

# Classificação de Fluxo Óptico por Rede Neural e Lógica Fuzzy

Breno Alves Guimarães

Departamento de Engenharia Elétrica  
Centro Federal de Educação Tecnológica  
de Minas Gerais  
Belo Horizonte, Brasil  
Email: brenoaeronave@gmail.com

Vinícius Osvaldo Miranda Alves

Departamento de Engenharia Elétrica  
Centro Federal de Educação Tecnológica  
de Minas Gerais  
Belo Horizonte, Brasil  
Email: vinioswaldo92@gmail.com

**Resumo**—Este trabalho tem o objetivo de criar um algoritmo em Matlab para classificar os movimentos de rotação de uma câmera a partir de um vídeo. Isso será feito por meio de ferramentas de Fluxo Óptico e Inteligência Artificial, a fim de, posteriormente, utilizar este sistema como controle de estabilidade de um drone construído por alunos do CEFET-MG.

**Palavras-chave**—Fluxo óptico, inteligência artificial, Rede Neural, Fuzzy, odometria visual, visão computacional.

## I. Introdução

A Odometria Visual (*Visual Odometry*) é o processo de estimar a posição relativa de uma câmera a partir de uma sequência de imagens [1]. Ela pode ser alcançada em conjunto de uma ou mais técnicas como (1) ferramentas de visão computacional (2) deep learning e inteligências artificiais, (3) fusão com sensores de medição inercial (Visual-Inercial Odometry), GPS, entre outros. O termo foi cunhado por David Nister em 2004 [2] e ainda é explorado em vários campos da robótica, carros autônomos, drones, entre outros. Neste trabalho, será feito um processo simples de classificação do movimento angular de uma câmera por meio do cálculo do Fluxo Óptico de vídeos e classificar por meio de duas técnicas de Inteligência Artificial: Rede Neural e Lógica Fuzzy.

## II. Objetivos

### A. Cálculo do Fluxo Óptico dos objetos de uma tela

Neste trabalho, será explorada a tarefa de estimar o sentido de rotação de uma câmera por meio do comportamento dos pixels da imagem. Ao movimentar a câmera em uma direção qualquer, por exemplo, um objeto adquire um movimento aparente na direção oposta, sendo possível determinar o movimento da câmera.

### B. Classificação do movimento

Após analisar o movimento dos pixels, será realizada a classificação do movimento nas seguintes direções: alto, baixo, direita e esquerda utilizando uma rede neural simples e um sistema Fuzzy e comparar os resultados.

## III. Descrição

### A. Fluxo Óptico

O fluxo óptico de um objeto é um conceito de visão computacional que calcula o movimento aparente de objetos, superfícies e paredes em um cenário visual [3].

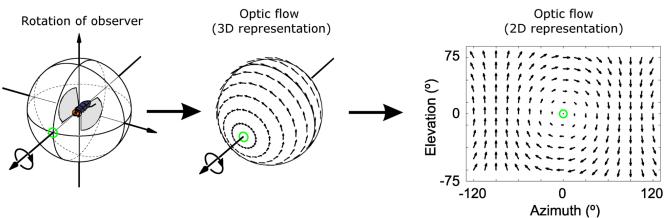


Fig. 1. Representação do fluxo óptico

### B. Rede Neural MLP para classificação

Uma rede Perceptron Multicamadas (MLP) é uma rede neural que pode ser utilizada de forma eficaz para realizar tarefas de classificação. Neste trabalho será usada como entradas da rede os vetores horizontal  $u$  e  $v$  e, como saídas, quatro classificações (alto, baixo, direita, esquerda).

