient training procedure for viola-jones face detector. In: *2017 International Conference on Computational Science and Computational Intelligence (CSCI)*. [S.l.: s.n.], 2017. p. 828–831. Citado na página 18.

Tong, Y.; Liao, W.; Ji, Q. Facial action unit recognition by exploiting their dynamic and semantic relationships. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 29, n. 10, p. 1683–1684, 2007. Citado na página 13.

VASCONCELOS, G. C.; FAIRHURST, D. B. Investigating feedforward neural networks with respect to the rejection of spurious patterns. *Pattern Recognition Letters*, v. 16, p. 208–209, 1995. Citado na página 26.

VIOLA, P.; JONES, M. Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, v. 1, p. 511, 2001. Citado na página 1.

Wendl, C.; Marcos, D.; Tuia, D. Novelty detection in very high resolution urban scenes with density forests. In: *2019 Joint Urban Remote Sensing Event (JURSE)*. [S.l.: s.n.], 2019. p. 1. Citado na página 66.

Wildermann, S.; Teich, J. A sequential learning resource allocation network for image processing applications. In: *2008 Eighth International Conference on Hybrid Intelligent Systems*. [S.l.: s.n.], 2008. p. 132–137. Citado 2 vezes nas páginas 7 e 63.

Wu, S.; Nagahashi, H. Parameterized adaboost: Introducing a parameter to speed up the training of real adaboost. *IEEE Signal Processing Letters*, v. 21, n. 6, p. 687, 2014. Citado na página 18.

Xue, Y.; Mao, X.; Zhang, F. Beihang university facial expression database and multiple facial expression recognition. In: *2006 International Conference on Machine Learning and Cybernetics*. [S.l.: s.n.], 2006. p. 3282–3287. Citado na página 18.

Yong Liu. Create stable neural networks by cross-validation. In: *The 2006 IEEE International Joint Conference on Neural Network Proceedings*. [S.l.: s.n.], 2006. p. 3925–3926. Citado na página 40.

Yongmian Zhang; Qiang Ji. Active and dynamic information fusion for facial expression understanding from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 27, n. 5, p. 699, 2005. Citado na página 14.

Yuhua Li. A surface representation approach for novelty detection. In: *2008 International Conference on Information and Automation.* [S.l.: s.n.], 2008. p. 1464–1465. Citado 3 vezes nas páginas 22, 23 e 24.

Zafaruddin, G. M.; Fadewar, H. S. Face recognition: A holistic approach review. In: *2014 International Conference on Contemporary Computing and Informatics (IC3I).* [S.l.: s.n.], 2014. p. 175–176. Citado na página 13.

ZAKARIA, Z.; ISA, N. A. M.; SUANDI, S. A. A study on neural network training algorithm for multiface detection in static images. *International Journal of Computer and Information Engineering*, World Academy of Science, Engineering and Technology, v. 4, n. 2, p. 345 – 348, 2010. Citado na página 25.

Zhang, G. P. Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, v. 30, n. 4, p. 451–462, 2000. Citado na página 40.

Zhihao, H. et al. Human emotion recognition in video. In: *Proceedings of the 2019 11th International Conference on Machine Learning and Computing.* [S.l.: s.n.], 2019. p. 374–379. Citado na página 9.

Zhu, J.; Chen, Z. Real time face detection system using adaboost and haar-like features. In: *2015 2nd International Conference on Information Science and Control Engineering.* [S.l.: s.n.], 2015. p. 404–407. Citado 2 vezes nas páginas 15 e 16.

6 Apêndice - Artigo Científico Publicado como Resultado da Pesquisa

Bove, M. S. P., Cerqueira, J. J. F., Simas Filho, E. F. (2020). *Novelty Detection Applied in Recognition of Facial Expressions.* XXIII Congresso Brasileiro de Automática, 1-8.

Novelty Detection Applied in Recognition of Facial Expressions

Márcio S. P. Bove* Jés J. F. Cerqueira**
Eduardo F. Simas Filho**

** Feira de Santana Higher Education Unit, Feira de Santana, Brazil. E-mails: bove@ufba.br, engenhariaeletrica@gruponobre.net.*
*** Electrical Engineering Graduate Program, Federal University of Bahia, Salvador, Brazil. E-mails:{jes, eduardo.simas }@ufba.br.*

Abstract: This research investigates the capacity of the Multilayer Perceptron (MLP) and Radial Basis Function (RBF) networks in the task of Novelty Detection (ND) in the recognition of facial expressions using video resources. The video data set used is produced by professional actors in the studio with basic affective states of the human face. The Viola-Jones, Kanade-Lucas-Tomasi (KLT) and Principal Component Analysis (PCA) algorithms are used in the pre-processing phase to extract features from the face. The results evaluate the performance of the MLP and RBF networks in the ND task, using new facial expressions compatible with those used in the training phase and also examines the capacity of the networks in ND using the faces of actors never before seen by the networks. In this process, the MLP and RBF networks have an accuracy of 98% for classification task, 68% and 98% for ND with data similar to the data from the training phase and 100% for ND with totally new data. Thus, this work brings together methods and techniques applied in ND using Artificial Neural Networks (ANN) aiming at the production of interactive cognition systems in the field of affective computing, based on techniques of Artificial Intelligence (AI) and Computer Vision.

Resumo: Esta pesquisa investiga a capacidade das redes Perceptron de Múltiplas Camadas (MLP) e Função de Base Radial (RBF) na tarefa de Detecção de Novidade (DN) no reconhecimento de expressões faciais usando recursos de vídeo. O conjunto de dados de vídeo utilizado é produzido por atores profissionais em estúdio com estados afetivos básicos do rosto humano. Os algoritmos Viola-Jones, Kanade-Lucas-Tomasi (KLT) e Análise de Componentes Principais (PCA) são usados na fase de pré-processamento para extrair atributos da face. Os resultados avaliam o desempenho das redes MLP e RBF na tarefa DN, usando novas expressões faciais compatíveis com as utilizadas na fase de treinamento e também examinam a capacidade das redes em DN usando as faces de atores nunca antes vistos pelas redes. Neste processo, as redes MLP e RBF têm uma precisão de 98% para tarefa de classificação, 68% e 98% para DN com dados semelhantes aos dados da fase de treinamento e 100% para DN com dados totalmente novos. Assim, este trabalho reúne métodos e técnicas aplicadas na DN utilizando Redes Neurais Artificiais (RNA), visando a produção de sistemas interativos de cognição no campo da computação afetiva, baseados em técnicas de Inteligência Artificial (IA) e Visão Computacional.

Keywords: Novelty Detection, Neural Networks, Viola-Jones, Kanade-Lucas-Tomasi, Principal Component Analysis.

Palavras-chaves: Detecção de Novidade, Redes Neurais, Viola-Jones, Kanade-Lucas-Tomasi, Análise de Componentes Principais.

