

# Reconhecimento de placas de velocidade de trânsito baseado no método de Transformada Rápida de Fourier (FFT)

1<sup>st</sup> Enrique Laborão Monteiro  
*Universidade Federal De Pernambuco*  
Recife, Brasil  
elm2@cin.ufpe.br

2<sup>nd</sup> Hugo Alves Cardoso  
*Universidade Federal De Pernambuco*  
Recife, Brasil  
@cin.ufpe.br

3<sup>rd</sup> Lucca Morosini Gioia  
*Universidade Federal De Pernambuco*  
Recife, Brasil  
lmg2@cin.ufpe.br

4<sup>th</sup> Williams Santiago de Souza Filho  
*Universidade Federal De Pernambuco*  
Recife, Brasil  
wssf@cin.ufpe.br

## DESAFIOS

Inicialmente, propomos uma solução para o problema de reconhecimento de impressões digitais. Porém, à medida que tentávamos nos organizar para avançar na nossa solução, encontramos algumas barreiras que nos fizeram mudar a escolha do tema. O principal impecilho está relacionado ao fato de que não conseguimos encontrar datasets adequados ao que pretendíamos fazer. Para fazer o reconhecimento, seria necessário um grande número de imagens.

Além disso, percebemos que o impacto da aplicação da Transformada de Fourier nos dataset que encontramos era mínimo, o que acabava maquiando o propósito fundamental do projeto.

Dessa forma, decidimos que alterar o objetivo do nosso projeto para detecção de placas de velocidade de trânsito foi o melhor a se fazer, visto que é um tema também de grande relevância e que nos permitiu trabalhar, nos limites do escopo do projeto, com o que é de fato requerido.

## I. INTRODUÇÃO

O objetivo do nosso projeto é fazer um algoritmo de reconhecimento de placas de transito utilizando a transformada de fourier. Nossa abordagem utiliza de métodos estatísticos para pré-processar a imagem, a transformada rápida de fourier para extrair as

features e as bibliotecas OpenCV e EasyOCR para acharmos os contornos de interesse.

## II. MOTIVAÇÃO

A quantidade de carros que vêm sendo produzidos com ferramentas para automatizar a direção está crescendo rapidamente nos últimos anos, com a Tesla sendo o principal nome. Porém, até fabricantes como BMW e AUDI vêm tentando adicionar métodos novos de melhorar o conforto e auxiliar o motorista, como uma tecnologia que mantém o carro na faixa correta ou que regula a velocidade baseado na distância ao carro da frente. Dessa forma, outra ferramenta de grande importância seria a capacidade de detectar e decifrar placas de velocidade para automaticamente regular a velocidade do carro de acordo com a via. Assim, a proposta é criar uma ferramenta que consiga detectar tais placas usando a Transformada de Fourier.

## III. METODOLOGIA

Nosso método utiliza de funções específicas e um filtro estatístico para aprimorarmos os contornos e eliminarmos ruídos da imagem. Depois, utilizamos da transformada rápida de fourier (FFT) para obter contornos mais nítidos e segmentarmos a imagem. O próximo passo é aplicar a biblioteca OpenCV

para detectarmos os contornos existentes na imagem. Por último, a biblioteca EasyOCR nos auxilia a encontrar textos nas áreas destacadas por contornos, podemos então filtrar as áreas que possuem textos numéricos.

#### A. Pré-processamento

O primeiro passo no pré processamento é escalonarmos todas as imagens para o mesmo formato de 512 pixels de largura. Também convertemos a figura para escala cinza (grayscale) para simplificar as operações com cores, tendo agora, apenas um valor de intensidade para cada pixel.



Fig. 1: Imagem original



Fig. 2: Imagem em escala cinza

Em seguida, aplicamos um filtro anti-glimmer para remover reflexos e brilhos intensos que possam ser interpretados como ruídos e impedir futuramente a identificação de contornos.

Para finalizarmos o pré-processamento, utilizamos um filtro estatístico para destaque das



Fig. 3: Imagem pré-processada

bordas da imagem em questão, tornando-as mais nítidas. Esse filtro escolhido é fruto de uma combinação linear do delta de Dirac ( $\delta$ ) com um filtro passa-alta. Definimos um kernel para tal:

$$\text{kernel} = \frac{1}{10} \times \begin{pmatrix} -1 & -1 & -1 \\ -1 & 18 & -1 \\ -1 & -1 & -1 \end{pmatrix}$$

Esse kernel será, finalmente convolvido com a imagem para que tenhamos nossa imagem enfim pré-processada como output.

#### B. Transformada de Fourier para obtenção do contorno

Com nossa imagem pré-processada, podemos aplicar o método da transformada de fourier. Nessa aplicação escolhemos a transformada rápida de fourier por possuir amplo uso e melhor desempenho comparado com outros métodos, enquanto a Transformada de Fourier tem complexidade  $O(N^2)$ , a rápida tem complexidade  $O(n \log n)$ . A transformação retorna uma representação da nossa imagem no domínio da frequência, onde podemos aplicar diferentes operações e posteriormente desfazer a transformação através da transformada inversa de fourier (IFFT).

