

DESENVOLVIMENTO DE UM ROBÔ SEGUIDOR DE LINHA UTILIZANDO PROCESSAMENTO DE IMAGENS

Alex Alves Pereira (aluno)
alexalves0619@gmail.com

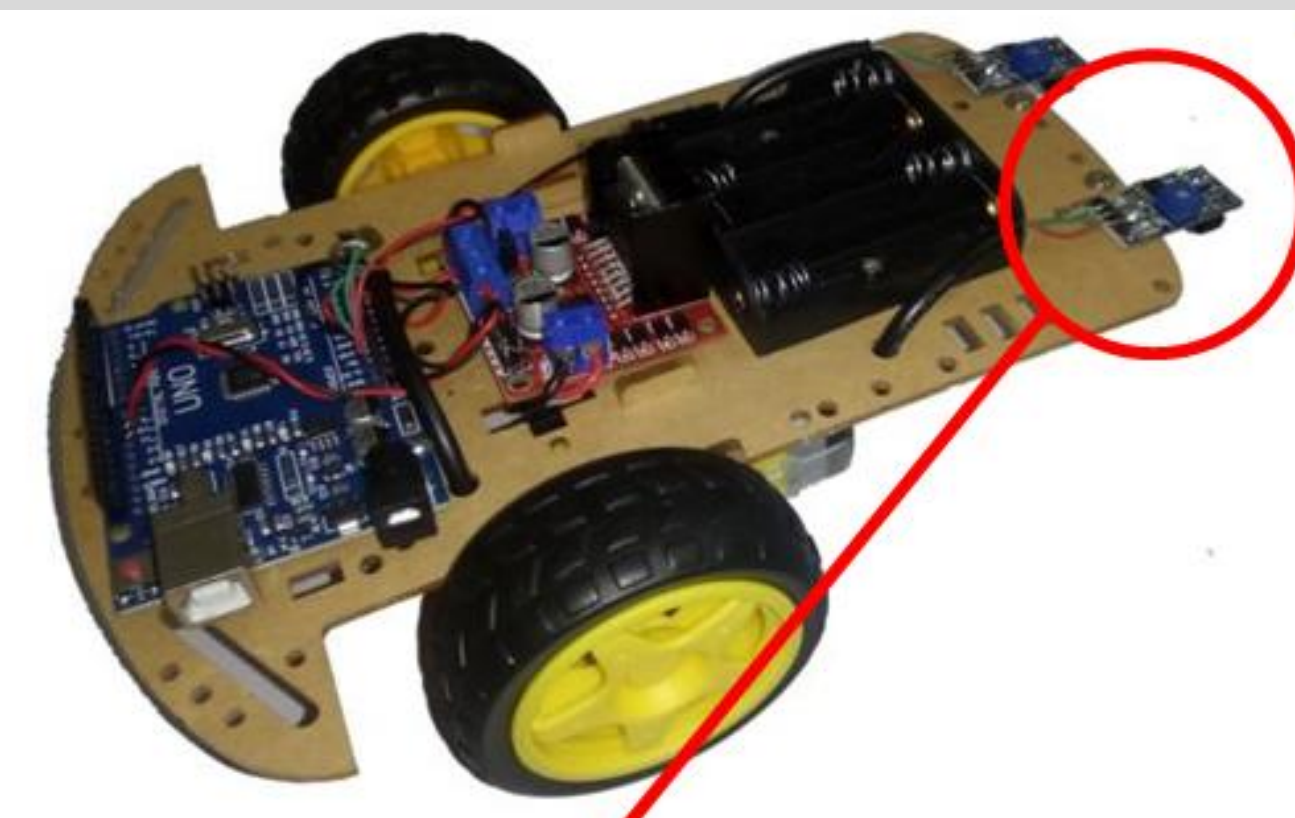
Prof. Juliano Coêlho Miranda, Dr. (orientador)
juliano.coelhomiranda@cefetmg.br

Prof. Lázaro Eduardo da Silva, Dr. (coorientador)
lazaroz@cefetmg.br

INTRODUÇÃO

Os robôs seguidores de linha utilizados em fábricas e armazéns para o transporte automatizado, no ensino e nas competições de robótica, são máquinas capazes de se mover sobre trajetórias fixas e restritas, seguindo marcas contínuas ou espaçadas, previamente fixadas ou pintadas no solo.

