**OpenCV\_Python图像融合时两张图片大小不一致的解决方法**

当进行[图像融合](https://so.csdn.net/so/search?q=%E5%9B%BE%E5%83%8F%E8%9E%8D%E5%90%88&spm=1001.2101.3001.7020)时，被叠加的两张图片的大小、类型（高度/宽度/通道数）必须相同。[请点击查看](https://blog.csdn.net/sinat_38814578/article/details/83153400)

但是如果这两张图片大小不相同，怎么解决？有两种方法可以解决这个问题：

1. 重置其中一张图片的大小类型，使其与另一张图片大小类型相同；
2. 在较大的图片中创建感兴趣区域roi，roi的大小类型应与另一张图片的相同。

**注意：方法1改变图片大小时，图片的分辨率也会发生变化，因此图片的内容会产生形变；方法2没有改变图片的大小，故不会有这种问题产生。**

**主要函数**

1. dst = cv2.resize(src, dsize[, dst[, fx[, fy[, interpolation]]]])

* scr：原图像
* dsize：输出的图像大小
* dst：输出的图像。当参数dsize不为0时，dst的大小为size；否则，它的大小需要根据src的大小，fx和fy决定。d s i z e = S i z e ( r o u n d ( f x ∗ s r c . c o l s ) , r o u n d ( f y ∗ s r c . r o w s ) ) dsize = Size(round(fx\*src.cols), round(fy\*src.rows))*dsize*=*Size*(*round*(*fx*∗*src*.*cols*),*round*(*fy*∗*src*.*rows*))dst的类型与原图像相同
* fx:沿水平轴的比例因子 ( d o u b l e ) d s i z e . w i d t h / s r c . c o l s (double)dsize.width/src.cols(*double*)*dsize*.*width*/*src*.*cols*
* fy:沿垂直轴的比例因子 ( d o u b l e ) d s i z e . h e i g h t / s r c . r o w s (double)dsize.height/src.rows(*double*)*dsize*.*height*/*src*.*rows*

**参数dsize和参数(fx, fy)不能够同时为0**

* interpolation：插值方法，共5种
  + INTER\_LINEAR - 双线性插值（默认）（放大图像推荐使用）
  + INTER\_NEAREST - 最近邻插值
  + INTER\_AREA - 基于像素局部的重采样插值（缩小图像推荐使用）。该方法对于图像抽取（image decimation）来说可能更好，但如果是放大图像，和最近邻插值效果类似。
  + INTER\_CUBIC - 基于4x4像素邻域的3次插值（放大图像推荐使用）
  + INTER\_LANCZOS4 - 基于8x8像素邻域的Lanczos（兰索斯）插值

引用官方文档[Geometric Image Transformations](http://www.opencv.org.cn/opencvdoc/2.3.2/html/modules/imgproc/doc/geometric_transformations.html?highlight=resize#cv.Resize)

dst – Destination image. It has the size dsize (when it is non-zero) or the size computed from src.size() , fx , and fy . The type of dst is the same as of src .  
dsize – Destination image size. If it is zero, it is computed as:  
d s i z e = S i z e ( r o u n d ( f x ∗ s r c . c o l s ) , r o u n d ( f y ∗ s r c . r o w s ) ) dsize = Size(round(fx\*src.cols), round(fy\*src.rows))*dsize*=*Size*(*round*(*fx*∗*src*.*cols*),*round*(*fy*∗*src*.*rows*))

**注意：也就是说cv2.resize函数的dst和dsize参数输入是 （宽度，高度），与img.shape相反。**  
对于img.shape的输出参数以及对应坐标请[点击这里查看](https://blog.csdn.net/sinat_38814578/article/details/83117996)

1. cv2.addWeighted(src1, alpha, src2, beta, gamma[, dst[, dtype]]) 图像融合函数请[点击这里查看](https://blog.csdn.net/sinat_38814578/article/details/83153400)

方法1：重置其中一张图片的大小类型，使其与另一张图片大小类型相同

import cv2

import numpy as np

import matplotlib.pyplot as plt

import os

os.chdir('C:/Users/lenovo/Desktop/')

bear = cv2.imread('BearBrown.jpg')

sky = cv2.imread('Sky.jpg')

rows, cols = sky.shape[:2] #获取sky的高度、宽度

#print(sky.shape[:2]) #(800, 1200)