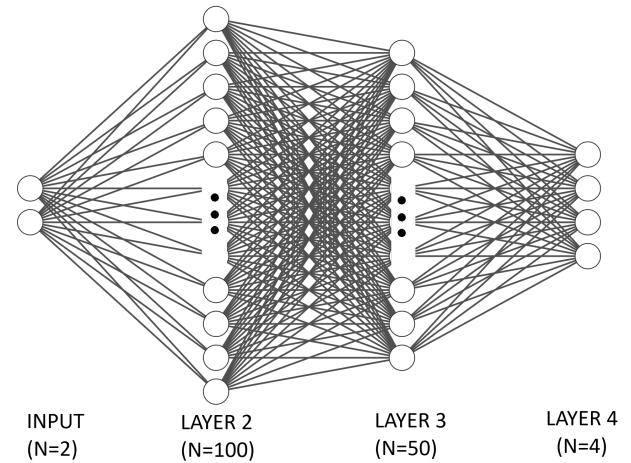


Fig. 2. Diagrama detalhando a Rede MLP utilizada

## IV. Metodologia

Para a análise do movimento foi utilizado um dataset com vídeos da câmera se movendo nas quatro direções. Três scripts foram utilizados: um para a visão de ângulo, um para criação da rede neural e outro para a lógica Fuzzy.

### A. Visão de ângulo

Neste script foi feito o cálculo do fluxo óptico dos pixels do vídeo. Com isso, são obtidos os vetores velocidade em

pixels/frame. Após isso, é feita uma média dos vetores para cada vídeo, obtendo assim, 96 valores, que serão incorporados no DataSet. Foram utilizadas as seguintes etapas para o script:

- 1) Loop while para calcular a cada frame do vídeo
  - 2) Cálculo do Fluxo Óptico
- Para o cálculo e exibição na tela foram utilizadas as seguintes funções:
- a) `opticalFlowLK()`: estimativa da direção e sentido de um objeto utilizando o método de Lucas-Kanade [4].
  - b) `estimateFlow()`: estimativa do fluxo óptico entre dois frames. Recebe como entradas os vetores velocidade de componente horizontal e vertical em pixels/frame.
  - c) `mean()`: calcula a média dos vetores obtidos
  - d) `classify()`: realiza a classificação da rede com base nas entradas dos vetores  $\bar{u}$  e  $\bar{v}$ .
- 3) Plota os resultados na tela.

#### B. Rede Neural

Para a realização da Rede Neural foram utilizados os seguintes parâmetros:

- 1) Banco de dados

Contém vídeos com desenhos de duas figuras na tela (triângulo e círculo). Para cada movimento foram gravados 24 vídeos de 2s a 4s, totalizando 96 vídeos.



Fig. 3. Objetos usados no dataset

- 2) Hiperparâmetros

Foram utilizadas 4 camadas de rede definidas da seguinte maneira:

- . Entrada: 2 neurônios
  - . Camada oculta 1: 100 neurônios
  - . Camada oculta 2: 50 neurônios
  - . Saída: 4 neurônios
- 3) Funções de ativação: ReLU (camadas ocultas) e SoftMax (saída).

TABELA I.

Rede MLP		
Layer 1	Entrada	2 neurônios
Layer 2	Camada Oculta	100 neurônios
Layer 3	Camada Oculta	50 neurônios
Layer 4	Saída	4 neurônios

#### C. Sistema Fuzzy

Foi utilizado também um Sistema Fuzzy com os seguintes parâmetros para a classificação:

- . Funções de pertinência pseudotrapezoidais (`trimf`)
- . Variáveis de entrada: vetores  $u$  e  $v$  (pixels/frame)
- . Variáveis de saída:  $x \in [1,4]$  para as quatro direções
- . Valores das funções de pertinência: -1,-0.2,0,0.2,1 (fig. 4)

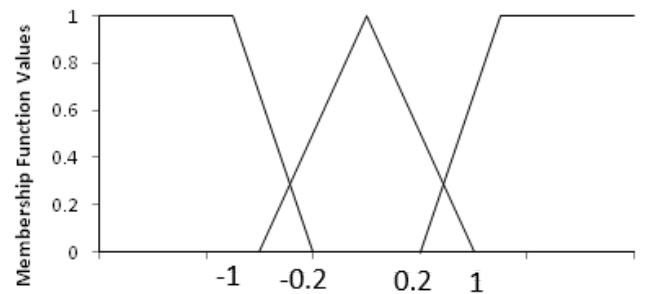


Fig. 4. Regra de definição das funções de pertinência

Além disso, foi aplicado um filtro de Kalman para redução do ruído nos resultados.

#### V. Resultados e Análises

Para validar os resultados dos dois métodos, foi utilizado um vídeo de 34s com sequências de movimentos contendo as quatro direções possíveis.

