

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



Aplicação de Reconhecimento de Padrões em Microcontrolador Embarcado

Relatório 3 de Atividades PFC1

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

18 de julho de 2022

Sumário

1	Resumo	2
2	espGG	2
3	Etapa Atual	3
4	Próximos Passos	3

1 Resumo

Esse relatório tem como objetivo expor as pesquisas e atividades realizadas na terceira etapa do PFC1.

O grafo de Gabriel[4] é um método de reconhecimento de padrões que separa regiões de dados de forma geométrica. O grafo foi desenvolvido pela comunidade da UFMG e demonstrou boa acurácia e baixo uso de recursos computacionais. Por esse motivo foi escolhido como modelo de reconhecimento de padrões a ser utilizado.

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.

Essa 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 espaço de tempo pré-definido.

2 espGG

No último relatório deste projeto foi encontrado o código fonte da implementação do Grafo de Gabriel utilizado na biblioteca GGClassification¹ disponível na plataforma CRAN[1] para a linguagem R[2].

A biblioteca foi implementada na linguagem C++ e utiliza uma interface com a linguagem R por meio das bibliotecas Rcpp². O código fonte da biblioteca pode ser encontrado no git: <https://github.com/cran/GGClassification>

Dessa maneira, foi utilizado o código GGClassification como base para o desenvolvimento do projeto espGG, sendo essa a implementação do grafo de Gabriel em um microcontrolador esp32. Foi também implementado um algoritmo para cálculo de acurácia da predição com os dados de teste.

Utilizando os dados de um dos exemplos disponível na documentação do GGClassification vemos na seguinte imagem os resultados da saída serial do microcontrolador. Na Figura 1 vemos primeiramente valores finais (cauda) da matrix xTrain. Vemos também os valores de saída yTrain. Os valores de xTest e yTest foram omitidos. Em seguida temos avisos de que a função de treinamento (Model) e a função de predição (Predict) foram executadas com sucesso. Logo abaixo vemos o resultado da predição e o valor de acurácia comparado ao resultado do mesmo algoritmo (GGClassification) em R.

¹<https://CRAN.R-project.org/package=GGClassification>.

²<https://github.com/RcppCore/Rcpp>.

```

2.18 2.72
2.23 2.72
3.63 3.48
2.52 2.15
3.56 3.73
]
y_train = [
0 1 1 0 0 1 0 1 1 0 1 0
]
Model Done!
Predict Done!
Prediction = [
1 0 0 0 0 1 0 1 0 1 0 0
]
Acurácia: 100.00%

```

Figura 1: Resultados do teste realizado.

3 Etapa Atual

No momento, com o auxílio do professor orientador estudei conteúdos relacionado ao uso de dados de acelerômetro em reconhecimento de padrões. Dentre esses materiais temos a base de dados de um acelerômetro[3] no repositório da UCI para prever sobre vibração de máquinas elétricas, o qual estou tentando implementar no espGG.

Um ponto de atenção é com relação à memória e processamento limitados do microcontrolador. Por esse motivo uma etapa importante do trabalho será a de realizar o pré-processamento dos dados de forma a otimizar o algoritmo e reduzir a necessidade desses recursos.

4 Próximos Passos

Os próximos passos será criar maior familiaridade com a utilização do algoritmo no esp e saber com mais exatidão os limites do microcontrolador. Além disso implementar exemplos de dados de acelerômetro.

Foram adquiridos dois sensores MPU (Acelerômetro + Giroscópio) de modelos diferentes para avaliar qual será melhor para a aplicação.

Referências

- [1] Jan Philipp Dietrich e Waldir Leoncio. *citation: Software Citation Tools*. R package version 0.6.2. 2022.
- [2] 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/>.
- [3] Gustavo Scalabrini Sampaio et al. “Prediction of Motor Failure Time Using An Artificial Neural Network”. Em: *Sensors* 19.19 (out. de 2019), p. 4342. DOI: [10.3390/s19194342](https://doi.org/10.3390/s19194342). URL: [\[Web%20Link\]](#).
- [4] 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.