#print(bear.shape[:2]) #(224, 224)

bear\_dst = cv2.resize(bear,(cols,rows),interpolation=cv2.INTER\_CUBIC) #放大图像

add\_img = cv2.addWeighted(bear\_dst,0.6,sky,0.4,0) #图像融合

# 显示图片

titles = ['BearBrown','Sky','add\_img']

imgs = [bear,sky,add\_img]

for i in range(len(imgs)):

plt.subplot(2,3,i+1)

imgs[i]=cv2.cvtColor(imgs[i],cv2.COLOR\_BGR2RGB)

plt.imshow(imgs[i],'gray')

plt.title(titles[i])

plt.axis('off')

plt.show()

效果展示：（布朗熊的高度宽度其实比天空的小）  


方法2：在较大的图片中创建感兴趣区域roi，roi的大小类型应与另一张图片的相同

import cv2

import numpy as np

import matplotlib.pyplot as plt

import os

os.chdir('C:/Users/lenovo/Desktop/')

bear = cv2.imread('BearBrown.jpg')

sky = cv2.imread('Sky.jpg')

# 根据小图像的大小，在大图像上创建感兴趣区域roi（放置位置任意取）

rows, cols = bear.shape[:2] #获取bear的高度、宽度

roi = sky[0:rows, 0:cols]

dst = cv2.addWeighted(bear,0.8,roi,0.2,0) #图像融合

add\_img = sky.copy() #对原图像进行拷贝

add\_img[0:rows, 0:cols] = dst # 将融合后的区域放进原图

# 显示图片

titles = ['BearBrown','Sky','add\_img']

imgs = [bear,sky,add\_img]

for i in range(len(imgs)):

plt.subplot(2,3,i+1)

imgs[i]=cv2.cvtColor(imgs[i],cv2.COLOR\_BGR2RGB)

plt.imshow(imgs[i],'gray')

plt.title(titles[i])

plt.axis('off')

plt.show()

效果展示：（布朗熊的高度宽度其实比天空的小）  
  
方法2——没有小图像的大小，同时还可以自由设置小图像在大图像的什么位置融合。

专栏目录

[python opencv.怎么合并两张大小不一的图\_Python：使用OpenCV混合两个不同大小的图像...](https://blog.csdn.net/weixin_36391538/article/details/113962919" \t "_blank)

[python设置子图的坐标\_python如何调整子图的大小?\_weix...](https://blog.csdn.net/weixin_39708502/article/details/110447186" \t "_blank)

[python对比两张大小不同图片\_[274]用python对比两张图片的不同](https://blog.csdn.net/weixin_36200490/article/details/112025680" \t "_blank)

[opencv中实现两幅不同尺寸大小图片的相加](https://blog.csdn.net/muzhiai/article/details/8857899" \t "_blank)

[OpenCV——两幅相同与不同尺寸图像融合](https://blog.csdn.net/chen134225/article/details/80900926" \t "_blank)

[python设置子图的大小\_python - 使用matplotlib中的许...](https://blog.csdn.net/weixin_39557419/article/details/110369987" \t "_blank)

[python:matplotlib画图总结,同时展示不同大小的图,显示...](https://blog.csdn.net/Leon_winter/article/details/90202239" \t "_blank)

[OpenCV如何叠加大小不同的图片](https://blog.csdn.net/weixin_41424481/article/details/96474330" \t "_blank)

[python中matplotlib绘制子图\_python matplotlib 绘制子...](https://blog.csdn.net/weixin_42524883/article/details/113968419" \t "_blank)

3-15

[python中matplotlib绘制子图\_python matplotlib 绘制子图,如何保证大小相同 def show\_img\_bright(rgb,hsv,mask,masked): fig,(ax1,ax2,ax3,ax4) = plt.subplots(1,4,figsize=(10,10))](https://blog.csdn.net/weixin_42524883/article/details/113968419" \t "_blank)

[Python绘图基础:认识绘图区 + subplot创建子图\_猪脚SHO...](https://blog.csdn.net/weixin_47269399/article/details/124289755" \t "_blank)

