Instituto Federal de Educação, Ciência e Tecnologia do Ceará

Engenharia da Computação

Computação Gráfica

Prof. Lucas Silva, de Sousa

André Vieira da Silva

Relatório 03 : Manipulação de Imagens através da variação de alpha

Fortaleza

Agosto de 2018

Sumário

[O canal Alpha 3](#_Toc522458432)

[Resultados Obtidos 4](#_Toc522458433)

[Conclusão 4](#_Toc522458434)

[Referencias 4](#_Toc522458435)

[Implementação 6](#_Toc522458436)

# O canal Alpha

Para a computação gráfica, a composição Alfa é o processo de combinar uma imagem com um fundo para criar a aparência de transparência. Ela é usada para processar elementos da imagem em passos separados, para combinar várias imagens em uma só.   
 Para isso é preciso do canal Alfa. Ele é uma máscara que específica como as cores serão juntadas com outros pixels quando estiverem um em cima do outro.

A combinação das imagens na manipulação do canal alpha pode ser obtida pela formula:

Onde :

.

.

É possível dizer que o canal alfa é o quarto elemento da cor de um pixel. Ou seja, além de *R*ed (vermelho), *Green*(verde), *Blue*(azul) — que formam a sigla RGB — também há o Alfa, que define quão opaca ou transparente será a imagem. Isso é muito comum em editores de imagem, por exemplo.

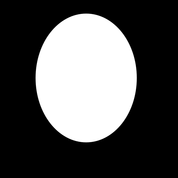
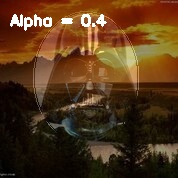
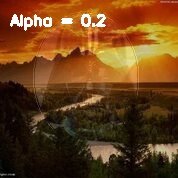
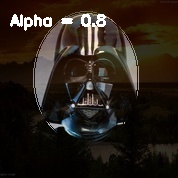
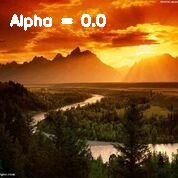
Abaixo as imagens que serão utilizados no processo de manipulação de alpha:

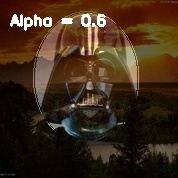
Figura 1 Mascara para delimitar a área que serão sobreposta.

Figura 2 Imagem de Fundo que será sobreposta

Figura 3 Imagem que sobrepora o Fundo

# Resultados Obtidos

Variando o valor de alpha no intervalo já informado em incrementos de 0.2 foram obtidos os seguintes resultados:



# Conclusão

A manipulação do valor de alpha permite obter a combinação de imagens através da produto escalar dos complementos dos valor de alpha para as respectivas matriz que representam as imagens.Este valor no praticamente define o quanto de cada imagem de entrada esta presente na imagem resultante .

# Referencias

1. [https://www.tecmundo.com.br/video-game-e-jogos/1358-o-que-e-o-canal-alfa-e-para-que-ele-serve-.htm em 19/08/2018](https://www.tecmundo.com.br/video-game-e-jogos/1358-o-que-e-o-canal-alfa-e-para-que-ele-serve-.htm%20em%2019/08/2018)
2. <https://www.dropbox.com/sh/rv3qc5tcjo0vp3z/AADXFLiVL2ID8vJMF85Lhh0Pa/praticas/canal_alpha?dl=0&preview=descricao.pdf&subfolder_nav_tracking=1> em 19/08/2018

# Implementação

A implementação usou três imagens de entrada para a obtenção dos resultados já apresentados. O código fonte foi redigido em linguagem python 3 com ajuda da ide Pycharm da JetBrains fazendo uso de paradigma de Orientação a Objetos. E por fim foi feito o uso das bibliotecas Opensource OpenCv , e Numpy apenas para a abertura ,salvamento e acompanhamento do processo de processamento das imagens. Segue abaixo o código fonte usado para este trabalho.

1 import cv2

2 from config import config

3 import copy

4

5

6 class Image:

7 def \_\_init\_\_(self, nameimage):

8 self.image = None

9 self.nameimage = nameimage

10

11 def load\_image(self, modo = 0):

12 try:

13 path\_image = config.PATH\_INPUT\_IMAGES + self.nameimage

14 if(modo in [0,1,2,3]):

15 self.image = cv2.imread(path\_image, modo)

16 else:

17 self.image = cv2.imread(path\_image,cv2.IMREAD\_COLOR)

18 if self.image is None:

19 raise IOError("FILE {namefile} NOT FOUND".format(namefile=self.nameimage))

20 except IOError as exc:

21 print("ERROR : {args} ".format(args=exc.args))

22

23 def show\_image(self):

24 try:

25 if self.image is None:

26 raise IOError("FILE {namefile} NOT FOUND".format (namefile=self.nameimage))

27 cv2.imshow(self.nameimage, self.image)

28 cv2.waitKey(0)

29 cv2.destroyAllwindows()

30 except IOError as exc:

31 print("ERROR : {args} ".format(args=exc.args))

32 def write\_text(self,image,text):

33 font = cv2.FONT\_HERSHEY\_SIMPLEX

34 bottomLeftCornerOfText = (10, 25)

35 fontScale = 1/2

36 fontColor = (255, 255, 255)

37 lineType = 2

38 cv2.putText(image, text,

39 bottomLeftCornerOfText,

40 font,

41 fontScale,

42 fontColor,

43 lineType)

44

45

46 def alpha\_channel(self,background,mask,alpha\_factor):

47 filtered\_image = copy.deepcopy(background)

48 # b,g,r =

49 # channels = [b,g,r]

50 for l in range(0, self.image.shape[0]):

51 for co in range(0, self.image.shape[1]):

52 for ch in range(0, len(cv2.split(self.image))):

53 filtered\_image[l][co][ch] = ((mask[l][co][ch] and (self.image[l][co][ch])) \* (alpha\_factor)) +\

54 (background[l][co][ch] \* (1 - alpha\_factor))

55

56 self.write\_text(filtered\_image, "Alpha = "+str(alpha\_factor))

57 cv2.imwrite(config.PATH\_OUTPUT\_IMAGES + "ALPHA\_"+ str(alpha\_factor) + self.nameimage + ".jpg",

58 filtered\_image)

59

60

61 foreground = Image("foreground.png")

62 foreground.load\_image('a')

63

64

65 background = Image("background.jpeg")

66 background.load\_image('a')

67

68

69 mask = Image("alpha\_channel.png")

70 mask.load\_image("a")

71 for i in range(0,11,2):

72 foreground.alpha\_channel(background.image,mask.image,i/10)

73