

Universidade Federal do Rio Grande do Norte

Programa de Pós-graduação em Engenharia Elétrica e de Computação
EEC1515 - VISÃO COMPUTACIONAL

RELATÓRIO

IMPLEMENTAÇÃO DE UM DETECTOR DE QUADRADOS BASEADO NA TRANSFORMADA HOUGH

Autor: Luís Gabriel Pereira Condados

Professor orientador: Rafael Beserra Gomes

Natal-RN

2020

Universidade Federal do Rio Grande do Norte

Programa de Pós-graduação em Engenharia Elétrica e de Computação
EEC1515 - VISÃO COMPUTACIONAL

RELATÓRIO

Relatório apresentado à disciplina de EEC1515- Visão Computacional, correspondente a 1º unidade do semestre 2020.2, sob orientação do **Prof. Rafael Beserra Gomes**.

Autor:Luís Gabriel Pereira Condados

Sumário

Sumário	3
Lista de ilustrações	4
1 INTRODUÇÃO	1
2 METODOLOGIA	1
3 RESULTADOS	3
4 CONCLUSÃO	3
REFERÊNCIAS	3
5 ANEXO 1	3

Lista de ilustrações

Figura 1 – Parametrização do quadrado em uma imagem $f(x, y)$	1
Figura 2 – Operadores de <i>Sobel</i>	2
Figura 3 – Cálculo do módulo do gradiente da imagem $f(x, y)$	2
Figura 4 – Identificação dos parâmetros do candidato à quadrado.	2

1 INTRODUÇÃO

A transformada de *Hough* é um método comum para detecção de formas que são facilmente parametrizáveis (formas comuns: linhas e círculos). Geralmente essa transformada é utilizada após a etapa de pré-processamento da imagem, principalmente após detecção de bordas.

O método consiste em mapear determinados pixels da imagem numa determinada célula no espaço de parâmetros que definem a figura geométrica. Esse procedimento é repetido para toda a imagem e as células do espaço de parâmetros são incrementadas, servindo assim de indicadores da existência de uma determinada forma.

Neste trabalho será apresentado uma implementação da transformada *Hough* para a identificação de quadrados pretos em imagens com fundo branco baseado no gradiente. Para isso foi feito a parametrização de um quadrado com a informação do centro do quadrado, tamanho do lado e orientação com relação ao eixo horizontal (x_c, y_c, l, θ) , por isso o espaço de configuração terá dimensão 4. Para fazer o mapeamento entre o espaço de imagem e o de parâmetros foi utilizado a informação do gradiente da imagem e um ponto de borda, para com isso obter-se as normais do quadrado e com isso estimar os quatro parâmetros. Após o mapeamento/acumulo dos indicadores é feito uma etapa de filtragem dos quadrados, para melhorar a precisão da detecção.

2 METODOLOGIA

Para a transformada de *Hough* aplicada à quadrados será utilizado a parametrização do quadrado em: coordenadas centrais (x_c, y_c) ; orientação com relação ao eixo horizontal (θ) e o comprimento de seu lado (l) em pixel. A Figura 1 ilustra essa parametrização.

A proposta do trabalho é realizar o mapeamento do espaço de imagem para o espaço de parâmetros fazendo uso do gradiente da imagem.

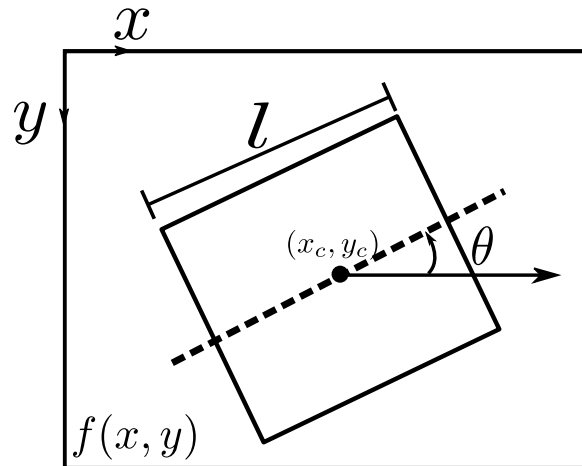


Figura 1 – Parametrização do quadrado em uma imagem $f(x, y)$.

$$H_{Sobel_x} = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

(a) Operador de *Sobel* na direção x .

$$H_{Sobel_y} = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

(b) Operador de *Sobel* na direção y .