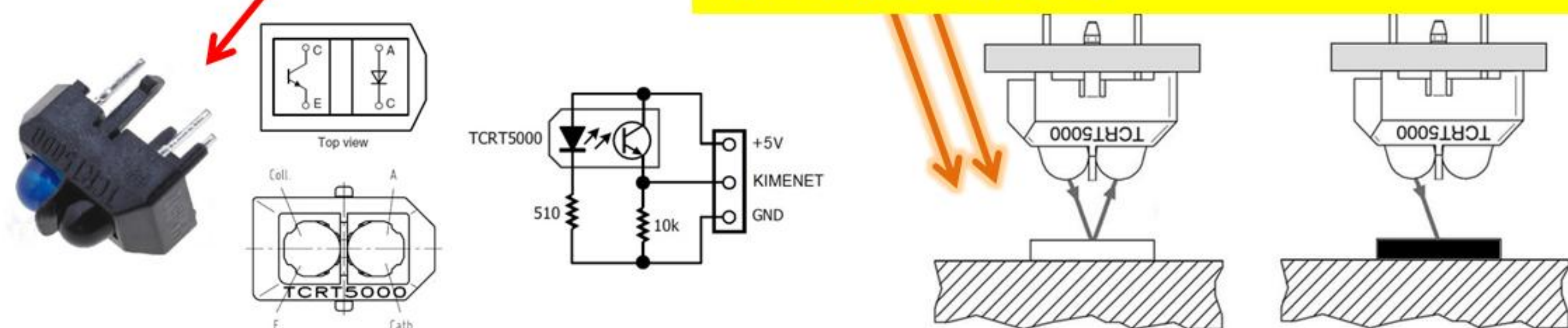
PROBLEMA



O robô seguidor de linha tradicional, tem como base sensores formados por pares simples de emissor e detector de raios infravermelhos.

O PROBLEMA

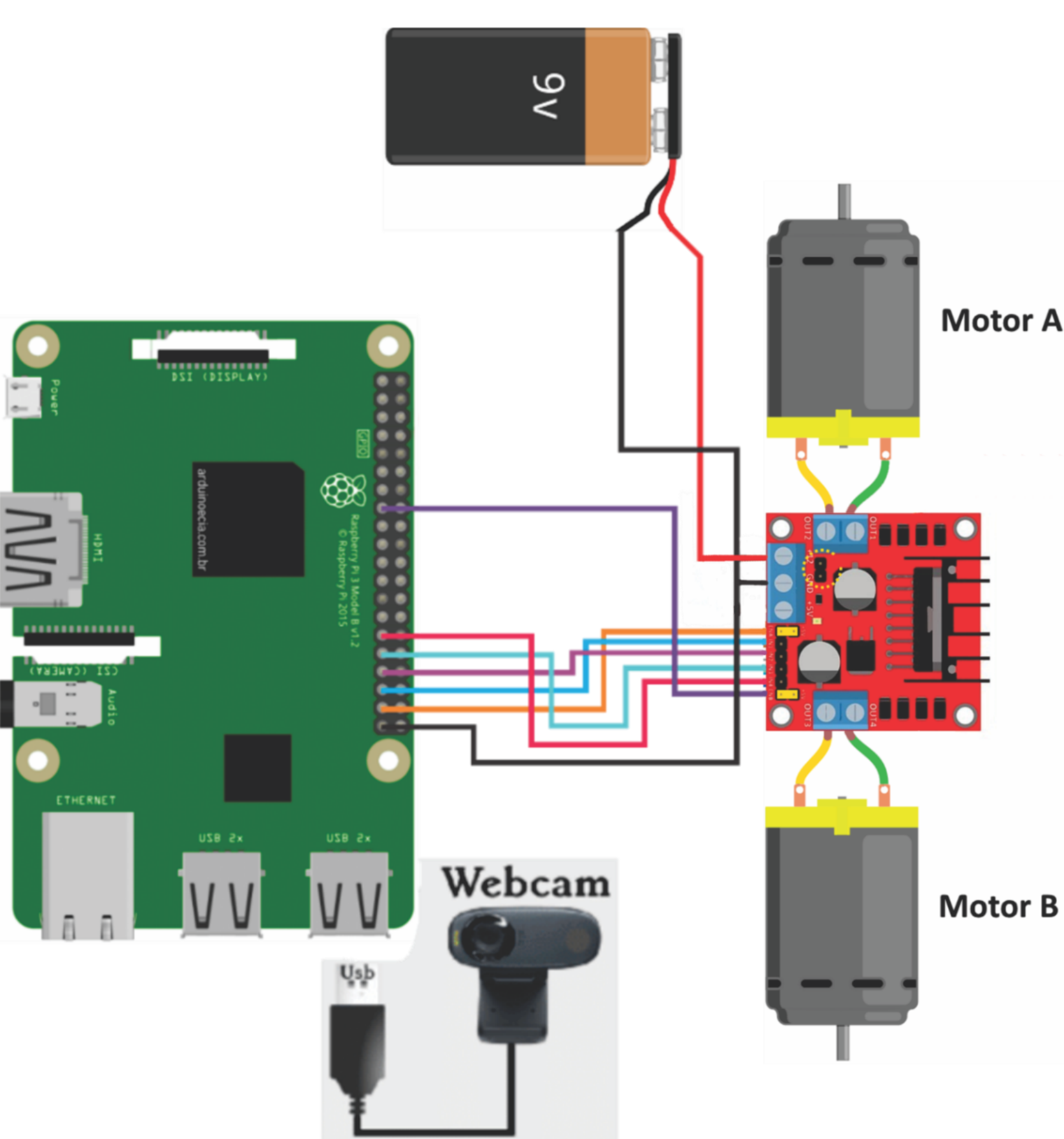
Estes sensores sofrem interferências de fontes de calor, radiação infravermelha do sol e da iluminação ambiente.



