

**Projeto**  
**Reconhecimento do último dígito de placas de automóveis**  
**MAC5832 - Aprendizagem Computacional:**  
**Modelos, Algoritmos e Aplicações**

Diogo Haruki Kykuta    6879613  
Bruno Padilha            xxxxxxxx

## 1 O problema

Na cidade de São Paulo, temos um rodízio municipal de veículos no qual automóveis com placas terminadas em certos dígitos não podem trafegar em certas regiões da cidade. Então, reconhecer o último dígito de uma placa de carro a partir de uma foto pode automatizar a fiscalização do rodízio, em busca de infratores. Isso torna este um estudo interessante.

## 2 Solução Proposta

Nossa proposta de solução consiste de várias partes, podendo ser agrupadas em dois grandes grupos:

### 2.1 Treinamento

A primeira parte da solução consiste em realizar o treinamento, para que consigamos criar classificadores bons e os gravamos. Precisamos de dois classificadores: um para extrair a placa de uma foto da traseira do carro e outro para reconhecer o dígito depois de extraído.

### 2.2 Predição

Usando os resultados obtidos pelos dois treinamentos mencionados acima, recebemos uma foto (supostamente contendo uma placa) como argumento e o programa reconhece qual é o último dígito. Essa informação está no programa na forma de um inteiro, mas por se tratar apenas de um protótipo para estudo, essa informação é apenas impressa no terminal. Mas poderia isso poderia ser usado dentro de outra aplicação maior.

## 3 Problemas Encontrados

A detecção de placas mostrou-se um problema...

## 4 Solução desenvolvida

A linguagem de programação escolhida para o programa ser feito foi Python, pela facilidade que a linguagem proporciona em lidar com vetores, matrizes e também pela grande lista de bibliotecas a disposição. Dentre elas, algumas foram usadas em nosso programa, e estão listadas a seguir:

- numpy Essa biblioteca permite a manipulação de vetores e matrizes de forma mais fácil e intuitiva. Permitindo uma melhor leitura do código, além de uma alta performance. Todas as suas operações são muito otimizadas.
- opencv Usamos a versão 2 do OpenCV, cuja principal característica (interessante para nós) é a representação de imagens como objetos do numpy. Dessa forma, temos fácil acesso aos pixels da imagem, ainda podendo usar todo o ferramental dado pelo OpenCV. Essa biblioteca nos permite realizar várias operações na imagem, como operações morfológicas (como abertura e fechamento), limiarização, redimensionamento, dentre outras. OpenCV possui funções de aprendizagem computacional, mas optamos por escolher uma outra biblioteca para isso.
- sklearn Essa biblioteca nos coloca a disposição métodos de aprendizagem computacional otimizados, e escolhemos usar *Support Vector Machine (SVM)* para tratar nosso problema. A implementação que nos é fornecida permite múltiplas classes, o que é essencial para lidarmos com reconhecimento de dígito.

### 4.1 Descrição dos passos do programa

#### Passo 1: Extrair a placa

A partir de uma foto como entrada.... [TODO]

#### Passo 2: Identificar regiões de interesse na placa

Encontrar os dígitos

#### Passo 3: Recortar o último dígito

Recortar de tamanho XXX o retangulo

#### Passo 4: Identificar o último dígito

Usando SVM, com um treinamento feito previamente, identificamos o último dígito. A etapa de aprendizado usa um kernel RBF (*Radial Basis Function*), com parâmetros  $C$  e  $\gamma$  escolhidos por um passo de validação cruzada (*cross-validation*).

## 5 Resultados

[TODO: Mostrar 5-10 imagens de placas e o que o algoritmo retornou. Delas, 1 placa vermelha e 1-2 placas detectadas errado por classificador ruim e 1 placa detectada errada do fusquinho (6 antigo mais arredondado).]

## Bibliografia