

Laboratory  
of Critical  
Embedded  
Systems

# Estimativa de Posição 3D de Obstáculos para VANTs

Rodolfo Barros Chiaramonte, Kalinka Regina Lucas Jaquie  
Castelo Branco

Institute of Mathematics and Computing Sciences (ICMC)  
University of São Paulo (USP), São Carlos, São Paulo, Brazil  
[rodolfobch@usp.br](mailto:rodolfobch@usp.br), [kalinka@icmc.usp.br](mailto:kalinka@icmc.usp.br)

# Conteúdo

- Introdução
- Método de Cálculo
- Diferenças com o Estado da Arte
- Simulação
- Resultados
- Conclusão

# Veículos Aéreos Não Tripulados

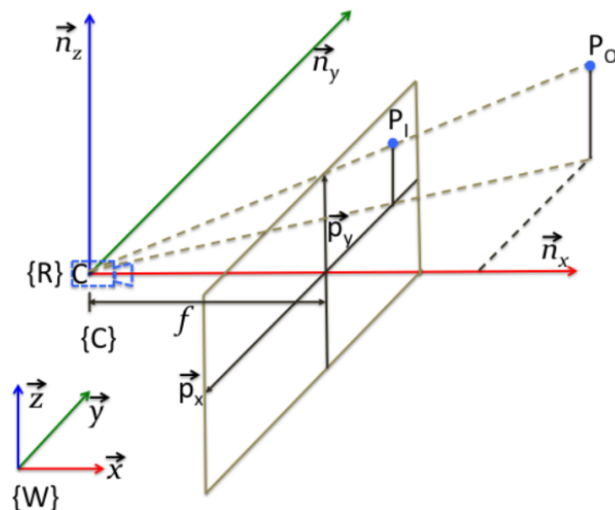
Laboratory  
of Critical  
Embedded  
Systems

- Permitem a execução de missões;
- Necessidade de navegação autônoma:
  - Detecção de Colisões
  - Necessidade de conhecer a posição de possíveis obstáculos;
  - Restrições de *payload* quanto ao uso de câmera estéreo:
    - **Necessidade de utilizar uma única câmera para cálculo da posição de obstáculos.**

# Método de Cálculo de Posição

- Um ponto 3D é projetado em uma imagem 2D por meio da formula de projeção:

$$P_I(x_I, y_I) = \frac{f}{\vec{n}_x \cdot \vec{CP}_o} (-\vec{n}_y \cdot \vec{CP}_o, \vec{n}_z \cdot \vec{CP}_o)$$



[Saha et al. 2014]

Onde:

$(x_I, y_I) \rightarrow$  posição do ponto na imagem;

$f \rightarrow$  distância focal da câmera (em *pixels*);

$\vec{n}_x, \vec{n}_y, \vec{n}_z \rightarrow$  vetores de orientação da câmera;

$\vec{CP}_o \rightarrow$  Vetor entre Câmera e o Obstáculo;

# Método de Cálculo de Posição

- Como uma coordenada 3D é projetada em uma coordenada 2D ocorre a perda de uma dimensão.
- Para recuperar esta informação gerando novamente a posição do obstáculo são necessárias as informações:
  - Características da câmera;
  - Posicionamento da câmera;
  - Orientação da câmera;
  - Posição do obstáculo na imagem.

# Método de Cálculo de Posição

- Com base nas informações descritas, é possível elaborar um sistema de equações para recuperar as coordenadas  $(x,y,z)$  de um obstáculo;
- Para que seja possível resolver o sistema, é necessário utilizar dados obtidos em dois instantes de tempo distintos.

# Diferença com o Estado da Arte

- O método proposto por [Saha et al. 2014] considera que:

$$\vec{n}_z = \vec{z}$$

- Ou seja, que o vetor de orientação vertical da câmera sempre coincide com o vetor vertical do mundo real;
- Com isto, a única movimentação considerada por [Saha et al. 2014] é em relação ao ângulo de *yaw*.

# Diferença com o Estado da Arte

- No presente trabalho, o sistema de equações foi reelaborado, considerando as matrizes de rotação  $R_z$  e  $R_y$ ;
- Com isto, é possível considerar movimentação da câmera em torno dos ângulos de *yaw* e *pitch*.



