

# Processamento Digital de Imagens

## Trabalho prático 5

### Classificadores

Josué Nunes - 3465  
Mateus Coelho - 3488  
Ricardo Spínola - 3471

19 de outubro de 2022

## Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
<b>2</b>	<b>Conceitos</b>	<b>2</b>
2.1	<i>Clusters</i> . . . . .	2
2.2	ROI <i>picker</i> . . . . .	2
2.3	<i>Kernel</i> . . . . .	2
<b>3</b>	<b>Resultados</b>	<b>2</b>
3.1	Questão 1 . . . . .	2
3.2	Questão 2 . . . . .	3
3.3	Questão 3 . . . . .	4
<b>4</b>	<b>Conclusão</b>	<b>4</b>

# 1 Introdução

O intuito dessa prática é mostrar a utilização de diferentes maneiras de realizar classificações de objetos em uma imagem, através da seleção de diferentes áreas das imagens, e diferentes números de *clusters* utilizados para realizar a diferenciação entre elementos das imagens, para no final compararmos os resultados obtidos.

## 2 Conceitos

### 2.1 *Clusters*

Os *clusters* são utilizados como uma maneira de diferenciar as cores que foram selecionadas, ou separadas pelo próprio algoritmo, então aumentando os *clusters* permite que haja mais diferenciação dos elementos. Por exemplo, utilizando 2 *clusters*, só haveria a diferenciação de 2 classes de elementos, como se é ou não é uma pedra, por exemplo, fazendo isso pela comparação de cores entre os pontos. A medida que precisamos diferenciar diferentes objetos e elementos, devemos aumentar o número de *clusters* para que fique mais claro para o programa realizar a separação entre essas classes.

### 2.2 ROI *picker*

O ROI *picker* é uma ferramenta utilizada para selecionar pontos, que na verdade são áreas em uma imagem, e com essa seleção é possível realizar a separação entre as cores selecionadas, para que, dependendo do número de *clusters* utilizados, o algoritmo possa realizar a diferenciação e agrupamento dessas áreas selecionadas.

### 2.3 *Kernel*

Um *kernel* é uma matriz que é utilizada para aplicar uma máscara à imagem original, e utilizamos 2 tipos de *kernel* nessa prática, o linear, e o rbf, que apresentam resultados finais diferentes devido à suas diferenças de construção.

## 3 Resultados

### 3.1 Questão 1

Na questão 1, utilizamos de diferentes números de *clusters* utilizados para plotar imagens, e do "falsa cor" para realizar a coloração dos grupos através de uma cor compatível para cada grupo. Utilizamos de 2 a 5 *clusters(k)*, obtendo os resultados obtidos nas figuras 1, 2, 3 e 4.



Figura 1: Falsa cor com  $k=2$ .

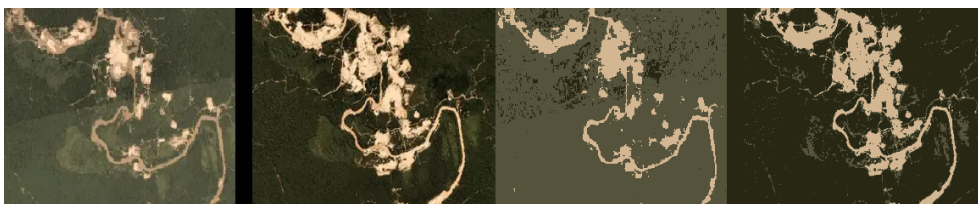


Figura 2: Falsa cor com  $k=3$ .

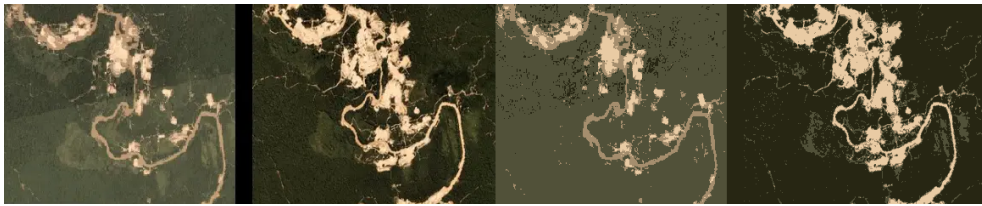


Figura 3: Falsa cor com  $k=4$ .

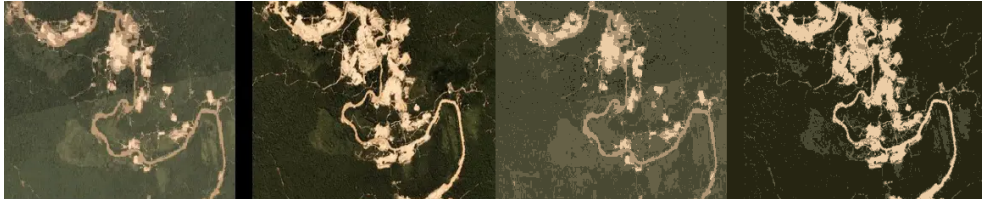


Figura 4: Falsa cor com  $k=5$ .

