

**DCC029/868 – Processamento de Imagens Digitais**

Prof. Jefersson Alex dos Santos (jefersson@dcc.ufmg.br)

Monitor Caio Cesar Viana da Silva (caiosilva@ufmg.br)

## TP1: COMPOSIÇÃO CHROMA KEY

### 1 Descrição

O objetivo deste trabalho é a implementação de um programa na linguagem Python, em um notebook Jupyter, para realizar a composição de imagens com fundo verde.

Em processamento de imagens, o processo de *chroma key* consiste em substituir o fundo de uma imagem por outra imagem. Uma das formas mais simples de fazer isso é utilizar uma imagem com fundo de uma cor uniforme, como por exemplo verde, desta forma a técnica se resume em substituir os pixels que apresentam a cor uniforme pelos pixels da outra imagem. Este processo é muito utilizado em filmes para facilitar a criação de cenários ou até mesmo possibilitar a criação de objetos/personagens e locais fictícios. Esta técnica de chroma key também é muito utilizada para realizar a composição de imagens, ou seja, juntar diversas imagens com fundo uniforme em uma apenas de forma a criar uma nova cena, esta é a principal aplicação do chroma key nos filmes.

Para realizar uma composição de imagens, as mesmas passam pela remoção do fundo, assim como operações radiométricas, geométricas e o uso de filtros de forma a ficarem mais adequadas ao resultado final.

Neste trabalho prático, você irá realizar uma composição das imagens de entrada, utilizando as operações radiométricas, geométricas, filtros e chroma key, resultando em uma imagem apenas. A Figura 1 apresenta um exemplo de imagens utilizadas na composição chroma key. Já a Figura 2 apresenta o resultado da composição.

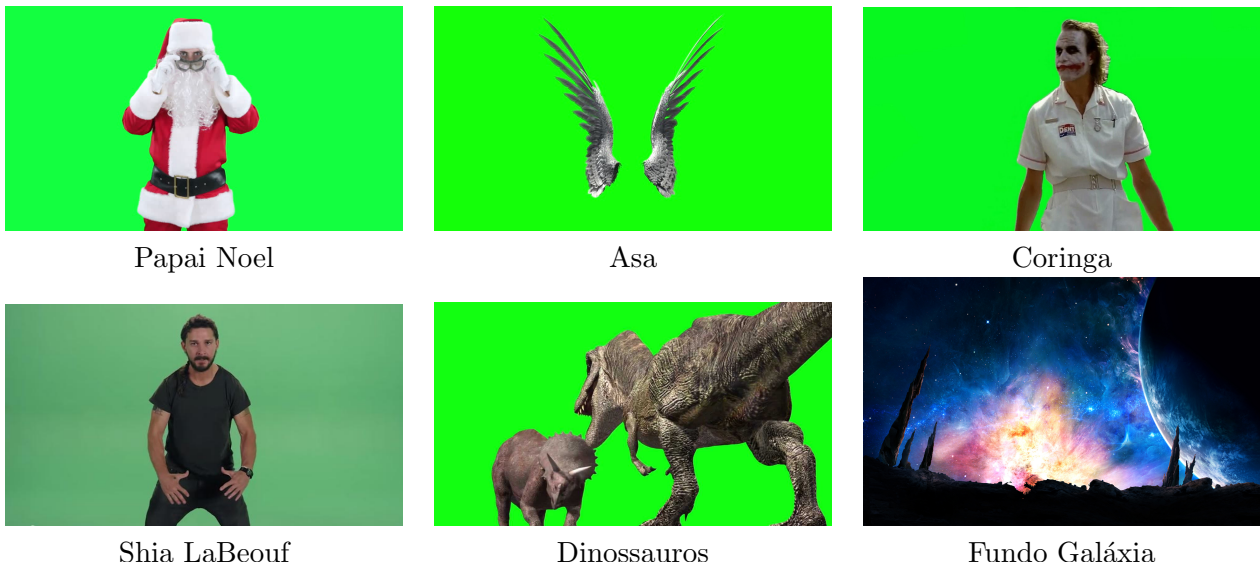


Figura 1: Exemplos de imagens utilizadas na composição Chroma Key.



Figura 2: Exemplos de resultado da composição das imagens.

## 2 Formato da entrada e da saída

Nesse trabalho prático, você deverá implementar um programa que recebe como **entrada** no mínimo 5 imagens, sendo uma para ser utilizada como fundo e as demais com fundo uniforme (verde, por exemplo). Como **saída**, o programa deverá gravar em disco e exibir a imagem da combinação das anteriores.

## 3 O que precisa ser feito

O seu programa deverá cumprir os seguintes requisitos:

1. Carregar no mínimo 5 imagens
2. Aplicar a técnica de chroma key nas imagens (menos na de fundo)
3. Aplicar operações radiométricas, geométricas e de filtros nas imagens (no mínimo uma operação de cada)
4. Combinar todas as imagens em uma só
5. Salvar e exibir a imagem resultante

## 4 O que será fornecido

Qualquer imagem pode ser utilizada no programa. No entanto, será fornecido:

- 5 imagens com fundo uniforme (verde);
- 5 imagens de fundo;

## 5 Entrega do código / Documentação

O código e a documentação devem ser entregues em um único arquivo notebook Jupyter (.ipynb). Caso sejam utilizadas imagens não fornecidas pelo professor, deve-se entregar um arquivo Zip com as imagens utilizadas e o arquivo notebook Jupyter.

O texto da documentação deve ser breve, de forma que o corretor possa entender o que foi feito no código sem ter que entender linha a linha dos arquivos. Implementações modularizadas deverão mencionar quais funções são implementadas em cada módulo ou classe. A documentação deve conter os seguintes itens:

- Sumário do problema a ser tratado.
- Uma descrição sucinta dos algoritmos, das principais funções, e procedimentos.
- Decisões de implementação que porventura estejam omissos na especificação.

O código deve estar inserido ao longo da documentação e deve estar funcional, de forma que a avaliação será realizada ao executar o notebook Jupyter para obter a imagem final.

Obs.: Programas que não compilarem não serão corrigidos. Trabalhos que forem entregues sem a documentação não serão corrigidos. Trabalhos que forem entregues em algum formato que não notebook Jupyter não serão corrigidos.

## 6 Avaliação

Esse trabalho prático vale **15 pontos**, sendo os pontos distribuídos para a implementação correta de cada um dos requisitos detalhados na Seção 3. O trabalho pode ser penalizado (obter pontos negativos) se a documentação, clareza do código e funcionamento correto do programa não estiverem de acordo com o requerido.

O melhor resultado, a ser avaliado pelo professor e monitor da disciplina, em quesitos criatividade/-qualidade/operações realizadas ganhará pontos extras na próxima atividade avaliativa.

As regras para desconto de nota por atraso são:

- Atraso de até 24 horas: -1.5 pontos.
- Atraso de 24 horas até 48 horas: -3 pontos.
- Atraso de 48 horas até 72 horas: -4.5 pontos.
- Atraso de mais de 72 horas: -10 pontos.