1. INTRODUCTION

Novelty detection (ND) consists of the ability to identify new or previously unknown situations, being considered an extremely complex and important task, so that many methods are proposed for ND (Pimentel et al., 2014; Markou and Singh, 2003a,b; Domingues et al., 2018). Approaches based on different categories such as probability, reconstruction, domain, information theory and distance can be used to ND (Pimentel et al., 2014). Thus, ND can be compared to a classifier that produces results for normal patterns and another for unknown patterns, where a description of normality is learned by fitting a model to the set of normal examples, and previously invisible patterns are tested by comparing their score of novelty with some decision limit (Sameer and Markou, 2004).

In different areas, as industrial monitoring, sensor networks, robotics, signal processing, computer vision, pattern recognition, text mining, information security, diagnosis and medical supervision (Pimentel et al., 2014; Oliveira, Moisés A. et al., 2020), studies in the scope of ND has contributed to the development of intelligent

systems. In the field of affective computing, natural verbal and non-verbal signals of human emotions, cognitions, perceptions and behaviors are used in the production of efficient and satisfactory interaction systems between man and machine, where the human face is used of several applications (Calvo and D'Mello, 2010; Chatterjee and Chandran, 2016).

In this work methods as Viola-Jones (Viola and Jones, 2001), Kanade-Lucas-Tomasi (KLT) (Chai and Shi, 2011; Barnouti et al., 2018), Principal Component Analysis (PCA) (Liu and Kau, 2017) and Artificial Neural Networks (ANN) (Markou and Singh, 2003b; Pimentel et al., 2014) are used in the production of a compact algorithm with low computational cost with the purpose of ND in the recognition of facial expressions in real time using video feature.

In the pre-processing phase the Viola-Jones, KLT and PCA algorithms are used to extract features from the face and produce the features vector. The Viola-Jones algorithm initially detects the face on video (Viola and Jones, 2001) and consecutively the KLT algorithm identifies facial features and tracks the face throughout the video (Chai and Shi, 2011; Barnouti et al., 2018). The PCA algorithm characterized as a statistical method used to extract the main components of a data set (Liu and Kau, 2017) is employed in this work to extract features of the data obtained with the KLT algorithm, where only the first component is used for coding the human face, enabling the production of the vector for training and network testing.

In the processing phase, ANN classifies or ND in the analysis of facial expressions in video. The Multilayer Perceptron (MLP) and Radial Basis Function (RBF) networks are evaluated for their respective performance in assertiveness to ND. The used haves video database a dynamic and multimodal set of facial and vocal expressions in English assessing the emotional authenticity of professional actors for basic affective states (Livingstone, 2018).

In this way, this article investigates ND for a multi-class approach using the MLP and RBF neural networks, evaluating facial expressions similar to those used in the training phase and evaluating entirely new facial expressions. Consecutively motivated in the integration of artificial intelligence and computer vision methods and techniques to implement a compact algorithm that detects new facial expressions or classifies facial expressions in real time video stream.

The organization of the paper is as following. In section 2 are presented considerations for the use of ANN and significant aspects to enable the MLP and RBF networks for ND. Besides main aspects of the KLT and PCA algorithms to extract facial features from video are highlighting. Section 3 presents details of the data set and also presents the methods used to enable the MLP and RBF networks for ND. Section 4 presents the results obtained with the application of the algorithm developed for ND, as well as graphical results that evaluate the MLP and RBF networks for ND using facial expressions similar to the expressions used in the training phase and facial expressions of new actors. The conclusions are presented in section 5.

2. BACKGROUND

In this section, initially the aspects for ND using ANN are described. Then, the KLT and PCA methods are presented as a resource for extracting face attributes.

2.1 ANN

ANN are conceptualized as adaptive computational systems that learn from data representative of the problem by using training to adjust their synaptic weights. Its ability to learn complex boundaries to identify classes and the ability to model implicit data autonomously makes neural networks a widely used method for ND (Markou and Singh, 2003b).

In practice, ANNs do not automatically perform ND because they act as discriminators, not as detectors (Markou and Singh, 2003b), requiring training, adjustments and tests to determine limits to perform ND (Hodge and Austin, 2004). ND using ANN considers the evaluation of the results presented in the output layer of the network compared based on a limit value that makes it possible to identify when a vector displayed at the network entrance different from the vector used in the training phase (Augusteijn and Folkert, 2002; Vasconcelos and Fairhurst, 1995; Sameer and Markou, 2004).

In applications with video processing in which the same object may change gradually during operation due to different lighting conditions, exposure times and other reasons the neural networks becomes a very useful technique (Markou and Singh, 2003b). Successively because neural networks do not need data recycling as in the statistic methods for detecting new events (Markou and Singh, 2003b). In several applications, the supervised learning neural networks most used for ND in a multi-class approach are MLP and RBF (Markou and Singh, 2003b; Barreto and Frota, 2012).

Markou and Singh (2006) search in a model for ND with the image sequence analysis using neural networks. This model uses experiments with video-based image sequence data containing several novel classes. In this process, the neural network MLP is used in three configurations, with the function softmax, without rejection filter and with rejection filter. The best results are presented by the network with the rejection filter.

Vasconcelos and Fairhurst (1995) investigate the ability to ND using standard the MLP networks, MLP with Gaussian Activation Function (GMLP) and the RBF. In this evaluation is possible to identify the reasons for the unreliability of standard MLP networks in rejecting unknown patterns and consecutively that alternative configurations such as the GMLP are candidates for a more reliable structure for ND. The researchers also present technical characteristics of the RBF network that justify its greater capacity to ND compared to the MLP and GMLP networks.

Using a example from the field of speech recognition Albrecht et al. (2000) show the functioning of a generalized RBF network that can self-organize to form a Bayesian classifier and ND. For this purpose, the researchers introduce stochastic rules that concern the centers, shapes and

widths of the receptive fields of neurons allowing a joint optimization of all network parameters for ND.

2.2 MLP

MLP networks are successfully applied to solve complex problems using the backpropagation algorithm. The use of the Gaussian activation function for the MLP network forces the receptive field of neurons to be more selective, being activated only for a restricted region of the input space, optimizing the performance of the network for ND (Barreto and Frota, 2012; Vasconcelos and Fairhurst, 1995). The parameterization of the GMLP network with only a single hidden layer also improves its ability for ND, so that this technique allows to detect arbitrarily complex class limits (Hodge and Austin, 2004).

In cases where the GMLP network has a few hundred neurons in the hidden layer, its sensitivity to detect new patterns can again be amplified using the training algorithm Levenberg-Marquardt (Hagan and M.B., 1994). The Levenberg-Marquardt algorithm is an approximation to Newton's Method. The use of the softmax function in the output layer of the GMLP network makes it possible to visualize the activation values of neurons in a probabilistic way, allowing to more accurately distinguish the reference value for ND (Sameer and Markou, 2004). The reference value predefined by the user comes from the test process of the trained network with the vectors used in the training phase together with the test of vectors never before seen by the network. This testing process reveals the reference activation value for the output neurons to normal patterns and also shows the different levels of activation of the output neurons for unknown vectors. In this way, an alternative for ND can be achieved with the calculation of the distance of the winning neuron for a reference value defined by the user in the test with the normal vectors (Augusteijn and Folkert, 2002).