OBJETIVO

Para amenizar esses inconvenientes, esta pesquisa utilizou a visão computacional. Basicamente, um sistema de visão é formado por uma câmera, que capta uma imagem e entrega um sinal elétrico representativo dessa imagem a uma interface adequada. Neste contexto, o presente trabalho de pesquisa tem como objetivo o desenvolvimento de um robô seguidor de linha utilizando processamento de imagens.

O CIRCUITO

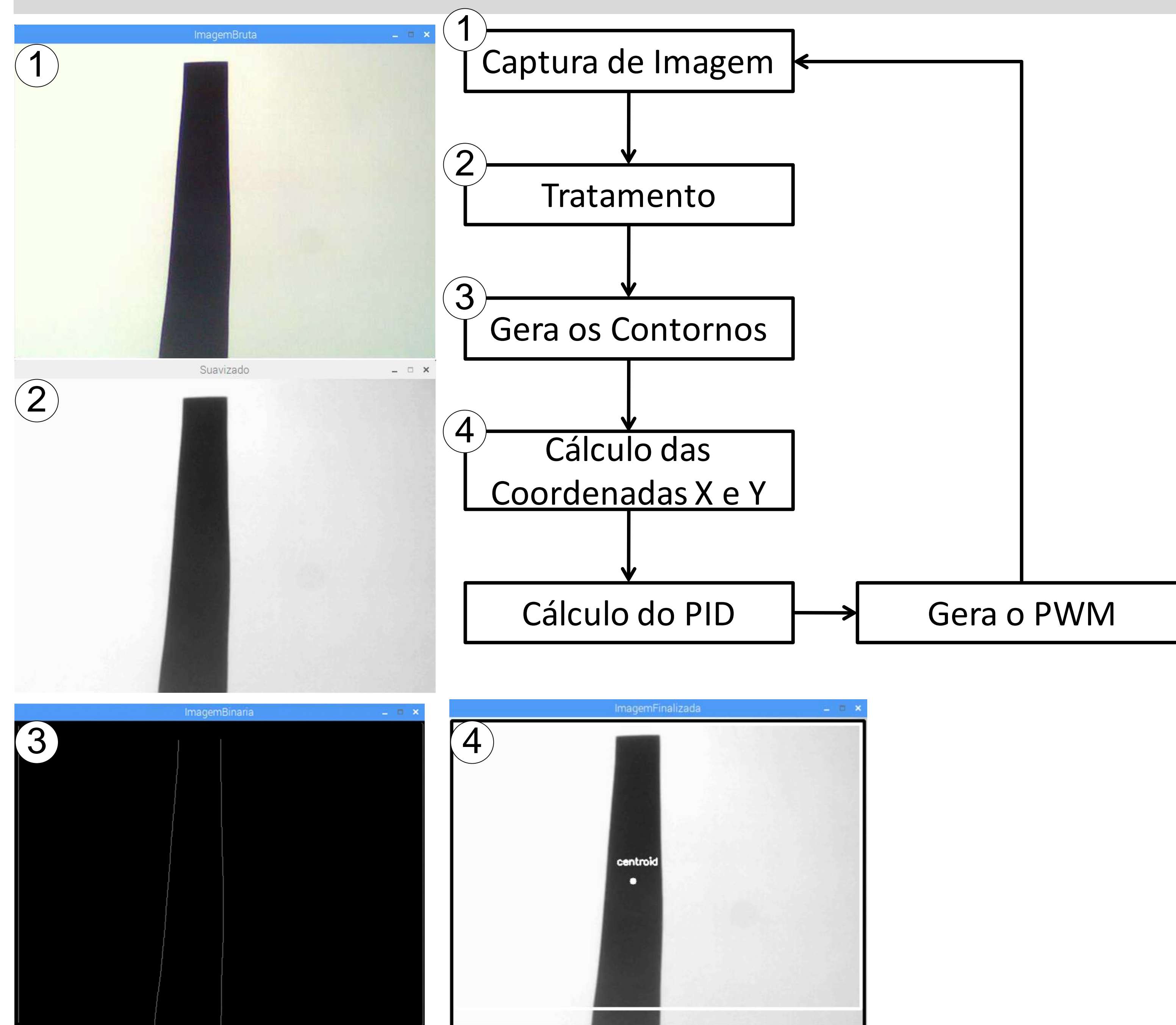


O **circuito eletrônico** do robô consiste em um Raspberry PI Versão 2 modelo B, uma webcam, o motor drive L298N, dois motores DC de 3 a 6 V e uma bateria para alimentação dos motores.

METODOLOGIA

O trabalho foi desenvolvido de acordo com as seguintes etapas: revisão bibliográfica, definição das ferramentas para desenvolvimento do algoritmo e da plataforma de hardware, desenvolvimento da visão computacional e testes do sistema.

O ALGORITMO



Quando a faixa, a linha, está centralizada obtemos um determinado valor para 'x', que ao ser aplicado no cálculo do PID retorna um valor, o erro, próximo de zero. Neste caso, o valor do PWM, do *Duty Cycle*, ou seja, da tensão aplicada para cada motor, da esquerda e da direita, são praticamente iguais.

Quando a linha se encontra mais para direita, o valor gerado pelo PID é positivo, e o *Duty Cycle* para o motor esquerdo é maior, afim de centralizar o robô.

