P1 - PI II Vinícius Ramos __/__/__

1- a imagem foi carregada com o opencv imread e os canais BGR foram extraídos com a função split, após isso foram calculados as médias de valores dos pixels de cada canal. Por fim essas médias foram comparadas para determinar a cor dominante (entre vermelho, verde e azul). Também foram feitos casos especiais (todos os valores = 0 significa que a imagem é preta e todos os valores = 255 a imagem é branca).

2- foi feita uma moldura de zeros com a função copyMakeBorder, então foram criadas matrizes vazias para os canais RGB. Após isso, percorri a imagem com um for aninhado e passei o valor do pixel em grayscale para o canal correspondente.

	Coluna par	Coluna impar
Linha par	vermelho	verde
Linha impar	verde	azul

+ por conta da borda de zeros, a indexação das linhas e colunas começam em 1 ao invés de 0.

após esse processo, converti os canais RGB para int32 pois tive problemas de overflow. Para a interpolação, o valor de um pixel desconhecido é a média dos 4 pixels que o cercam. Isso é feito nos 3 canais. Por fim os dados são convertidos de volta para unsigned int 8 e salvos.

3- para passar a imagem para escala de cinza novamente utilizei a fórmula $G_R = (0.299 \cdot R) + (0.587 \cdot G) + (0.114 \cdot B)$ onde G_R = cinza. para achar a linha, apliquei os 4 passos do algoritmo irodata: linha inicial é a média dos pixels em escala de cinza e a nova linha

é a média das médias dos grupos abaixo e acima do limiar antigo. A partir disso o limiar antigo recebe o novo e o processo se repete até que a mudança no limiar seja menor que 0.5. Esse valor apresentou baixo tempo de execução e um resultado aceitável. É possível distinguir os telhados e topo do farol das paredes, a cerca do chão e algumas partes das nuvens.

5- Começo o código definindo as listas de pontos 2D e 3D em coordenadas homogêneas, em seguida formula as equações conforme as projeções. O sistema é resolvido pelo SVD do módulo de álgebra linear do numpy. Assim é obtida a matriz de projeção P. Para verificarmos sua acurácia, projetamos os pontos 3d usando a matriz P e calculamos o erro entre esses e os pontos 2d originais e a acurácia será 1 - erro médio. Nesse caso a acurácia foi de 82,42%.