2.3 RBF

RBF networks have been widely used in classification problems with applications in speech recognition, medical diagnosis, handwriting recognition, image processing and fault diagnosis. In these applications, RBF networks are often used with the "Winner Takes All" (WTA) (Pont and Barrie Jones, 2002) output rule. Its only hidden layer uses functions with a radial base that make it possible to model high-dimensional spaces with higher performance for learning speed, memory requirements and generalization in comparison with MLP (Vasconcelos and Fairhurst, 1995).

The use of the Bayesian regularization training algorithm self-organizes the RBF network to form a Bayesian classifier and successively amplifies its capacity for ND (Pimentel et al., 2014; Albrecht et al., 2000). Bayesian regularization minimizes a linear combination of squared errors and weights being to able produce networks which have excellent generalization capabilities. This Bayesian regularization takes place within the Levenberg-Marquardt algorithm where it calculate the Jacobian matrix (Foresee and Hagan, 2017). Similar to the MLP network, we can use the softmax function to provide probabilistic values for the output neurons of the RBF network and define the decision threshold value for ND.

2.4 KLT

Object detection and tracking from a video-sequence are becoming an interesting subject in many computer vision applications and artificial intelligence, as bank security, border crossings, airport check-in, home monitoring, office remote meeting, prisons and factories (Barnouti et al., 2018; Chatterjee and Chandran, 2016). The KLT method to allows track a set of feature points in video frames. Its computational efficiency and robustness to scale change are relevant aspects that make its use widely feasible in the development of computer vision systems (Barnouti et al., 2018; Chatterjee and Chandran, 2016).

The structure of the KLT algorithm is composed of two phases of operation, features extraction and tracking (Chai and Shi, 2011). In the features extraction phase, initially the Viola-Jones algorithm detects the face in the video (Barnouti et al., 2018; Chatterjee and Chandran, 2016) and feature points are identified around of face. With the feature points identified on image, the task of feature tracking is to track them from frame to frame (Chai and Shi, 2011).

2.5 PCA

The human face has a complex and dynamic structure (Delac et al., 2005), being the one of the most important information in biometric sciences based on personal identification (Abbas et al., 2017). The technique of analysis and understanding of images have gained prominence in recent years with successful in applications of facial recognition (Delac et al., 2005). The PCA algorithm allows the identification of patterns in the data maintain their identification status and effectively reducing the dimensions in human face images (Abbas et al., 2017). This reducing eliminates information irrelevant or redundant for to arrives in a higher compression ratio for the first component (Liu and Kau, 2017) producing a low-dimensional representation of the input data without significant loss of the original data. The mathematical formula of PCA is based in the standard deviation, and the eigenvectors and eigenvalues, seing considered a robust technique with a process simple, fast and what works well under constrained environment for facial recognition (Abbas et al., 2017).

3. METHODOLOGY

Novelty detection or novelty recognition is based on the normal class modeling technique, that is, a description of normal data is learned by the network so that the it is able to detect new events. This methodology is appropriate for ND with static or dynamic data (Hodge and Austin, 2004). Therefore, in this section, initially are presented the details of the data set used to model the normal classes and for ND, and consecutively the methods used in the training and testing phase of the MLP and RBF networks. In Figure 1 is show the model used for ND applied in recognition of facial expressions using of video resource.

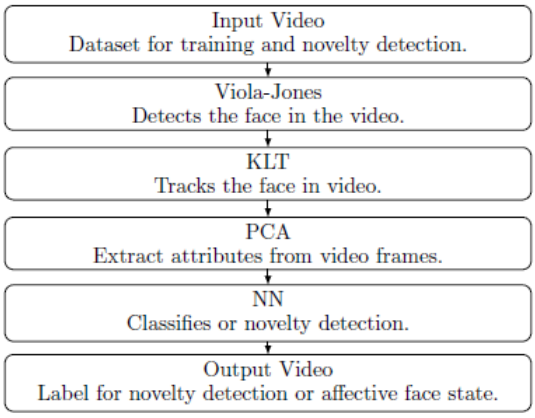

Figure 1. Model applied to novelty detection in video resource.

3.1 Dataset

The database proposed as visual resource for research, account with a selection dynamic and multimodal facial and vocal expressions assessing emotional authenticity of 24 professional actors (12 women and 12 men). The emotions calm, happy, sad, angry, fear, surprise and disgust are included. Each expression is selected on two levels of emotional arousal (normal and strong) with an additional neutral expression. All conditions are available in three data formats: audio only (16 bits, 48kHz, Type .wav), audio-video only (Width 1280 pixels, Height 720 pixels, Rate 29.97 fbs, AAC 48kHz, Type .mp4) and video-only (no sound) (Livingstone, 2018). The Figure 2 and Table 1 show the selection of video resources used in the training and testing phases of the MLP and RBF networks. In Figure 2 on the left side column, the sequence (happy, sad, angry and neutral) of actor 1 is used in the training phase of the network, in the validation of the classification of affective states and in the analysis process to determine the decision threshold for ND. On the center, the sequence (calm, fear, surprise and disgust) of actor 1 are used for ND with affective states similar to used in the training phase. On the right side, the sequence (happy, sad, calm and happy) for new actors are used for ND in totally new conditions.

Figure 2. Actors of the Database (Livingstone, 2018) used for training and ND with MLP and RBF.

Table 1. Database used for training and ND of MLP and RBF networks.

Video	Number	Phase	Affection	Frames
Actor 1	3	Training	Happy	399
Actor 1	3	Training	Sad	426
Actor 1	3	Training	Angry	426
Actor 1	3	Training	Neutral	294
Actor 1	3	Test ND	Calm	294
Actor 1	3	Test ND	Fear	336
Actor 1	3	Test ND	Surprise	318
Actor 1	3	Test ND	Disgust	351
Actor 2	3	Test ND	Happy	312
Actor 5	3	Test ND	Sad	322
Actor 6	3	Test ND	Calm	310
Actor 10	3	Test ND	Happy	320

3.2 Training Phase

In the first phase, the face is segmented by the holistic method, that is, the face is processed as a whole, considering the facial information of the nose, eyes, mouth and hair. In this process, Viola-Jones algorithm initially detects and locates the face in a video frame and in the sequence feature points are used to map the face region.

In the second step, KLT algorithm tracks the set of points considering the bounding box around the face to estimate the position of the face frame by frame according to the dynamics of the video.

In the third phase, PCA algorithm is used to calculate the first axis of the main component by extracting and reducing the dimensionality of the frames with low loss of information.

In the fourth phase, MLP and RBF networks are parameterized and trained for ND considering a single hidden layer according to the topology of Figure 3.

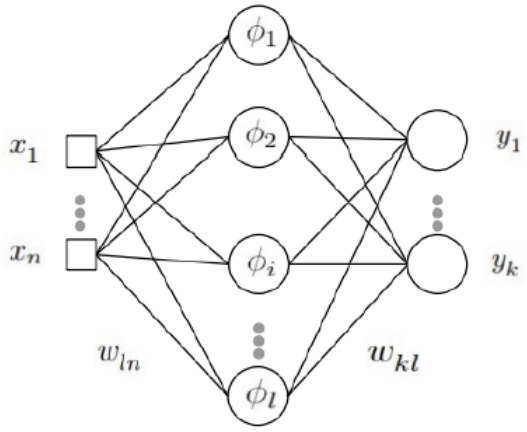

Figure 3. Topology of MLP and RBF networks.