# Sistema de Equações Proposto

Considerando:

$$a = \cos(\text{yaw})$$

$$b = \sin(\text{yaw})$$

$$c = \cos(\text{pitch})$$

$$d = \sin(\text{pitch})$$

$$e_1 = -x_{I_1} f_y d_1 + y_{I_1} f_x b_1 c_1$$

$$e_2 = y_{I_1} f_x a_1$$

$$e_3 = x_{I_1} f_y c_1 + y_{I_1} f_x b_1 d_1$$

$$e_4 = -x_{I_2} f_y d_2 + y_{I_2} f_x b_2 c_2$$

$$e_5 = y_{I_2} f_x a_2$$

$$e_6 = x_{I_2} f_y c_2 + y_{I_2} f_x b_2 d_2$$

$$e_7 = y_{I_1} a_1 c_1 - y_{I_2} a_2 c_2 + f_y d_1 - f_y d_2$$

$$e_8 = -y_{I_1} b_1 + y_{I_2} b_2$$

$$e_9 = y_{I_1} a_1 d_1 - y_{I_2} a_2 d_2 - f_y c_1 + f_y c_2$$

$$B_1 = x_{I_1} (-f_y d_1 x_1 + f_y c_1 z_1) + y_{I_1} (f_x b_1 c_1 x_1 + f_x a_1 y_1) + y_{I_1} f_x b_1 d_1 z_1$$

$$B_2 = x_{I_2} (-f_y d_2 x_2 + f_y c_2 z_2) + y_{I_2} (f_x b_2 c_2 x_2 + f_x a_2 y_2) + y_{I_2} f_x b_2 d_2 z_2$$

$$B_3 = y_{I_1} (a_1 c_1 x_1 - b_1 y_1 + a_1 d_1 z_1) + y_{I_2} (-a_2 c_2 x_2 + b_2 y_2 - a_2 d_2 z_2) + f_y (d_1 x_1 - c_1 z_1 - d_2 x_2 + c_2 z_2)$$

Resolve-se:

$$Ax = B$$

onde,

$$A = \begin{bmatrix} e_1 & e_2 & e_3 \\ e_4 & e_5 & e_6 \\ e_7 & e_8 & e_9 \end{bmatrix}, x = \begin{bmatrix} x_O \\ y_O \\ z_O \end{bmatrix}, B = \begin{bmatrix} B_1 \\ B_2 \\ B_3 \end{bmatrix}$$

# Simulação

- Para a simulação, foram considerados valores fixos para:
  - Posição do obstáculo;
  - Características da câmera.
- E gerados valores aleatórios para:
  - Posicionamento da câmera;
  - Orientação da câmera;
- Com base nestas informações foram calculadas as posições dos obstáculos nas imagens;