Figura 2 – Operadores de *Sobel*.

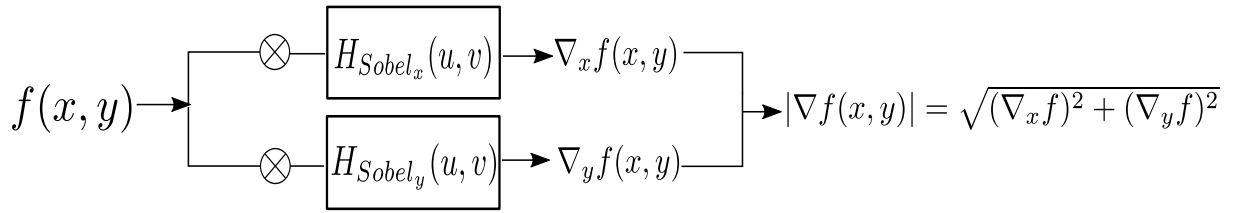


Figura 3 – Cálculo do módulo do gradiente da imagem $f(x, y)$.

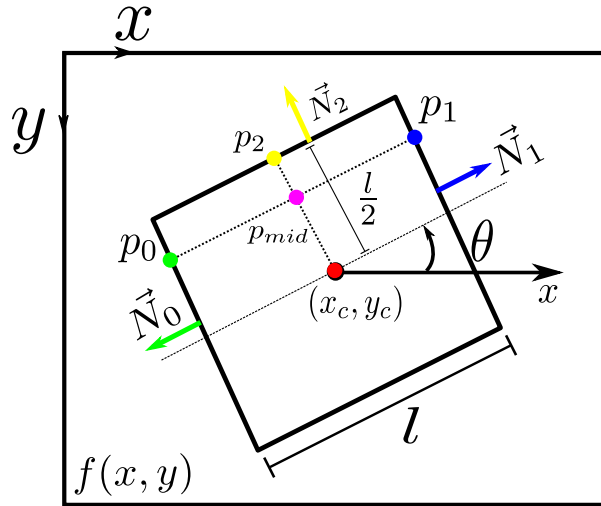


Figura 4 – Identificação dos parâmetros do candidato à quadrado.

Algorithm 1 A Hough Transform for Squares Detection

```
1: initialize  $M$  with zero. ▷ Voting matrix initialized with zero.
2: for all  $(x', y')$  in  $f(x, y)$  do
3:   if  $|\nabla f(x', y')| \geq \text{edge threshold}$  then ▷ It's an edge.
4:      $p_0 \leftarrow (x', y')$ 
5:      $\theta \leftarrow \angle \nabla f(x', y') \bmod 90^\circ$  ▷ Square's angle.
6:      $\vec{N}_0 \leftarrow \nabla f(x', y') / |\nabla f(x', y')|$ 
7:      $\vec{N}_1 \leftarrow -\nabla f(x', y') / |\nabla f(x', y')|$ 
8:      $\vec{N}_2 \leftarrow \text{rotate } \vec{N}_1 \text{ on } 90^\circ$ 
9:
10:    Search for an edge point going in the direction of  $\vec{N}_1$  from  $p_0$ . ▷ That will be
    the  $p_1$  point.
11:
12:     $l \leftarrow |p_0 - p_1|$  ▷ Square's size.
13:     $p_{middle} \leftarrow (p_1 + p_0)/2$ 
14:
15:    Search for an edge point going in the direction of  $\vec{N}_2$  from  $p_{middle}$ . ▷ That will
    be the  $p_2$  point.
16:
17:     $p_{center} \leftarrow p_2 - \vec{N}_2 * l/2$  ▷  $p_{center} = (x_c, y_c)$ 
18:     $M[x_c][y_c][l][\theta] \leftarrow M[x_c][y_c][l][\theta] + 1$ 
19:  end if
20: end for
```

3 RESULTADOS

4 CONCLUSÃO

Referências

5 ANEXO 1

Código fonte: <https://github.com/Gabriellgpc/Sistemas_Roboticos/tree/master/program/p2m1>