

## 9-1 透過合成

プログラム名: Blend.py (GPU版はBlend\_gpu.py)  
 CPU版 <https://interface.cqpub.co.jp/9-1blend-py/>,  
 GPU版 /9-1blend\_gpu-py/



CPU版



GPU版



図1 ベース画像

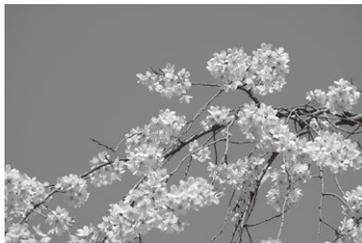


図2 合成する画像



図3 図1:図2=3:7の比率で透過合成



図4 図1:図2=7:3の比率で透過合成

2枚の画像を合成する手法は、画像解析の処理前、処理後で比較する場合などに使われます。その際にただ重ねただけでは、上の画像に下の画像が隠れてしまうので、透過を使用します。

## ● 画像処理の結果

図1をベース画像として図2をブレンドして、出力結果を見てみます。図3は「図1:図2=3:7」の比率、図4は「図1:図2=7:3」の比率で透過合成しています。図3および図4は透過度の設定比率が違うため、

画像の見え方が異なります。

以下に示すプログラムの使用方法は関数1つで簡単に実行できるので、引数のパラメータを変更しながら確認してみてください。なお、合成する画像のサイズは同じにする必要があります。

## ● CPU版…プログラム名: Blend.py

## ▶ リスト1: 020行…合成の実行

```
cv2.addWeighted(src1=img1, alpha=1,
src2=img2, beta=0.3, gamma=0)
```

第1引数 src1は入力画像(ベース画像, 図1)です。

第2引数 alphaはベース画像の透過比率です(重ね画像に対する)。

第3引数 src2は入力画像(重ね画像)です。

第4引数 betaは重ね画像の透過比率(ベース画像に対して)です。

第5引数 gammaは画像全体の明るさで、大きな値にするほど白っぽくなります。

alphaとbetaの比率については、合計値が1で

## リスト1 2つの画像を透過合成するCPU版プログラムBlend.py

```
000: import cv2
001:
002: def __main__():
003:     img1 = cv2.imread('IMG_0260.JPG')
004:     img2 = cv2.imread('IMG_0181.JPG')
005:
006:     img1 = getResize(img1)
007:     org = img1.copy()
008:     img2 = getResize(img2)
009:
010:     # ここで処理を実行
011:     imgMask = getMaskImg(img1, img2)
012:
013:     cv2.imshow('Original1', img1)
014:     cv2.imshow('Original2', img2)
015:     cv2.imshow('Final result', imgMask)
016:     cv2.waitKey(0)
017:     cv2.destroyAllWindows()
018: # 合成実行

019: def getMaskImg(img1, img2):
020:     compImg = cv2.addWeighted(src1=img1, alpha=1,
021:                               src2=img2, beta=0.3, gamma=0)
022:     return compImg
023:
024: def getResize(src):
025:     basePixSize = 1000 # 縦横で大きい辺の変更したいサイズ
026:     height = src.shape[0]
027:     width = src.shape[1]
028:
029:     largeSize = max(height, width)
030:
031:     # 大きい方の辺のサイズ
032:     resizeRate = basePixSize / largeSize
033:
034:     # 変更比率を計算
035:     # shotSize = min(height, width) * resizeRate
036:     dst = cv2.resize(src, (int(width *
037:                               resizeRate), int(height *
038:                               resizeRate)))
```