# Simulação

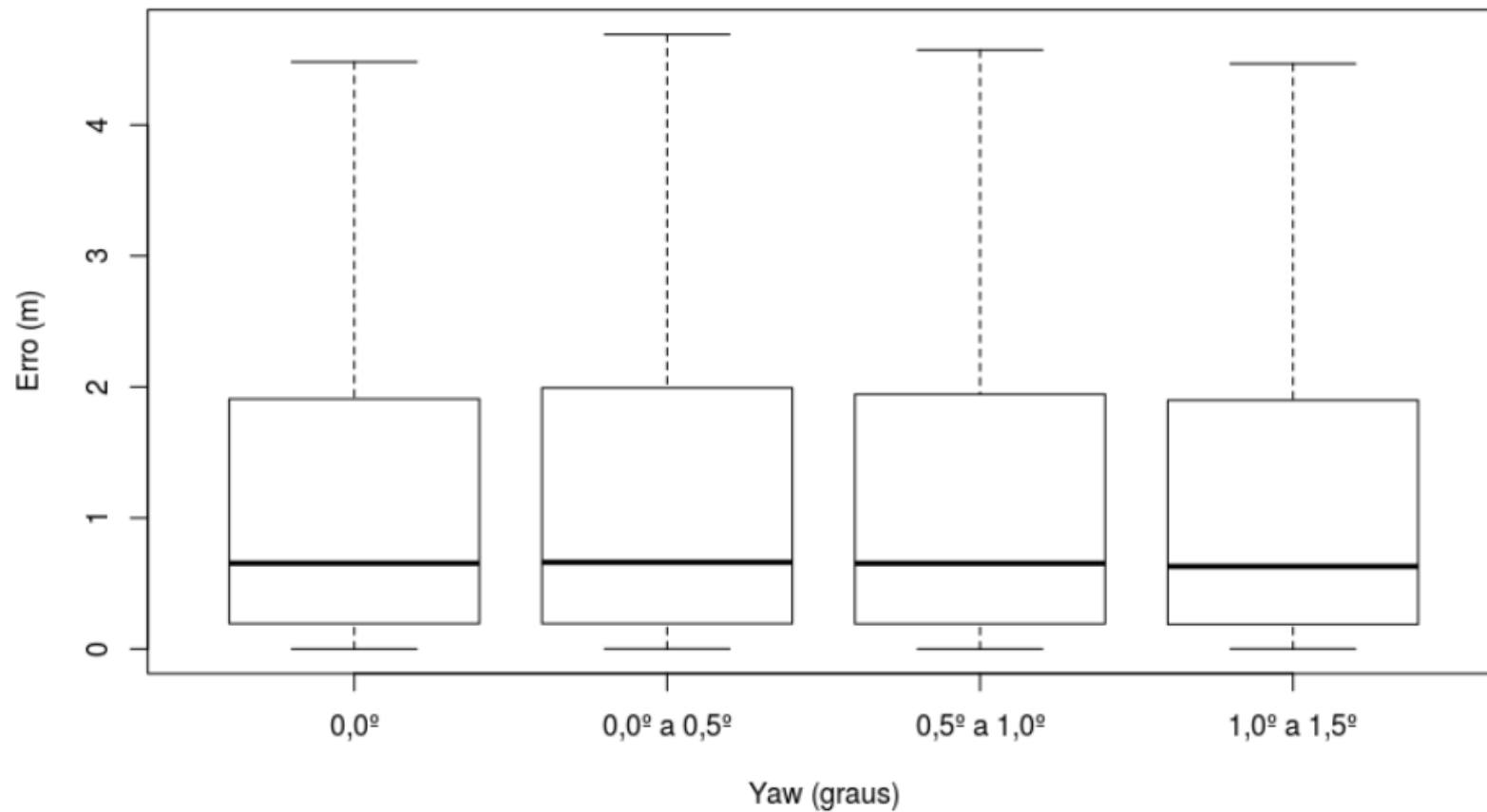
- Critérios para geração dos valores aleatórios:
  - Obstáculo posicionado a:
    - 10m, 20m, 30m, 40m e 50m.
  - Variação da posição do *drone* em:
    - 1cm a 2cm, 2cm a 3cm, ..., 49cm a 50cm.
  - Variação dos ângulos de *yaw* e *pitch*:
- Para cada combinação de critérios foram geradas 100 amostras.

<i>Yaw</i>	<i>Pitch</i>
0°	0°
0° a 0,5°	0°
0,5° a 1,0°	0°
1,0° a 1,5°	0°
0°	0° a 0,5°
0° a 0,5°	0° a 0,5°
0,5° a 1,0°	0° a 0,5°
1,0° a 1,5°	0° a 0,5°
0°	0,5° a 1,0°
0° a 0,5°	0,5° a 1,0°
0,5° a 1,0°	0,5° a 1,0°
1,0° a 1,5°	0,5° a 1,0°
0°	1,0° a 1,5°
0° a 0,5°	1,0° a 1,5°
0,5° a 1,0°	1,0° a 1,5°
1,0° a 1,5°	1,0° a 1,5°