Podemos perceber que com 2 *clusters*, há uma divisão muito imprecisa entre os tipos de vegetação, pois no resultado tudo fica com a mesma cor, e as nuvens também são vistas como desmatamento. Já com 3 *clusters*, há uma maior diferenciação da vegetação, mas as nuvens continuam no mesmo grupo do desmatamento. Com 4 *clusters*, já há essa diferenciação entre nuvens e áreas desmatadas, e por fim, com 5 *clusters*, há uma pequena variação novamente na vegetação, mas notamos que não é de grande usabilidade para nós nesse momento utilizarmos de 5 *clusters* para fazer uma análise por exemplo da área desmatada.

### 3.2 Questão 2

Na questão 2, aplicamos o *kernel* linear e o *kernel* rbf para termos uma saída com cores mais destacadas, e escolhemos a imagem resultante da questão 1 utilizando 3 *clusters*, pelo fato de que nela, ainda não há a diferenciação de nuvens e desmatamento, mas como não sabemos o que existe por baixo da nuvem, optamos por considerá-la como área desmatada. Os resultados obtidos podem ser observados nas imagens 5 e 6.

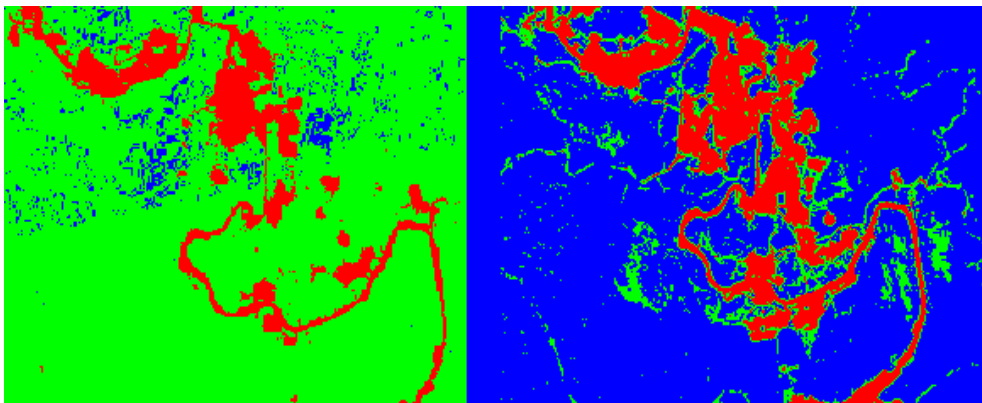


Figura 5: Resultado do *Kernel* linear

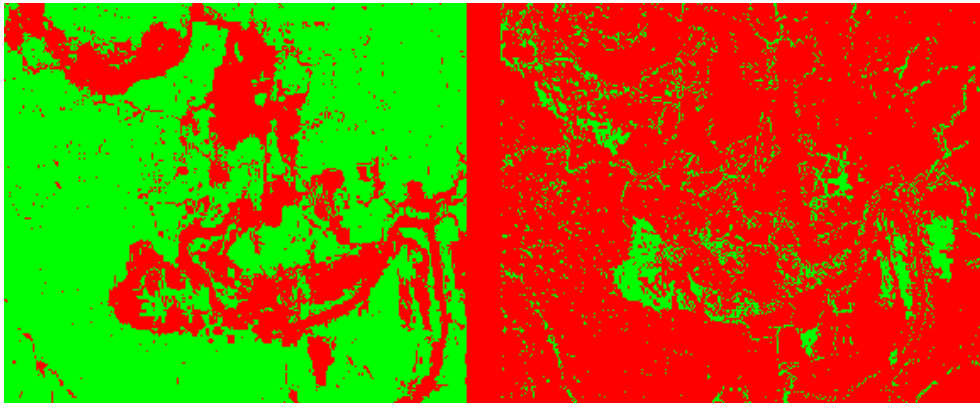


Figura 6: Resultado do *Kernel* rbf

Não obtivemos um resultado muito satisfatório utilizando o rbf, mas já com o linear, podemos ver com muita clareza que o azul e o verde são cores e elementos da vegetação, e o vermelho é a área desmatada.

### 3.3 Questão 3

Para realizarmos o cálculo de quanto havia de área desmatada contamos os *pixels* vermelhos, do antes(esquerda) e também do depois(direita), e assim foi possível verificarmos que antes haviam 7243 *pixels*, e depois haviam 9966 *pixels* vermelhos, portanto, realizando os cálculos com esses valores obtidos, concluímos que houve um aumento de aproximadamente 37,59%

## 4 Conclusão

Com essa prática podemos concluir que há uma grande área de aplicação dos algoritmos de classificações, e que a escolha de quais métodos e quantos *clusters* devem ser utilizados podem influenciar bastante no resultado final que será obtido pelo programa, pois se a quantidade de *clusters* for baixa, o resultado será muito impreciso, mas que também não é necessário usar muitos, pois isso resultaria em um maior custo computacional, sendo que para muitos casos não altera tanto o resultado de acordo com que o programador deseja encontrar.

**Obs.:** Todas as imagens resultante estão no arquivo que acompanha este documento.