For MLP network, the input layer own 6400 neurons, the only hidden layer own 190 neurons with the Gaussian activation function, the output layer own 4 neurons and the network synaptic weights are adjusted by the Levenberg-Marquardt training algorithm. In this case, the parameterization of the hidden layer with only 190 neu-

rons is justified for the use of the Levenberg-Marquardt training algorithm, which has better efficiency when the network has a few hundred neurons in the hidden layer and consecutively amplifies the MLP network capacity for ND (Hagan and M.B., 1994). The induced local field of the neuron $v_i(n)$ adjusts the height of the Gaussian function and the constant γ_i adjusts the radius of the function. Where the outputs are obtained by $y_k(\mathbf{x},\mathbf{w})$ with $\mathbf{x} = [x_1, x_2, x_i, ...x_n]^T$ being the attribute vector.

$$v_i(n) = \sum_{i=1}^{n} w_{ji}(n)x_i(n) \tag{1}$$

$$\phi_i(\mathbf{x},\mathbf{w}) = exp\left[-\frac{v_i(n)^2}{\gamma_i^2}\right] \tag{2}$$

$$y_k(\mathbf{x},\mathbf{w}) = \sum_{j=1}^{l} w_{kj}\phi_i(\mathbf{x},\mathbf{w}) \tag{3}$$

For the RBF network, the input layer own 6400 neurons, the only hidden layer own 100 neurons with the Gaussian activation function, the output layer own 4 neurons and the network synaptic weights are adjusted by the Bayesian Regularization training algorithm. The Bayesian regularization use Gauss-Newton approximation to the Hessian matrix. The additional overhead of this Gauss-Newton approximation is minimal when Levenberg-Marquardt optimization algorithm is used to locate the optimal weights. This training method reduce the sum of squared errors and produces small values for synaptic weights, thus result a smoother network response that intensifier the capability to ND (Foresee and Hagan, 2017). The induced local field of the neuron $v_i(n)$ adjusts the Gaussian function at the center of the observed data. This process occurs by calculating the Euclidean distance $\| x_i(n) - c_i(n) \|$, considering the input vector $\mathbf{x} = [x_1, x_2, x_i, ...x_n]^T$ and the observed data centers $\mathbf{c} = [c_1, c_2, c_i, ...c_n]^T$. The standard deviation σ_i adjusts the radius of the Gaussian function.

$$v_i(n) = \| x_i(n) - c_i(n) \| \tag{4}$$

$$\phi_i(\mathbf{x},\mathbf{c}) = exp\left[-\frac{v_i(n)^2}{2\sigma_i^2}\right] \tag{5}$$

$$y_k(\mathbf{x},\mathbf{c}) = \sum_{j=1}^{l} w_{kj}\phi_i(\mathbf{x},\mathbf{c}) \tag{6}$$

At the end of the training phase, the function $softmax(y_i)$ is used to calculate the level of activation of the output neurons of the MLP and RBF networks to obtain values with a degree of probabilistic confidence, in order to create thresholds reliable for ND. Thus, the new outputs for ND are obtained using equation (8).

$$softmax(y_i) = \frac{e^{y_i}}{\sum_k e^{y_k}} \tag{7}$$

$$z(y_i) = softmax(y_i) - \delta_i \tag{8}$$

3.3 Testing Phase

After training the networks, they are tested with the vectors of the training phase and new vectors. Vectors of the training phase are used to check the accuracy of the networks in the classification task and consecutively to define the threshold vector for ND. The limit vector $\delta(n) = [\delta_1, \delta_2, \delta_i, ...\delta_k]^T$ for ND is defined by looking at the activation levels of the winning neurons of each class. The lowest level of activation among the winning neurons of each class is defined as the decision threshold for ND. The new vectors are used this process to check the sensitivity of the networks to ND. In this condition, the respective outputs present an activation level for the winning neuron lower than the level defined as threshold for ND. Thus, when all outputs $z(y_i)$ simultaneously have an activation level lower than zero, a novelty is detected.

4. RESULTS

In this section, are presented the results obtained with the proposed algorithm for classification and ND in recognizing facial expressions with video resource. In Figure 4 at the top the Viola-Jones and KLT algorithms are used to extract features from the face by the holistic method. At the bottom, the PCA algorithm is used to calculate the first axis of the main component, extracting and reducing the dimensionality of the frames.

Figure 4. Viola-Jones, KLT and PCA extracting face attributes in video.

The Figures 5 and 6 show the results obtained for the classification test with the MLP and RBF networks considering the affective states (happy, sad, angry and neutral) of the actor 1 as described in Table 1. These results represent the learning of normal patterns, qualifying the MLP and RBF networks to classify the affective states of actor 1 with 98% accuracy when evaluating video frames. Confusions occur between sad and angry affective states owing to the similarity of some facial expressions.

In Figure 7, the proposed algorithm evaluates actor 1 facial expressions. Application of the tag respective affective state is produced in real time, confirming the efficiency in the integration of Viola-Jones, KLT, PCA and ANN methods.

	Happy	Sad	Angry	Neutral
Happy	399	0	0	0
Sad	0	411	15	0
Angry	0	15	411	0
Neutral	0	0	0	294

Figure 5. Results of the MLP network for the classification of Happy, Sad, Angry and Neutral affective states represented by the confusion matrix.

	Happy	Sad	Angry	Neutral
Happy	399	0	0	0
Sad	0	408	12	0
Angry	0	18	414	0
Neutral	0	0	0	294

Figure 6. Results of the RBF network for the classification of Happy, Sad, Angry and Neutral affective states represented by the confusion matrix.

Figure 7. Classification in video resource for facial expressions to Actor 1 (Happy, Sad, Angry and Neutral).

In Figure 8, the algorithm evaluates actor 1 facial expressions for ND that are similar to the facial expressions used in the training phase. This selection of frames of actor 1, introduces in the MLP and RBF networks small disturbances that are more difficult to be recognized as novelty. At the bottom of the Figure 8, the selection of frames presents results for ND with facial expressions of new actors that produce into the MLP and RBF networks gross disturbances.

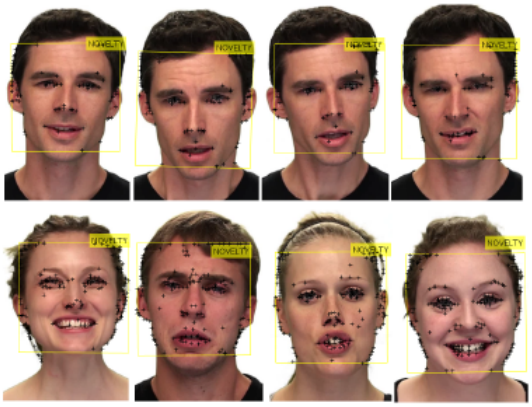

Figure 8. Novelty detection in video resource for facial expressions to Actor 1 (Calm, Fear, Surprise and Disgusted), Actor 2 (Happy), Actor 5 (Sad), Actor 6 (Calm) and Actor 10 (Happy).