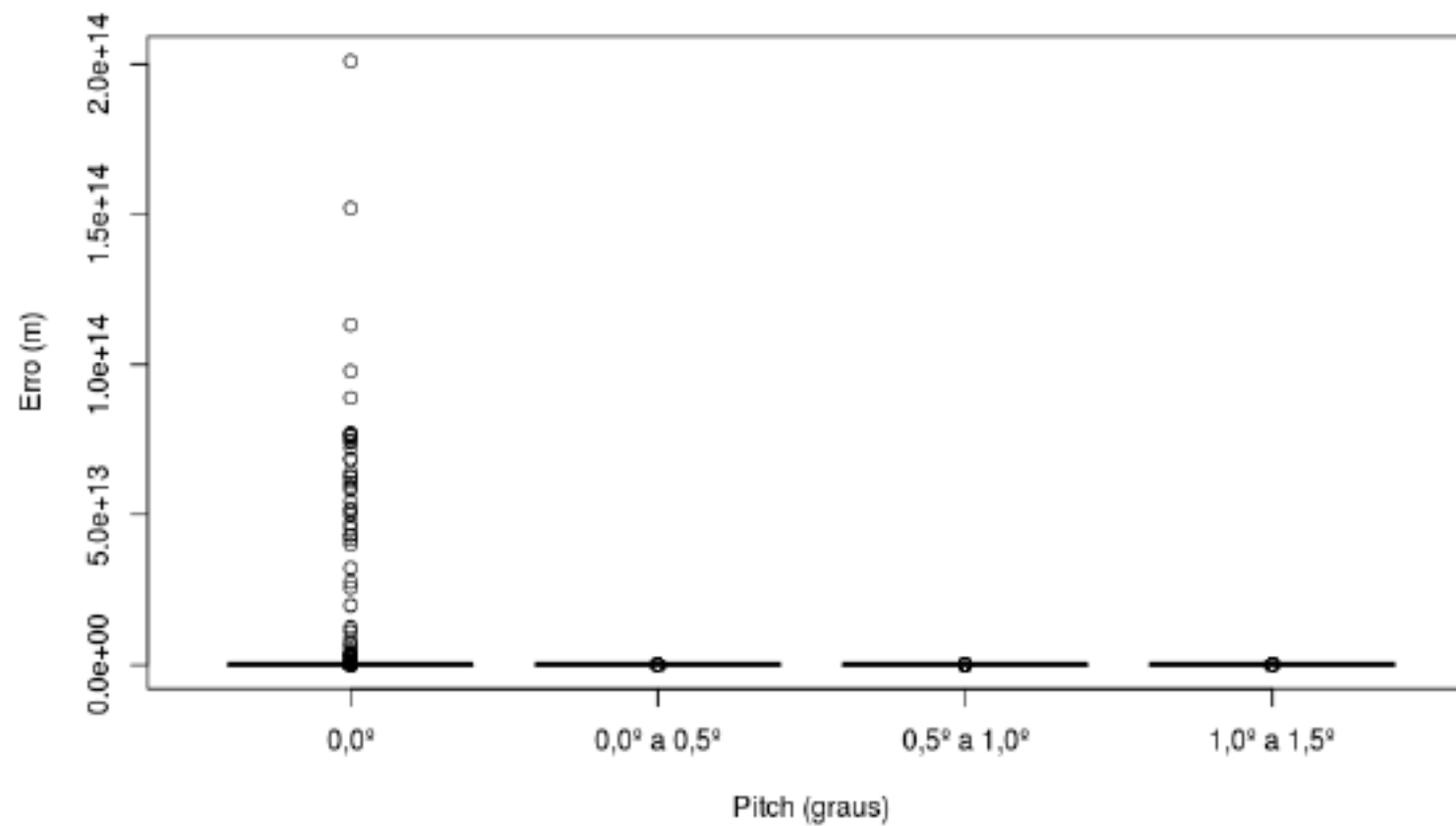
# Resultados

Relação do Erro x Yaw



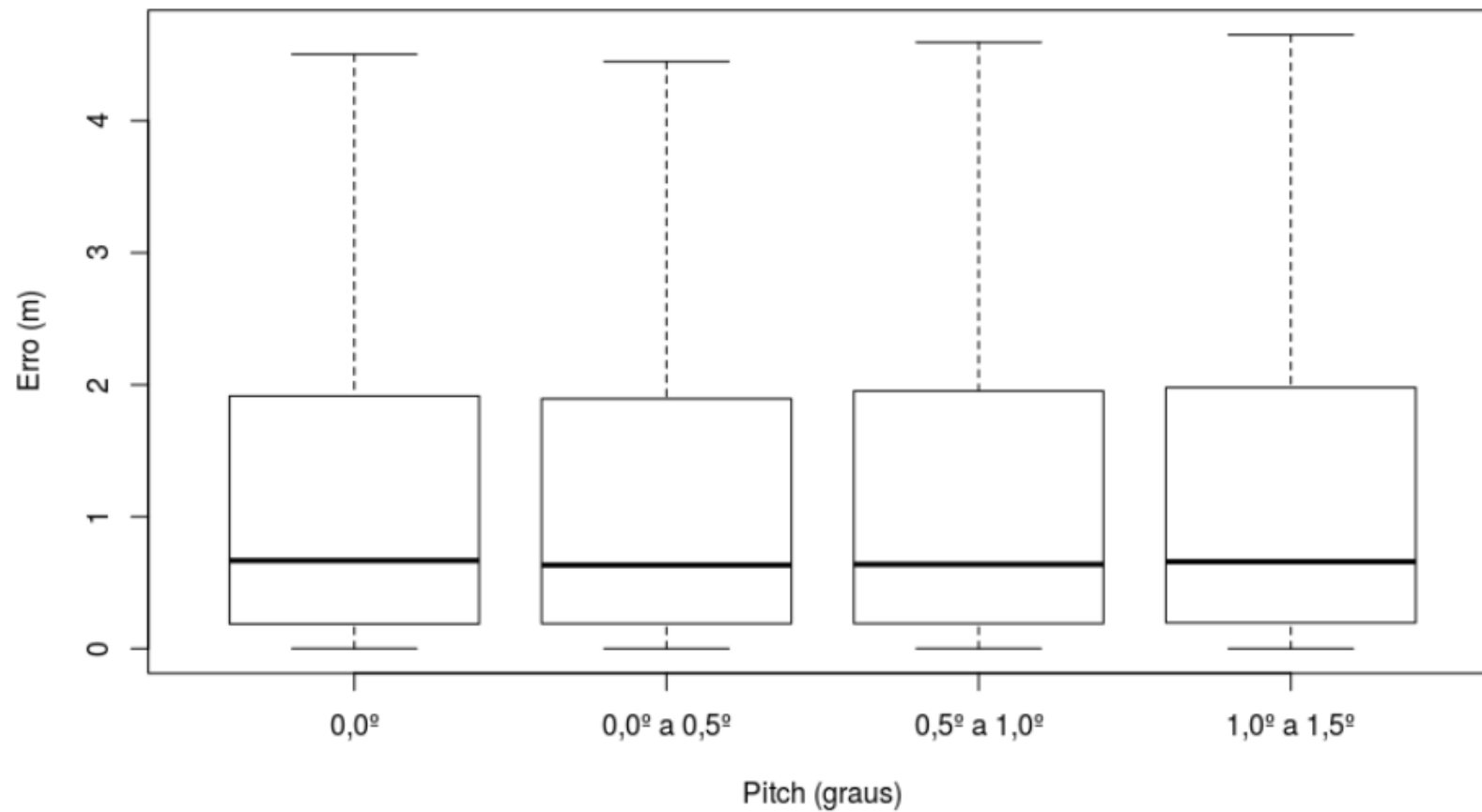
# Resultados

Relação do Erro x Pitch



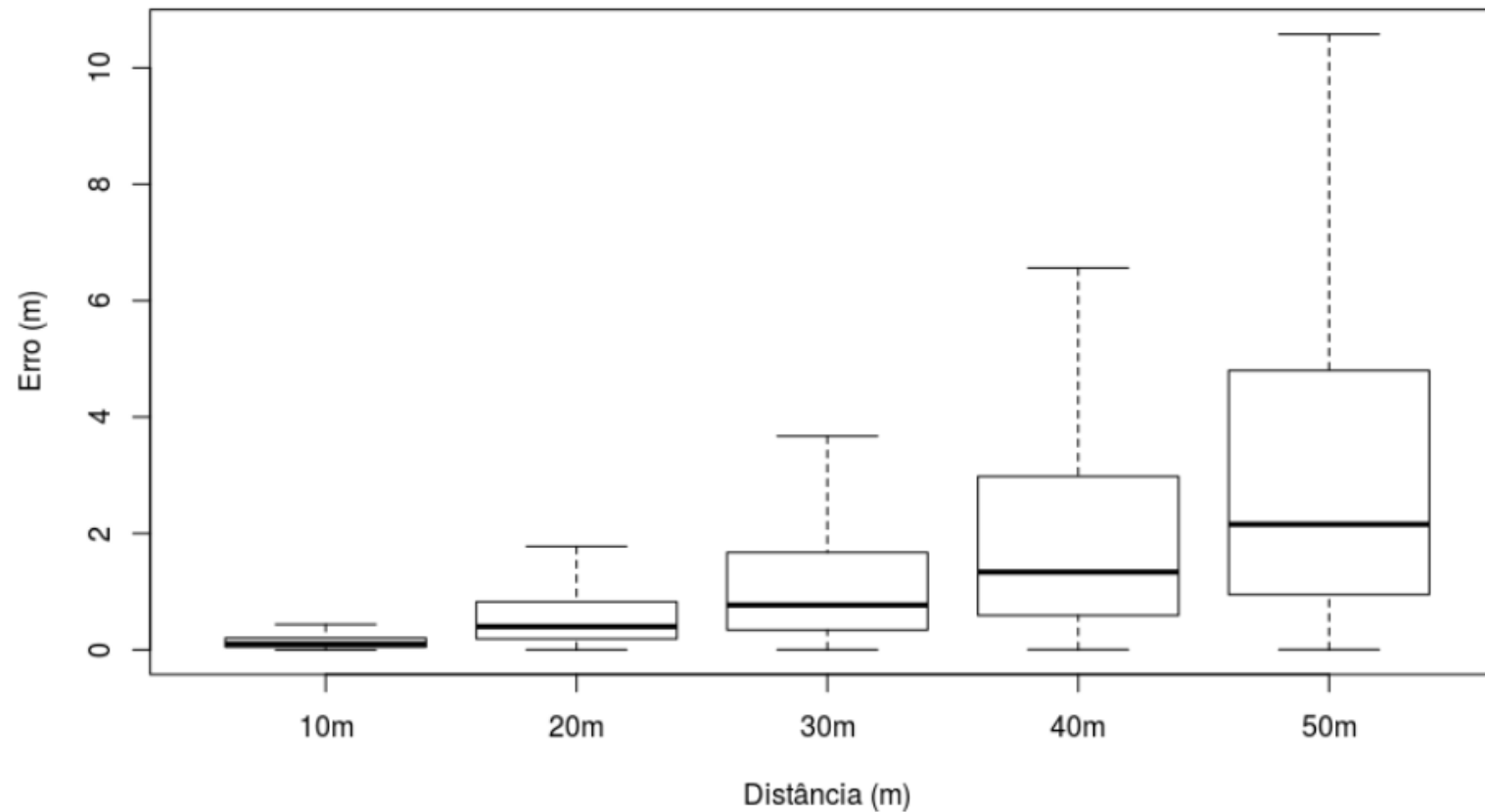
# Resultados

Relação do Erro x Pitch



# Resultados

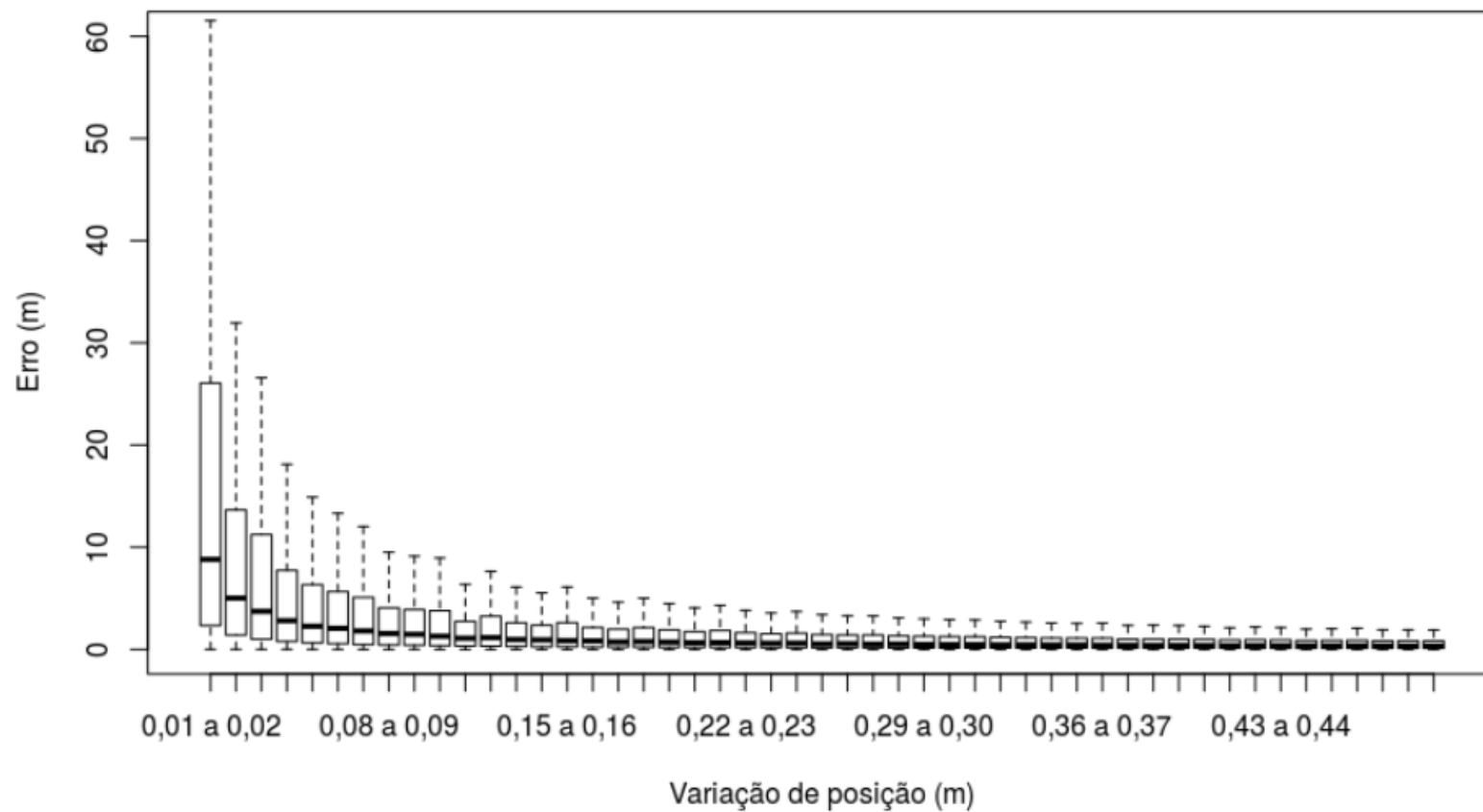
Relação do Erro x Distância





# Resultados

Relação do Erro x Variação de Posição



# Conclusões e Trabalhos Futuros

- Método se mostrou viável para a utilização em veículos aéreos não tripulados;
