

Universidade Federal de Minas Gerais  
Escola de Engenharia  
Curso de Graduação em Engenharia de Controle e Automação



## **Aplicação de Redes Neurais em Microcontrolador Embarcado**

Relatório de Atividades PFC1

Orientador: Prof. Antônio de Pádua Braga, Dr.  
Aluno: João Pedro Miranda Marques  
Matricula: 2017050495

1 de junho de 2022

## Sumário

1	Resumo	2
2	Introdução a Redes Neurais Artificiais	2
3	Validação do Hardware	2
4	Artigos Estudados	3

# 1 Resumo

Com o objetivo de concretizar estudos nas áreas de Redes Neurais Artificiais e em Arquitetura e Organização de Computadores esse projeto final de curso tem por finalidade implementar uma rede neural em um módulo microcontrolador ESP32 de arquitetura ARM.

## 2 Introdução a Redes Neurais Artificiais

Estudos ao livro *Redes neurais artificiais: teoria e aplicações* [3] foram realizados de forma a gerar maior compreensão sobre o assunto. A princípio, foi estudado o primeiro modelo matemático de neurônios artificiais, o neurônio MCP<sup>1</sup>. Em seguida, implementados na linguagem python o modelo de neurônio de aproximação linear Adaline<sup>2</sup> e de um modelo de Adaline com uma função de ativação, tornando um neurônio classificador com melhores resultados quando comparado ao MCP, o neurônio de classificação Perceptron Simples.

Entender o funcionamento do modelo computacional de um neurônio permite também melhor compreensão dos estudos de redes de neurônios. Para a aplicação a ser desenvolvida a ideia inicial é de detecção facial por meio de reconhecimento de imagens. Portanto, estudos sobre redes neurais convolucionais (CNN)<sup>3</sup> estão sendo feitos por se tratar de um modelo de rede utilizado para tratamento de imagens.

## 3 Validação do Hardware

O microcontrolador selecionado até o momento é o ESP32 da empresa espressif<sup>4</sup>. Esse microcontrolador apresenta dois núcleos de processamento com 40MHz. Apresenta módulo bluetooth e Wifi integrados. Um módulo mais completo é vendida pelo nome de espcam, onde apresenta uma câmera OV2640, sendo assim capaz de ser utilizado em aplicações com imagens.

O módulo espcam apresenta uma biblioteca de código aberto no GitHub chamada ESP-WHO<sup>5</sup> com a implementação de uma rede neural para detecção e reconhecimento facial. Foi realizado um teste desse projeto e os resultados foram aceitáveis. Portanto, validamos que o microcontrolador é capaz de processar uma aplicação de Redes Neurais Artificiais.

---

<sup>1</sup>MCP: McCulloch-Pitts Neuron Mankind's First Mathematical Model Of A Biological Neuron..

<sup>2</sup>Adaline: Adaptive Linear Neuron or later Adaptive Linear Element.

<sup>3</sup>CNN: Convolutional Neural Networks.

<sup>4</sup>espressif.com.

<sup>5</sup>ESP-WHO: <https://github.com/espressif/esp-who>.

## 4 Artigos Estudados

O artigo *Neural Networks Regularization With Graph-Based Local Resampling*[2] se refere ao tratamento dos dados utilizando o Grafo de Gabriel[5] antes do treinamento e inferência de redes neurais baseadas em perceptron com projeções randômicas (ELM). Método esse chamado de *Graph-based Local Resampling of perceptron-like neural networks with random projections (RN-ELM)*. O algoritmo consiste em tratar os dados geometricamente para o treinamento da rede neural. Problemas de classificação multi-objetivo<sup>6</sup>, quando aplicado esse método, possuem boa eficiência devido a complexidade da rede treinada ser reduzida quando os dados de entrada são tratados.

O artigo *Large Margin Gaussian Mixture Classifier With a Gabriel Graph Geometric Representation of Data Set Structure*[4] explicita o uso de Grafo de Gabriel para modelagem dos padrões de entrada nos problemas de classificação. O modelo é capaz de solucionar problemas de redes neurais que tem por objetivos encontrar uma solução ótima para maior capacidade de generalização e menor complexidade do algoritmo. Sendo assim problemas classificados como multiobjetivos. Problemas assim costumam ter não só uma solução ótima, mas um conjunto de soluções ótimas, denominado conjunto Pareto-Ótimas (PO). O algoritmo consiste nas etapas de construção geométrica do grafo sobre as amostras, remoção dos ruídos e, em seguida, detecção de bordas. Por fim, define-se a região de separação entre os conjuntos.