The results presented in Figures 9 and 10 for ND are obtained using the MLP network based on the data described in Table 1. The algorithm produces results with an efficiency of 100% for the detection of new actors. However, when evaluating actor 1's videos, this efficiency drops for 68%. This less accurate result is represented by video frames that have not been recognized as new. The video frames not recognized as new are classified with the closest facial expression used in the training phase. These results confirm the robustness of the MLP network for ND when input data produces gross errors and stability for a range when the input data produces a small disturbance.

That way, a system can perfectly adjust the noise to the data provided, leading to the phenomenon of overfitting. In overfitting situations, the learning process may be able to achieve lower training error levels because the learned function has tried to fit all data as close as possible (Chen-Chia Chuang et al., 2000). However, another set of data, which is not used in any way in the training process, have errors significantly be increased because the noise in the training patterns is different from that in the testing patterns (Chen-Chia Chuang et al., 2000).

The experimental results produced by Markou and Singh (2006) for ND in the analysis of video image sequences based on MLP neural networks show different levels of performance according to the Z metric. These results are generated using four different video sequences, where the best performance of Z with 92% occurs in the experiment with the second video sequence. Considering the results produced in this research and the results obtained in the experiment carried out by Markou and Singh (2006) it is confirmed that the MLP network can be adapted and adjusted to be used as an efficient alternative to perform ND.

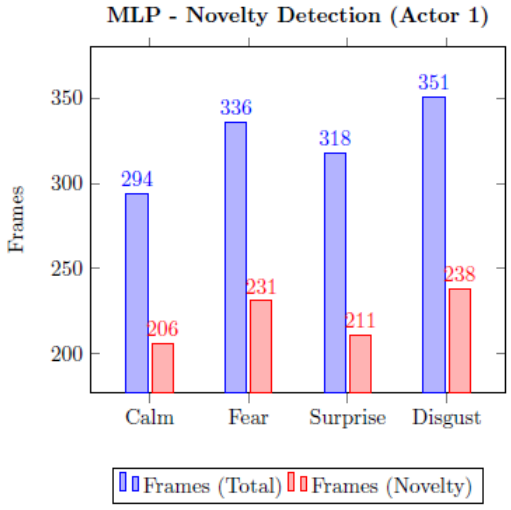

Figure 9. ND for video frames with Actor 1 (Calm, Fear, Surprise and Disgust).

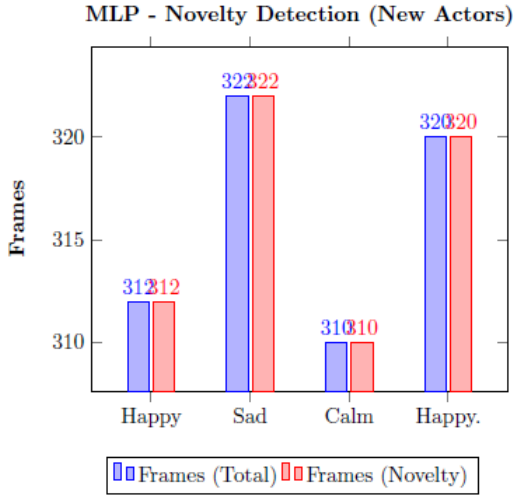

Figure 10. ND for video frames with Actor 2 (Happy), Actor 5 (Sad), Actor 6 (Calm) and Actor 10 (Happy).

In Figures 11 and 12 the algorithm produces results with an efficiency of 100% for the detection of new actors, but when evaluating actor 1's videos, this efficiency reduces for 98% owing the classification of facial expressions that are similar to those used in the training phase.

Different on the MLP network, each hidden unit in the RBF network responds to a receptive field located in the input space according to the Euclidean distance calculation. As a result, the network output reaches its maximum when the input pattern is close to the centroid and decreases monotonically when it is further away from the centroid, making it ideal for ND (Markou and Singh, 2003b). The self-organized Bayesian training algorithm updates the synaptic weights according to the Levenberg-Marquardt optimization (Foresee and Hagan, 2017), minimizing the errors and consecutively producing a robust response in the analysis of gross errors or small.

The results produced for ND by Albrecht et al. (2000) with the RBF network for voice recognition are based on a cumulative distribution function of log-likelihood. Where new uniformly distributed voice patterns are safely rejected and new patterns similar to training data are not safely detected for phoneme transitions. These results confirm the efficiency of the RBF network as a Bayesian classifier for ND.

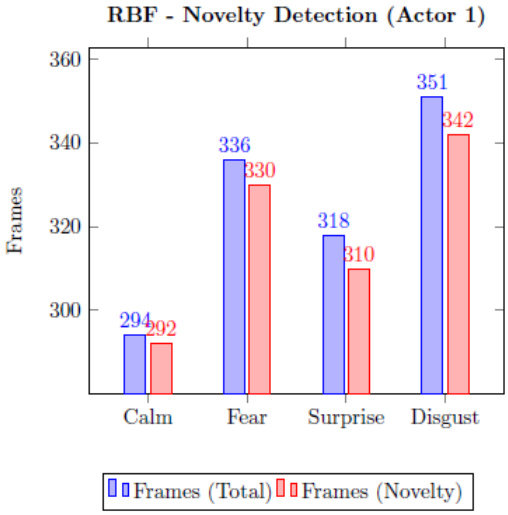

Figure 11. ND for video frames with Actor 1 (Calm, Fear, Surprise and Disgust).

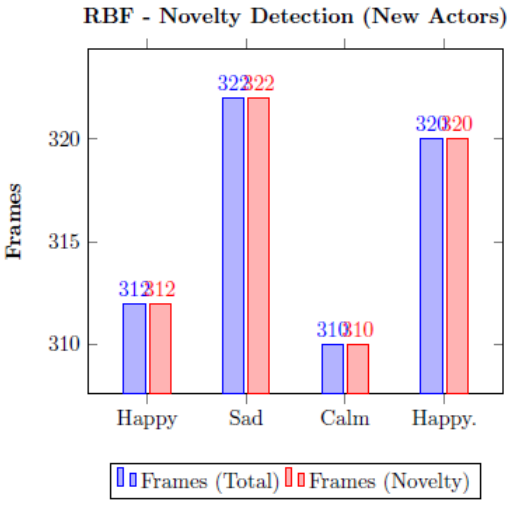

Figure 12. ND for video frames with Actor 2 (Happy), Actor 5 (Sad), Actor 6 (Calm) and Actor 10 (Happy).