- Melhores resultados quando o veículo aéreo está em deslocamento e próximo ao obstáculo.
- Em trabalhos futuros é possível:
  - Utilizar técnicas de filtragem para melhorar os resultados;
  - Estender o sistema de equações para permitir também a variação no ângulo de *roll*;
  - Embarcar a técnica proposta em um ambiente real.

# Referências

- Esrafilian, O. and Taghirad, H. D. (2016). Autonomous flight and obstacle avoidance of a quadrotor by monocular SLAM. In *2016 4th International Conference on Robotics and Mechatronics (ICROM)*, pages 240–245.
- Majumder, S., Shankar, R., and Prasad, M. S. (2015). Obstacle size and proximity detection using stereo images for agile aerial robots. In *2015 2nd International Conference on Signal Processing and Integrated Networks (SPIN)*, pages 437–442. IEEE.
- Moon, J. and Prasad, J. V. R. (2011). Minimum-time approach to obstacle avoidance constrained by envelope protection for autonomous UAVs. *Mechatronics*, 21(5):861–875.
- Mur-Artal, R., Montiel, J. M. M., and Tardos, J. D. (2015). ORB-SLAM: A Versatile and Accurate Monocular SLAM System. *IEEE Transactions on Robotics*, 31(5):1147–1163.
- Saha, S., Natraj, A., and Waharte, S. (2014). A real-time monocular vision-based frontal obstacle detection and avoidance for low cost UAVs in GPS denied environment. In *2014 IEEE International Conference on Aerospace Electronics and Remote Sensing Technology*, pages 189–195. IEEE.