Figura 1: Exemplo de Grafo de Gabriel para o dataset Two Moons.

O artigo *Enhancing Performance of Gabriel Graph-Based Classifiers by a Hardware Co-Processor for Embedded System Applications*[1] tem uma abordagem similar ao objetivo desse trabalho. Nele foi analisada a performance de um classificador utilizando Grafo de Gabriel implementado em Hardware Embarcado. A aplicação foi implementada em um FPGA<sup>7</sup>, onde se é capaz de reorganizar blocos lógicos de processamento de forma a obter um Hardware dedicado para

<sup>6</sup>Problemas multiobjetivo: Problemas com mais de um objetivo.

<sup>7</sup>FPGA: Field-programmable gate array.

aquela tarefa. Como a implementação de Redes Neurais requer elevado cálculo matricial e paralelo, a implementação em Hardware dedicado para tal modelo apresenta boa performance quando se comparado à implementação em software de alto nível. A linguagem de descrição de Hardware utilizada para programação do FPGA foi VHDL.<sup>8</sup>

Esse artigo se difere com projeto desejado no que se diz ao Hardware escolhido. No artigo foi-se utilizado um hardware capaz de processar a rede de forma mais performática. No projeto proposto nesse trabalho o hardware utilizado será de Arquitetura ARM<sup>9</sup> com baixa capacidade de processamento. Por esse motivo, o objetivo não é ter um resultado melhor do que os implementados em computadores pessoais mas sim em obter um resultado aceitável para a utilização em uma aplicação distribuída de baixo custo.

Por fim, foi estudado diversos projetos utilizando o microcontrolador ESP32 e algoritmos de predição utilizando Redes Neurais Artificiais. Portanto, creio ser um projeto realizável.

---

<sup>8</sup>VHDL: VHSIC Hardware Description Language..

<sup>9</sup>ARM is a family of reduced instruction set computer (RISC) instruction set architectures for computer pro

## Referências

- [1] Janier Arias-Garcia et al. “Enhancing Performance of Gabriel Graph-Based Classifiers by a Hardware Co-Processor for Embedded System Applications”. Em: *IEEE Transactions on Industrial Informatics* 17.2 (fev. de 2021), pp. 1186–1196. ISSN: 1941-0050. DOI: [10.1109/TII.2020.2987329](https://doi.org/10.1109/TII.2020.2987329).
- [2] Alex D. Assis et al. “Neural Networks Regularization With Graph-Based Local Resampling”. Em: *IEEE Access* 9 (2021), pp. 50727–50737. ISSN: 2169-3536. DOI: [10.1109/ACCESS.2021.3068127](https://doi.org/10.1109/ACCESS.2021.3068127).
- [3] A. de Pádua Braga, A.C.P. de Leon Ferreira Carvalho e T.B. Ludermit. *Redes neurais artificiais: teoria e aplicações*. Livros Técnicos e Científicos, 2000. ISBN: 9788521612186. URL: <https://books.google.com.br/books?id=cUgEaAEACAAJ>.
- [4] Luiz C. B. Torres et al. “Large Margin Gaussian Mixture Classifier With a Gabriel Graph Geometric Representation of Data Set Structure”. Em: *IEEE Transactions on Neural Networks and Learning Systems* 32.3 (mar. de 2021), pp. 1400–1406. ISSN: 2162-2388. DOI: [10.1109/TNNLS.2020.2980559](https://doi.org/10.1109/TNNLS.2020.2980559).
- [5] Wikipedia contributors. *Gabriel graph* — *Wikipedia, The Free Encyclopedia*. [Online; accessed 12-May-2022]. 2022. URL: [https://en.wikipedia.org/w/index.php?title=Gabriel\\_graph&oldid=1087461346](https://en.wikipedia.org/w/index.php?title=Gabriel_graph&oldid=1087461346).