5. CONCLUSION

This research investigated the capacity of MLP and RBF networks in the task of ND by evaluating facial expressions in video resources. The attributes of facial expressions are obtained by the holistic method, using the Viola-Jones, KLT and PCA algorithms, which extract attributes of the face in real-time video. The results confirm the 98% efficiency of the RBF network in ND for video frames with attributes close to the frames used in the training phase and efficiency of 100% for video frames not similar to the training data. The results of the MLP network are also validated in 100% for ND for frames without similarity to the frames used in the training phase of the network. However, when similar frames are presented for the MLP network, their efficiency reduces to a success rate of 68% to ND. Thus, this work with the integration of Viola-Jones, KLT, PCA and ANN methods confirms the ND in the evaluation of facial expressions in video resources in order to contribute to the development of intelligent systems in the field of affective computing.

REFERENCES

Abbas, E.I., Safi, M.E., and Rijab, K.S. (2017). Face recognition rate using different classifier methods based on pca. In *2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT)*, 37–38.

Albrecht, S., Busch, J., Kloppenburg, M., Metze, F., and Tavan, P. (2000). Generalized radial basis function networks for classification and novelty detection: self-organization of optimal bayesian decision. *Neural Networks*, 13, 1075–1076.

Augusteijn, M.F. and Folkert, B.A. (2002). Neural network classification and novelty detection. *International Journal of Remote Sensing*, 23, 2891–2893.

Barnouti, N.H., Al-Mayyahi, M.H.N., and Al-Dabbagh, S.S. (2018). Real-time face tracking and recognition system using kanade-lucas-tomasi and two-dimensional principal component analysis. *2018 International Conference on Advanced Science and Engineering (ICOASE), Duhok*, 24–28.

Barreto, A. and Frota, A. (2012). A unifying methodology for the evaluation of neural network models on novelty detection tasks. *Pattern Analysis and Applications*, 16, 85–89.

Calvo, R. and D'Mello, S. (2010). Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Transactions on Affective Computing*, 1, 1.

Chai, Z. and Shi, J. (2011). Improving KLT in embedded systems by processing oversampling video sequence in real-time. *2011 International Conference on Reconfigurable Computing and FPGAs*, 297–298.

Chatterjee, D. and Chandran, S. (2016). Comparative study of camshift and KLT algorithms for real time face detection and tracking applications. *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata*, 63–64.

Chen-Chia Chuang, Shun-Feng Su, and Chin-Ching Hsiao (2000). The annealing robust backpropagation (arbp) learning algorithm. *IEEE Transactions on Neural Networks*, 11(5), 1067–1077.

Delac, K., Grgic, M., and Grgic, S. (2005). Independent comparative study of pca, ica, and lda on the feret data set. *International Journal of Imaging Systems and Technology*, 15, 252–254.

Domingues, R., Filippone, M., Michiardi, P., and Zouaoui, J. (2018). A comparative evaluation of outlier detection algorithms: Experiments and analyses. *Pattern Recognition*, 74, 407–408.

Foresee, F.D. and Hagan, M.T. (2017). Gauss-newton approximation to bayesian learning. *Proceedings of International Conference on Neural Networks (ICNN'97)*, 3, 1930–1935.

Hagan, M. and M.B., M. (1994). Training feedforward networks with the marquardt algorithm. *IEEE Transactions on Neural Networks*, 5, 989–993.

Hodge, V.J. and Austin, J. (2004). A survey of outlier detection methodologies. *Artificial Intelligence Review*, 22, 85–122.

Liu, Y. and Kau, L. (2017). Scalable face image compression based on principal component analysis and arithmetic coding. *2017 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW)*, 256.

Livingstone, R. (2018). The ryerson audio-visual database of emotional speech and song. . URL `https://doi.org/10.1371/journal.pone.0196391`.

Markou, M. and Singh, S. (2003a). Novelty detection: a review—part 1: statistical approaches. *Signal Processing*, 83, 2481.

Markou, M. and Singh, S. (2003b). Novelty detection: a review—part 2: neural network based approaches. *Signal Processing*, 83, 2499–2503.

Markou, M. and Singh, S. (2006). A neural network-based novelty detector for image sequence analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10), 1664–1675.

Oliveira, Moisés A., Simas Filho, Eduardo F., Albuquerque, Maria C. S., Santos, Ygor T. B., Da Silva, Ivan C., and Farias, Cláudia T. T. (2020). Ultrasound-based identification of damage in wind turbine blades using novelty detection. *Ultrasonics*, 1–2.

Pimentel, A., Clifton, D., Clifton, L., and Tarassenko, L. (2014). A review of novelty detection. *Signal Processing*, 99, 217–232.

Pont, L.Y. and Barrie Jones, N. (2002). Improving the performance of radial basis function classifiers in condition monitoring and fault diagnosis applications where 'unknown' faults may occur. *Pattern Recognition Letters*, 23, 569–572.

Sameer, S. and Markou, M. (2004). An approach to novelty detection applied to the classification of image regions. *IEEE Transactions on Knowledge and Data Engineering*, 16, 396–397.

Vasconcelos, G.C. and Fairhurst, D.B. (1995). Investigating feedforward neural networks with respect to the rejection of spurious patterns. *Pattern Recognition Letters*, 16, 208–209.

Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1, 511.

ANEXO A – Código para Extração de Atributos Faciais em Fluxo de Vídeo

CÓDIGO PARA EXTRAÇÃO DE ATRIBUTOS FACIAIS EM FLUXO DE VÍDEO

```
% Abre contagem do tempo
tic

% Cria um objeto detector em cascata
faceDetector = vision.CascadeObjectDetector();

% Variável para contagem
tot = 0;

% Lê endereço de vídeo
videoFileReader = vsion.VideoFileReader('titulo_video.avi');
videoFrame = step (videoFileReader);

% Reduz tamanho do vídeo
videoFrame = imresize (videoFrame, 0.5);

% Desenha caixa delimitadora ao redor do rosto
bbox = step(faceDetector, videoFrame);
videoFrame = insertShape(videoFrame, 'Rectangle', bbox);
figure;
imshow(videoFrame);
title('Face Detectada');

% Converte a primeira caixa em uma lista de 4 pontos
% Permite visualizar a rotação do objeto
bboxPoints = bbox2points (bbox (1, :));

% Detecta pontos na região do rosto
points = detectMinEigenFeatures(rgb2gray(videoFrame), 'ROI', bbox);

% Mostra pontos na região do rosto
figure, imshow(videoFrame), hold on, title('Características Detectadas');
plot(points);

% Cria rastreador de pontos
pointTracker = vision.PointTracker('MaxBidirectionalError', 2);
points = points.Location;
initialize(pointTracker, points, videoFrame);

% Cria um objeto player de vídeo para exibir quadros de vídeo
videoPlayer = vision.VideoPlayer('Position',...
    [30 30 [size(videoFrame, 2), size(videoFrame, 1)]+30]);
```

```
Rastrear a Face
oldPoints = points;
while ~isDone(videoFileReader)
       % Incremento variável de contagem
       tot = tot+1;

       % Próximo quadro
       videoFrame = step(videoFileReader);

       % Reduz tamanho do vídeo
       videoFrame = imresize (videoFrame, 0.5);

       % Rastreia pontos
       [points, isFound] = step(pointTracker, videoFrame);
       visiblePoints = points(isFound, :);
       oldInliers = oldPoints(isFound, :);

  % Define no mínimo 2 pontos
   if size(visiblePoints, 1) >= 2;
       % Estima a transformação geométrica entre pontos antigos e novos
       [xform, inlierIdx] = estimateGeometricTransform2D (...
       oldInliers, visiblePoints, 'similarity', 'MaxDistance', 4);
       oldInliers    = oldInliers (inlierIdx, :);
       visiblePoints = visiblePoints(inlierIdx, :);

       % Aplica a transformação aos pontos da caixa
       bboxPoints = transformPointsForward (xform, bboxPoints);

       % Insere caixa ao redor do objeto rastreado
       bboxPolygon = reshape(bboxPoints', 1, []);
       videoFrame = insertShape (videoFrame, 'Polygon', bboxPolygon, ...
           'LineWidth', 2);

       % Exibe pontos rastreados
        videoFrame = insertMarker (videoFrame, visiblePoints, '+', ...
        'Color', 'white');

       % Reinicializa os pontos
         oldPoints = visiblePoints;
         setPoints (pointTracker, oldPoints);
   end
       % Exibi o quadro de vídeo
       step (videoPlayer, videoFrame);

       % Corta imagem ao redor da caixa delimitadora e ajusta para 80x80
       frame = imresize (double (imcrop(videoFrame,bbox)),[80 80]);

       % Remodela a matriz frame
       framex = reshape (frame, size(frame,1)*size(frame,2),3);

       % Frame de 80x80 com primeiro componente do PCA
       coeff = pca(framex);
    framepca = reshape(framex*coeff(:,1),size(frame,1),size(frame,2));
      subplot (1,1,1); imshow(framepca,[]); title('Componente 1');
end
% Limpa
release(videoFileReader); release(videoPlayer); release(pointTracker);

% Finaliza contagem do tempo
toc
```