Quando a linha se encontra mais para esquerda, o valor gerado pelo PID é negativo, e o *Duty Cycle* para o motor direito é maior, afim de centralizar o robô.

(324,	17.86,	17.14,	1.030046713555)
(324,	17.86,	17.14,	1.026901213555)
(324,	17.86,	17.14,	1.023755713555)
(324,	17.86,	17.14,	1.020610213555)
(324,	17.86,	17.14,	1.017464713555)
(324,	17.86,	17.14,	1.0143192135549999)
(324,	17.85,	17.15,	1.0111737135549999)
(324,	17.85,	17.15,	1.0080282135549998)
(324,	17.85,	17.15,	1.004882713555)
(324,	17.85,	17.15,	1.001737213555)

RESULTADOS E DISCUSSÕES

O robô foi capaz de completar uma pista simples, com êxito na execução do algoritmo, que envolve o processamento de imagens, e o controle dos motores por meio das técnicas de PID e PWM. O seguidor de linha obteve bom desempenho com baixa velocidade. As maiores dificuldades envolveram a capacidade de processamento do Raspberry PI 2 modelo B, e o uso de uma câmera comum, que interferiu no desempenho do algoritmo, e por conseguinte, do robô. Como benefício da pesquisa, é possível apontar o uso do Python, uma linguagem robusta e de alto nível, além da biblioteca OpenCV e seus recursos. Alterações são necessárias para que o processo aconteça em tempo real. Plataformas embarcadas com maior capacidade de processamento, e câmeras com diferentes resoluções, devem ser testadas na continuidade do projeto.