Comparando-se as duas técnicas, pode-se concluir que a Lógica Fuzzy apresentou um melhor desempenho na classificação. Além disso, ela possui a vantagem de não necessitar de um banco de dados como a rede neural. As figuras a seguir mostram os gráficos das velocidades na componente horizontal (em azul) e vertical (em vermelho) ao longo dos frames do vídeo.

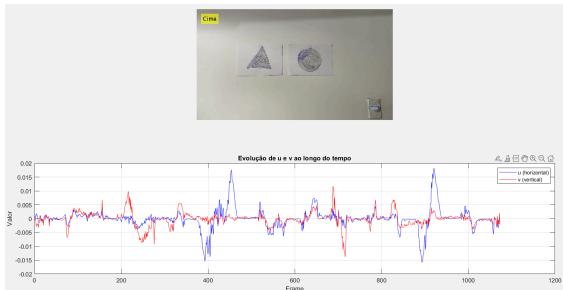


Fig. 5. Resultados do fluxo pela rede neural

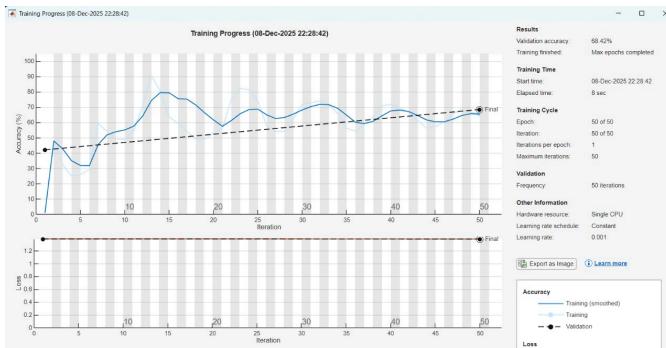


Fig. 6. Resultado do treinamento da rede neural

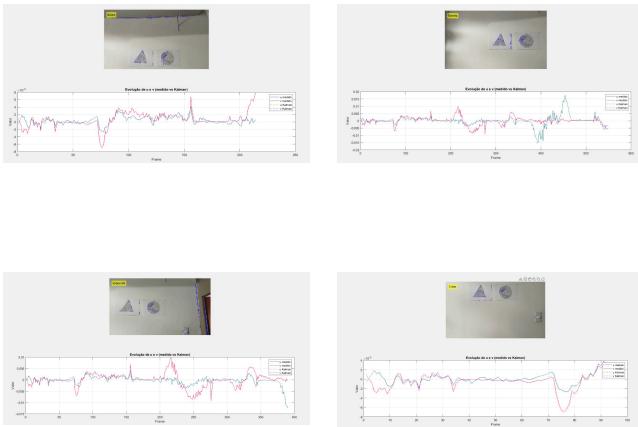


Fig. 7. Resultados pela Lógica Fuzzy

Apesar disso, algumas falhas foram encontradas principalmente devido à tremura no vídeo, o que causou a aparição de velocidades consideráveis nos momentos em que a câmera estava parada, principalmente.

## References

- [1] MathWorks. Monocular Visual Odometry. Disponível em: <https://www.mathworks.com/help/vision/ug/monocular-visual-odometry.html>
- [2] D. Scaramuzza and F. Fraundorfer. "Visual Odometry. Part I: The First 30 Years and Fundamentals". IEEE Xplore Full-Text PDF: Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6096039>
- [3] Ultralytics. Optical Flow: Definition, Algorithms & Applications. Disponível em: <https://www.ultralytics.com/glossary/optical-flow>
- [4] MATLAB. "OpticalFlowLK - Object for estimating optical flow using Lucas-Kanade method". Disponível em: <https://www.mathworks.com/help/vision/ref/opticalflowlk.html>

## VII. Conclusão

A partir dos resultados alcançados, pode-se concluir que o fluxo óptico é um método possível para classificar o movimento de rotação de uma câmera. Apesar disso, esta técnica pode ser refinada para determinação da posição da câmera com maior precisão e, assim, ser utilizado como ferramenta de odometria visual em equipamentos como drones.