ANEXO B – Código para Treinamento das Redes Neurais MLP e RBF

CÓDIGO DE TREINAMENTO DA REDE NEURAL MLP

```
% Importa dados de treinamento
VinT = importdata('VinT.mat');
VoutT = importdata('VoutT.mat');

% Rede com uma única camada oculta
netGMLP.numLayers = 1;

% Rede com 100 neurônios na camada oculta
netMLP = patternnet(30);

% Algoritmo de Treinamento Gradiente Conjugado em Escala
netMLP.trainFcn = 'trainscg';

% Rede com função de ativação gaussiana.
netGMLP.layers{1}.transferFcn = 'radbas';

% Parâmetros de treinamento.
netGMLP.divideParam.trainRatio = 70/100;
netGMLP.divideParam.valRatio = 15/100;
netGMLP.divideParam.testRatio = 15/100;

% Treina rede.
[netGMLP,trGMLP] = train(netGMLP,VinT,VoutT);


% Saída da rede (teste com vetores de entrada)
yGMLP = netGMLP(VinT);

% Saída da rede com função softmax
yGMLPs = softmax(yGMLP);


% Parâmetros de performance do treinamento
eGMLP = gsubtract(VoutT,yGMLP);
performanceMLP = perform(netGMLP,VoutT,yGMLP);
tindGMLP = vec2ind(VoutT);
yindGMLP = vec2ind(yGMLP);
percentErrors = sum(tindGMLP ~= yindGMLP)/numel(tindGMLP);

% Plotagem
view(netGMLP)
figure, plotperform(trGMLP)
figure, plottrainstate(trGMLP)
figure, ploterrhist(eGMLP)
figure, plotconfusion(VoutT,yGMLP)
figure, plotroc(VoutT,yGMLP)

% Salva
save('netGMLP.mat','netGMLP')
save('yGMLP.mat','yGMLP')
save('eGMLP.mat','eGMLP')
save('yGMLPs.mat','yGMLPs')
```

CÓDIGO DE TREINAMENTO DA REDE NEURAL RBF

```
% Importa dados de treinamento
VinT = importdata('VinT.mat');
VoutT = importdata('VoutT.mat');

% Meta Erro
mse =0;

% Raio da função gaussiana
raio = 20;

% Número de neurônios na camada oculta
neuronios=200;

% Passo de execução
passo = 5;

% Cria rede RBF
[netRBF Perf]= newrb(VinT,VoutT,mse,raio,neuronios,passo);

%Treinamento com o algoritmo Bayesian Regularization
netRBF.trainFcn='trainbr';

% Número de épocas
netRBF.trainParam.epochs = 200;

% Saída da rede (teste com vetores de entrada)
yRBF = sim(netRBF,VinT);


% Saída da rede com função softmax
yRBFs = softmax(yRBF);

% Parâmetros de performance do treinamento
e = gsubtract(VoutT,yRBF);
performance = perform(netRBF,VoutT,yRBF);

% Plotagem
view(netRBF)
figure, ploterrhist(e);
figure, plotconfusion(VoutT,yRBF);
figure, plotroc(VoutT,yRBF);

% Salva
save('netRBF.mat','netRBF');
save('yRBF.mat','yRBF');
save('e.mat','e');
save('yRBFs.mat','yRBFs');
save('performance.mat',' performance')
```