6-7

[Python绘图基础:认识绘图区 + subplot创建子图 本小白在用 Python 绘制图形的时候发现可以利用多种代码来完成相同的图形的绘制,今天就给大家分享自己在利用 Python 绘图过程中总结的思路。 话不多说,直接开始 !!!](https://blog.csdn.net/weixin_47269399/article/details/124289755" \t "_blank)

[python两张图片显示在一张图上](https://blog.csdn.net/weixin_41765699/article/details/83097429" \t "_blank)

[热门推荐](https://blog.csdn.net/weixin_41765699/article/details/83097429" \t "_blank)

[Opencv学习笔记（一）：图像融合函数 addWeighted](https://blog.csdn.net/flame_007/article/details/106523271" \t "_blank)

[使用Python调整图像大小](https://panchuangai.blog.csdn.net/article/details/108805963" \t "_blank)

[Python+OpenCV实现基于金字塔的图像融合](https://blog.csdn.net/qq_45717425/article/details/122638358" \t "_blank)

[opencv实现图像融合python\_python使用OpenCV模块如何实现图像融合 使用OpenCV模块实现图像融合代码示例...](https://blog.csdn.net/weixin_29437633/article/details/111901231" \t "_blank)

[C#实现在PictureBox中显示一张本地图片](https://blog.csdn.net/qq_43474959/article/details/109448837" \t "_blank)

[python openCV学习——快速的图像匹配](https://blog.csdn.net/Terie/article/details/102825967" \t "_blank)

[python opencv.怎么合并两张大小不一的图\_python-将两个图像与OpenCV组合](https://blog.csdn.net/weixin_32822843/article/details/113962916" \t "_blank)

[Python 调整子图大小、位置【plt.axes()自定义子图】](https://blog.csdn.net/LeopoldF/article/details/121737386" \t "_blank)

[python调整子图之间的间距\_python – 在matplotlib中改进子图大小/间距与许多子图...](https://blog.csdn.net/weixin_40008566/article/details/110703331" \t "_blank)

[图片找茬游戏——使用OpenCV查找两张图像的不同处并标记](https://blog.csdn.net/matt45m/article/details/121569616" \t "_blank)

[OpenCv中两幅图像的比较运算、逻辑运算（学习笔记）](https://blog.csdn.net/Xx_Studying/article/details/122241061" \t "_blank)

[案例-使用python实现基于opencv的图像拼接(合并)](https://blog.csdn.net/qq_33392383/article/details/122748501" \t "_blank)

[最新发布](https://blog.csdn.net/qq_33392383/article/details/122748501" \t "_blank)

[【OpenCV 例程200篇】16. 不同尺寸的图像加法](https://youcans.blog.csdn.net/article/details/121234811" \t "_blank)

[python-OpenCV图像的融合](https://blog.csdn.net/qq_39434183/article/details/119723645" \t "_blank)

**热门文章**

* [OpenCV\_Python图像融合时两张图片大小不一致的解决方法 [https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png](https://blog.csdn.net/sinat_38814578/article/details/83184742) 14436](https://blog.csdn.net/sinat_38814578/article/details/83184742)
* [OpenCV\_Python官方文档7——图像融合 [https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png](https://blog.csdn.net/sinat_38814578/article/details/83153400) 5209](https://blog.csdn.net/sinat_38814578/article/details/83153400)
* [OpenCV图像高宽及坐标的转换问题——cv2.resize()/cv2.warpAffine() [https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png](https://blog.csdn.net/sinat_38814578/article/details/83117996) 4291](https://blog.csdn.net/sinat_38814578/article/details/83117996)
* [OpenCV\_Python读取图片时如何设置工作路径 [https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png](https://blog.csdn.net/sinat_38814578/article/details/82952625) 4139](https://blog.csdn.net/sinat_38814578/article/details/82952625)
* [OpenCV\_Python官方文档7+——按位运算之给图像加logo [https://csdnimg.cn/release/blogv2/dist/pc/img/readCountWhite.png](https://blog.csdn.net/sinat_38814578/article/details/83184670) 3649](https://blog.csdn.net/sinat_38814578/article/details/83184670)