

# Detecção de Cones para a Formula Student Driverless

Nícolas Goeldner e Wesly Carmesini Ataide

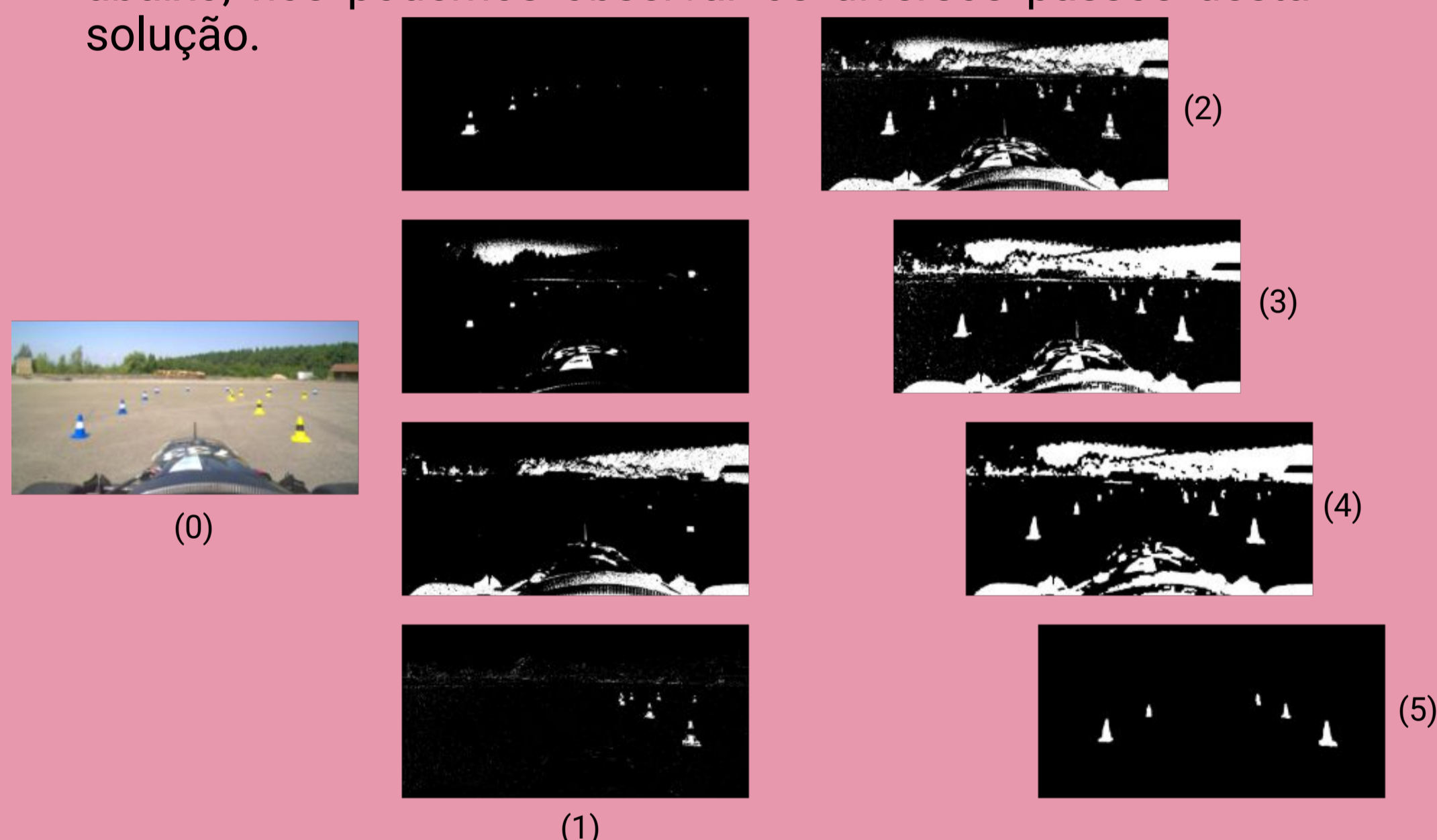
## Objetivo

O objetivo deste trabalho é desenvolver parte de um sistema de navegação autônoma para a Formula Student Driverless: a detecção de cones. Nas competições driverless, os cones sinalizam o caminho pelo qual o carro deve passar; esse caminho não é previamente conhecido. Demarcando o limite esquerdo estão os cones azuis e à direita os amarelos. Nós desenvolvemos duas soluções para a detecção de cones: uma das soluções utiliza métodos clássicos da visão computacional, já a outra solução é baseada em redes neurais convolucionais (CNN). A imagem à direita exemplifica como que são as imagens capturadas por uma câmera acoplada a um veículo da competição.



## Clássica

Esta nossa solução se baseia praticamente em técnicas de limiarização e classificação de objetos com base na sua forma. Primeiramente, nós usamos limiarização com distância de Mahalanobis para detectar o azul e o amarelo dos cones, além disso, usamos limiarização simples no RGB para detectar o branco e o preto do interior (1). Com isso, nós juntamos as imagens (2) e aplicamos uma operação de fechamento para preencher os pequenos buracos no interior dos cones (3) e, então, aplicamos uma operação de abertura para eliminar pequenos objetos que não nos interessam (4). Agora, vamos classificar os diversos objetos como cones ou não cones (5); para fazer essa classificação, nós levamos em conta diversas características dos objetos, como: o tamanho, posição na imagem, largura máxima e altura, se a largura do objeto diminui conforme olhamos mais para cima no objeto, se a base é  $x$  vezes maior que o topo do objeto. Na imagem abaixo, nós podemos observar os diversos passos desta solução.



## CNN

A nossa solução baseada em CNN utilizou como modelo o Detectron2 com Mask R-CNN e ResNet-50. O dataset que nós utilizamos foi o FSOCO, a partir do qual separamos 2640 imagens para o dataset de treinamento e 558 imagens para o dataset de validação. Nós treinamos o modelo por apenas 3 épocas, porém percebemos que foi suficiente visto que a validation loss já estava estável. Com esse modelo, nós conseguimos obter uma Average Precision de 0.817 para  $IoU \geq 50$ , já considerando  $IoU \geq 75$  nossa AP ficou em 0.565. Nós também calculamos uma forma de weighted accuracy, na qual consideramos o grau de certeza com que o modelo acredita que aquele objeto que ele detectou é um cone de determinada cor. Considerando essa weighted accuracy, nosso modelo atingiu 0.9978 para os cones azuis e 0.9992 para os cones amarelos. Na imagem abaixo, podemos ver as bounding boxes delimitando os vários cones que foram detectados e com seus respectivos graus de certeza. Aqui estamos detectando também os cones laranjas, pois nas competições eles também possuem um significado.



## Conclusão

Podemos perceber que ambos os métodos conseguiram detectar os cones na imagem, sendo que o método clássico detectou apenas os cones que estão mais próximos, porém esses são os mais importantes para a navegação autônoma na competição. Entretanto, o método clássico é extremamente sensível a variações de luminosidade, enquanto que o método de redes neurais, sofre minimamente com essas variações; por exemplo, seria bem mais difícil para o método clássico detectar cones em uma imagem capturada durante à noite. Além disso, o método clássico não consegue detectar muito bem cones que estão sobrepostos. Assim, fica evidente que, por causa das limitações do método clássico, o método baseado em CNN é a melhor escolha para ser usado no sistema de navegação autônoma para a Formula Student Driverless.