ANEXO C – Código para Detecção de Novidades em Fluxo de Vídeo

CÓDIGO DN COM REDE NEURAL MLP

```
% Abre contagem
tic

% Cria um objeto detector em cascata
faceDetector = vision.CascadeObjectDetector();

% Variável para contagem
tot =0;

% Lê endereço de vídeo
videoFileReader = vsion.VideoFileReader('titulo_video.avi');
videoFrame = step (videoFileReader);

%Reduz tamanho de vídeo
videoFrame = imresize(videoFrame,0.5);

% Desenha caixa delimitadora ao redor do rosto
bbox = step(faceDetector, videoFrame);
videoFrame = insertShape(videoFrame, 'Rectangle', bbox);

% Converte a primeira caixa em uma lista de 4 pontos
% Permite visualizar a rotação do objeto
bboxPoints = bbox2points(bbox(1, :));


% Detecta pontos na região do rosto
points = detectMinEigenFeatures(rgb2gray(videoFrame), 'ROI', bbox);

% Cria rastreador de pontos
pointTracker = vision.PointTracker('MaxBidirectionalError', 2);
points = points.Location;
initialize(pointTracker, points, videoFrame);

% Cria um objeto player de vídeo para exibir quadros de vídeo
videoPlayer  = vision.VideoPlayer('Position',...
    [30 30 [size(videoFrame, 2), size(videoFrame, 1)]+30]);

% Importa rede treinada
netGMLP = importdata('treino/netGMLP.mat');

% Váriavel Estado Afetivo
 EstadoAfetivo=0;
```

```
%Rastrear a Face
oldPoints = points;
while ~isDone(videoFileReader)

    % Incremento variável de contagem
    tot = tot+1;

    % Próximo quadro
    videoFrame = step(videoFileReader);

    % Reduz tamanho do vídeo
    videoFrame = imresize(videoFrame,0.5);

    % Rastreia pontos%
    [points, isFound] = step(pointTracker, videoFrame);
    visiblePoints = points(isFound, :);
    oldInliers = oldPoints(isFound, :);

   % Define no mínimo 2 pontos
   if size(visiblePoints, 1) >= 2;
    % Estima a transformação geométrica entre pontos antigos e novos
    [xform, oldInliers, visiblePoints]=estimateGeometricTransform(...
     oldInliers, visiblePoints, 'similarity', 'MaxDistance', 4);

    % Aplica a transformação aos pontos da caixa
    bboxPoints = transformPointsForward(xform, bboxPoints);

    % Insere caixa ao redor do objeto rastreado
    bboxPolygon = reshape(bboxPoints', 1, []);
    videoFrame = insertShape(videoFrame, 'Polygon', bboxPolygon, ...
            'LineWidth', 1);

    % Inserir texto
    position = [350 320];
    texto = [EstadoAfetivo];
    videoFrame = insertText(videoFrame,position,texto, ...
    'FontSize',14,'AnchorPoint','LeftBottom');


    % Exibe pontos rastreados
    videoFrame = insertMarker(videoFrame, visiblePoints, '+', ...
    'Color', 'black');

    % Reinicializa os pontos
    oldPoints = visiblePoints;
    setPoints(pointTracker, oldPoints);
    end

    % Exibi o quadro de vídeo
    step(videoPlayer, videoFrame);

    % Corta imagem ao redor da caixa delimitadora e ajusta para 80x80
    frame = imresize (double (imcrop(videoFrame,bbox)),[80 80]);

    % Remodela a matriz frame
    framex = reshape (frame, size(frame,1)*size(frame,2),3);

    % Frame de 80x80 com primeiro componente do PCA
    coeff = pca(framex);
    framepca = reshape(framex*coeff(:,1),size(frame,1),size(frame,2));
    framepcat = reshape(framepca,[],1);
```

```
    % Limiar para DN
    lim = [0.475;0.475;0.475;0.4744];

    % Saída rede
    out (:,tot) = netGMLP(framepcat);

    % Nova saída
    z = softmax (out)-lim;

    z1 = z(1)>=0;
    z2 = z(2)>=0;
    z3 = z(3)>=0;
    z4 = z(4)>=0;

     if z1 == 1
       EstadoAfetivo ='NEUTRO';

    elseif z2 == 1
       EstadoAfetivo ='FELIZ';

    elseif z3 == 1
       EstadoAfetivo ='RAIVA';

    elseif z4 == 1
       EstadoAfetivo ='TRISTE';

    else
      EstadoAfetivo ='NOVIDADE';
    end

end

% Clean up
release(videoFileReader);
release(videoPlayer);
release(pointTracker);

% Finaliza contagem
toc
```

CÓDIGO DN COM REDE NEURAL RBF

```
% Abre contagem
tic

% Cria um objeto detector em cascata
faceDetector = vision.CascadeObjectDetector();

% Variável para contagem
tot =0;

% Lê endereço de vídeo
videoFileReader = vsion.VideoFileReader('titulo_video.avi');
videoFrame = step (videoFileReader);

%Reduz tamanho de vídeo
videoFrame = imresize(videoFrame,0.5);

% Desenha caixa delimitadora ao redor do rosto
bbox = step(faceDetector, videoFrame);
videoFrame = insertShape(videoFrame, 'Rectangle', bbox);

% Converte a primeira caixa em uma lista de 4 pontos
% Permite visualizar a rotação do objeto
bboxPoints = bbox2points(bbox(1, :));


% Detecta pontos na região do rosto
points = detectMinEigenFeatures(rgb2gray(videoFrame), 'ROI', bbox);

% Cria rastreador de pontos
pointTracker = vision.PointTracker('MaxBidirectionalError', 2);
points = points.Location;
initialize(pointTracker, points, videoFrame);

% Cria um objeto player de vídeo para exibir quadros de vídeo
videoPlayer  = vision.VideoPlayer('Position',...
    [30 30 [size(videoFrame, 2), size(videoFrame, 1)]+30]);

% Importa rede treinada
netRBF = importdata('treino/netRBF.mat');

% Váriavel Estado Afetivo
 EstadoAfetivo=0;
```

```
%Rastrear a Face
oldPoints = points;
while ~isDone(videoFileReader)

    % Incremento variável de contagem
    tot = tot+1;

    % Próximo quadro
    videoFrame = step(videoFileReader);

    % Reduz tamanho do vídeo
    videoFrame = imresize(videoFrame,0.5);


    % Rastreia pontos%
    [points, isFound] = step(pointTracker, videoFrame);
    visiblePoints = points(isFound, :);
    oldInliers = oldPoints(isFound, :);

   % Define no mínimo 2 pontos
   if size(visiblePoints, 1) >= 2;
    % Estima a transformação geométrica entre pontos antigos e novos
    [xform, oldInliers, visiblePoints]=estimateGeometricTransform(...
     oldInliers, visiblePoints, 'similarity', 'MaxDistance', 4);

    % Aplica a transformação aos pontos da caixa
    bboxPoints = transformPointsForward(xform, bboxPoints);

    % Insere caixa ao redor do objeto rastreado
    bboxPolygon = reshape(bboxPoints', 1, []);
    videoFrame = insertShape(videoFrame, 'Polygon', bboxPolygon, ...
            'LineWidth', 1);

    % Inserir texto
    position = [350 320];
    texto = [EstadoAfetivo];
    videoFrame = insertText(videoFrame,position,texto, ...
    'FontSize',14,'AnchorPoint','LeftBottom');

    % Exibe pontos rastreados
    videoFrame = insertMarker(videoFrame, visiblePoints, '+', ...
    'Color', 'black');

    % Reinicializa os pontos
    oldPoints = visiblePoints;
    setPoints(pointTracker, oldPoints);
    end

    % Exibi o quadro de vídeo
    step(videoPlayer, videoFrame);

    % Corta imagem ao redor da caixa delimitadora e ajusta para 80x80
    frame = imresize (double (imcrop(videoFrame,bbox)),[80 80]);

    % Remodela a matriz frame
    framex = reshape (frame, size(frame,1)*size(frame,2),3);

    % Frame de 80x80 com primeiro componente do PCA
    coeff = pca(framex);
    framepca = reshape(framex*coeff(:,1),size(frame,1),size(frame,2));
    framepcat = reshape(framepca,[],1);
```

```
        % Limiar para DN
        lim = [0.4;0.4;0.4;0.37];

        % Saída rede
        out (:,tot) = netRBF(framepcat);

        % Nova saída
        z = softmax (out)-lim;

        z1 = z(1)>=0;
        z2 = z(2)>=0;
        z3 = z(3)>=0;
        z4 = z(4)>=0;

        if z1 == 1
           EstadoAfetivo ='NEUTRO';

        elseif z2 == 1
           EstadoAfetivo ='FELIZ';

        elseif z3 == 1
           EstadoAfetivo ='RAIVA';

        elseif z4 == 1
           EstadoAfetivo ='TRISTE';

        else
          EstadoAfetivo ='NOVIDADE';
        end

end

% Clean up
release(videoFileReader);
release(videoPlayer);
release(pointTracker);

% Finaliza contagem
toc
```