No domínio da frequência, aplicamos um filtro passa alta (HPF) para remover regiões semelhantes entre si, já que são representadas pelas baixas frequências. Nosso HPF, portanto, destaca os contornos. Então, podemos aplicar a transformada inversa para obtermos nossa imagem com os contornos melhor definidos.

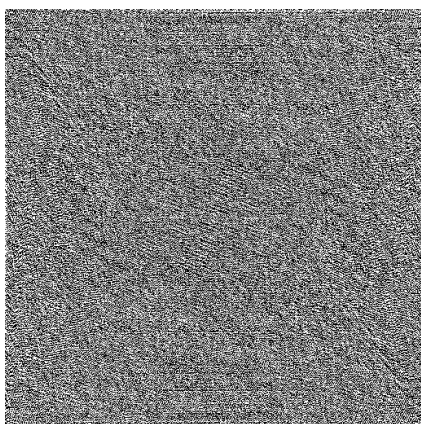


Fig. 4: Representação da imagem no domínio da frequência



Fig. 6: Imagem retornada após transformação e filtragem

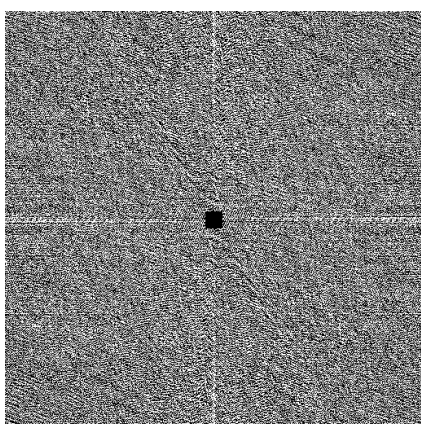


Fig. 5: Domínio da frequência filtrado

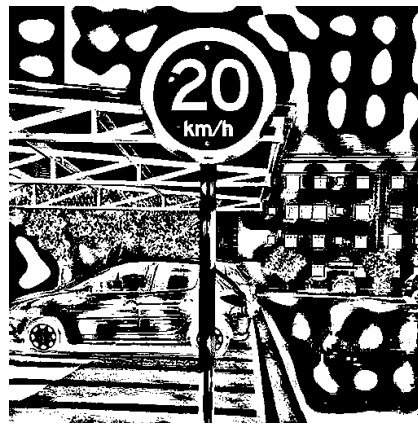


Fig. 7: Imagem normalizada

### C. Detecção dos contornos

Como é possível ver, a imagem está extremamente escura porque, durante a transformada inversa, alguns valores se tornam negativos. Precisamos normalizar esses valores para o intervalo  $[0, 255]$  e posteriormente aplicar uma aproximação para que se tornem valores binários 0 ou 255.

Agora utilizamos da biblioteca OpenCV para acharmos contornos dentro da imagem, mais especificamente a função **findContours**. O problema que agora temos é que na imagem existem diversos contornos, precisamos, portanto, filtrar as regiões que possuem maior chance de incluírem uma placa.

Para isso utilizamos a biblioteca EasyOCR para detectarmos números na imagem, removendo, portanto, áreas que não possuem nenhum número. Também filtramos por dimensão, já que as placas de sinalização possuem um formato circular, configu-

rando uma dimensão aproximada de 1:1, embora tenhamos relaxado levemente a proporção para não termos muitos casos de falso-negativo.

## IV. RESULTADOS

Para testarmos nosso modelo, utilizamos um dataset colhido manualmente do Google Imagens, onde temos placas de velocidade em diferentes condições. Nosso modelo teve bons resultados, achando regiões de interesse em 77.5% das imagens (em 31 das 40 totais) e tendo boa adesão às placas.

O maior problema para nosso modelo são imagens de baixa resolução, onde as bibliotecas não conseguem identificar os contornos das placas.

De maneira geral, se baseando na nossa motivação principal para essa abordagem, podemos concluir que a utilização de FFT em imagens pré-processadas para identificação de placas sinalizadoras de velocidade no trânsito é uma ótima



Fig. 8: Imagem original com o contorno alvo

abordagem, desde que se tenha um dataset com imagens de alta resolução. No que diz respeito aos veículos autônomos, é um investimento que pode vir aumentar, futuramente, a segurança das pessoas nos transportes rodoviários.



Fig. 9: Imagem sem contornos identificados

## REFERENCES

- [1] <https://towardsdatascience.com/image-processing-with-python-application-of-fourier-transformation-5a8584dc175b>
- [2] Varsha Nair. Improving Objects Recognition through Fast Fourier Transformation in Neural Network, Texas, 2019.
- [3] <https://datacarpentry.org/image-processing/06-blurring/>
- [4] <https://homepages.inf.ed.ac.uk/rbf/HIPR2/fourier.htm>: :text=The%20Fourier%20Transform%20is%20an,is%20the%20spatial%20domain%20equivalent.