

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



Aplicação de Redes Neurais em Microcontrolador Embarcado

Revisão Bibliográfica

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

21 de junho de 2022

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 um algoritmo de reconhecimento de padrões para solucionar alguma aplicação prática utilizando um microcontrolador. O algoritmo selecionado é o Grafo de Gabriel[5] e o hardware utilizado será um microcontrolador ESP32. O grafo foi desenvolvido pela comunidade da UFMG e demonstrou boa acurácia e baixo uso de recursos computacionais para separar regiões de dados por método geométrico. A ideia da aplicação a ser implementada é de construir um protótipo de um dispositivo wireless com bateria que possa ser um objeto paupável e faça reconhecimento de padrões de movimento do objeto. Para isso será utilizado um microcontrolador ESP32 e um sensor acelerômetro MPU6050. A aplicação pode ser realizável com a utilização do grafo de Gabriel se mapearmos geometricamente os dados do sensor acelerômetro por meio de uma série temporal da aquisição. Reconhecendo assim o movimento realizado em um certo espaço de tempo pré-definido.

Dentre as soluções já existentes que serviram de inspiração temos o projeto Magic Wand desenvolvido pela comunidade do TensorFlow Lite para exemplificar o uso da biblioteca de Machine Learning para microcontroladores. Há também uma versão do autor andriyadi Andri (github: @andriyadi)¹ que faz uma reformulação do exemplo Magic Wand implementado no framework de desenvolvimento para embarcados que pretendemos utilizar nesse trabalho PlatformIO. Ambos projetos serviram de inspiração para a aplicação escolhida por permitirem reconhecer padrões de movimento por meio de machine learning, porém o método implementado será outro. Enquanto esses projetos utilizam modelos de alto nível o objetivo nesse trabalho é de implementar as equações do algoritmo do Grafo de Gabriel utilizando bibliotecas de Álgebra Linear em C++. Na Plataforma CRAN[2] há a implementação do grafo desenvolvido pela equipe do laboratório de inteligência computacional da UFMG para sua utilização no programa R[3]. Esse algoritmo servirá como inspiração para a implementação do mesmo em C++ para embarcados. Já o artigo do autor Luiz Bambirra[4] demonstra e exemplifica 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. A proposta desse trabalho é de realizar esse algoritmo no microcontrolador embarcado.

O artigo dos autores Arias-Garcia[1] tem uma abordagem similar ao objetivo desse trabalho. Nele foi analisada a desempenho de um classificador utilizando Grafo de Gabriel implementado em Hardware Embarcado. A aplicação foi

¹[url {https://github.com/andriyadi/MagicWand-TFLite-Arduino}](https://github.com/andriyadi/MagicWand-TFLite-Arduino).

implementada em um FPGA², onde se é capaz de reorganizar blocos lógicos de processamento de forma a obter um Hardware dedicado para aquela tarefa. Como a implementação de Redes Neurais requer elevado cálculo matricial e paralelo, a implementação em Hardware dedicado para tal modelo apresenta bom desempenho quando se comparado à implementação em software de alto nível. Esse artigo se difere com projeto desejado no que se refere 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 ARM³ 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 de baixo custo.

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] Jan Philipp Dietrich e Waldir Leoncio. *citation: Software Citation Tools*. R package version 0.6.2. 2022.
- [3] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2021. URL: <https://www.R-project.org/>.
- [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.

²FPGA: Field-programmable gate array..

³ARM is a family of reduced instruction set computer (RISC) instruction set architectures for computer pro...