Be Warning While Switching Among OpenCV, Tensorflow, and Pillow

This article will discuss one such use case in detail where OpenCV and Tensorflow show the differences in **reading** and **resizing** a JPEG image. It will also show a way to make them work consistently.

The Diffenrences in Reading and Resizing

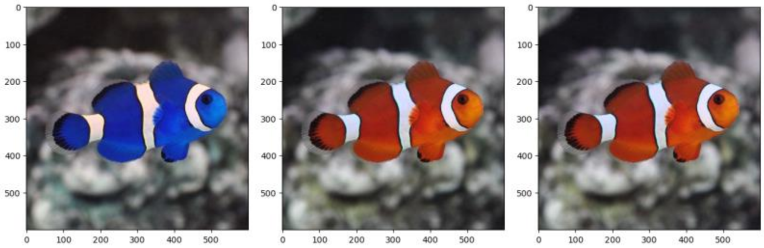
For this expreriment, the used packages is:

1. OpenCV v4.6.0
2. Tensorflow v2.3.0
3. Pillow v9.3.0

Step-1:
Preparing the test image

Figure 1 - Test Image (orange.jpg)

Step-2:
Read the test image using OpenCV, Tensorflow, and Pillow



Above images are images that have been readed by OpenCV -> Tensorflow -> Pillow

Step-3:

1. Substract the image that was readed by OpenCV with Tensorflow
2. Substract the image that was readed by OpenCV with Pillow
3. Substract the image that was readed by Pillow with Tensorflow

Result for the experiment no.1:

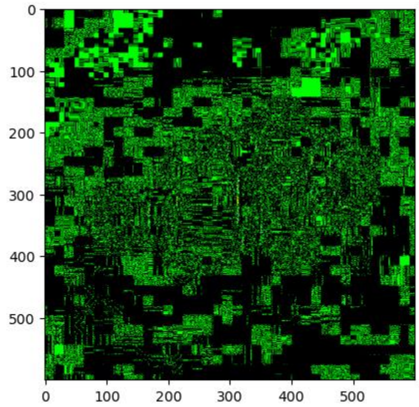


Figure 2 - Image Differences Between OpenCV and Tensroflow

Woah! That's a huge difference. What went wrong? Well, this difference is arising from the fact that OpenCV, by default, uses **Integer Accurate Decompression** of the JPEG image. In contrast, TensorFlow uses **Discrete Cosine Transform** as default. This type of decoding is inaccurate and so to make it the same as OpenCV, we need to decode it by using integer accurate decompression.

This can be done by setting the parameter **`dct_method='INTEGER_ACCURATE'`** as shown below.

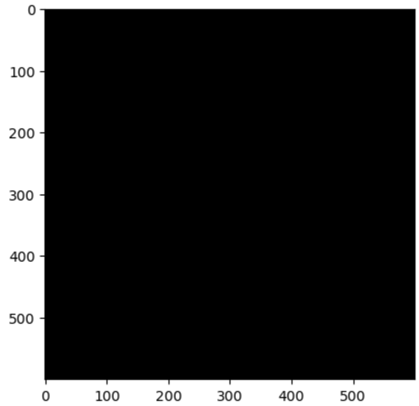


Figure 3 - Subtracted Image After INTEGER_ACCURATE parameter is set

Now, since we have read the image consistently with OpenCV and TensorFlow, let's try the resizing of the images with these frameworks. We'll begin with OpenCV resize first.

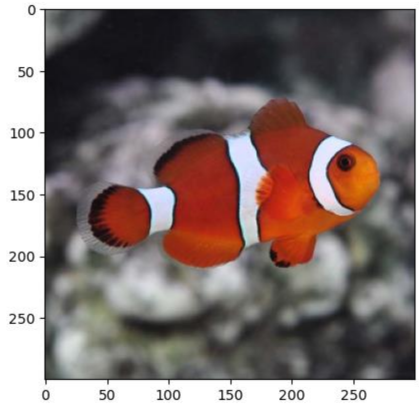


Figure 4 - Image Resizing using OpenCV

Then we resize the image using tf.image.resize, both images are resized using **Nearest Neighbours Interpolation** method, here the result of tensorflow resize function:

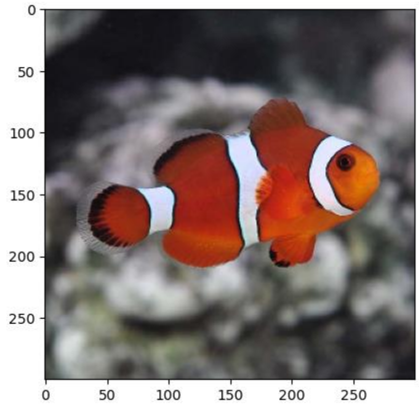


Figure 5 - Image Resizing using Tensorflow

Then after resizing, we subtract those images again, and here is the result:

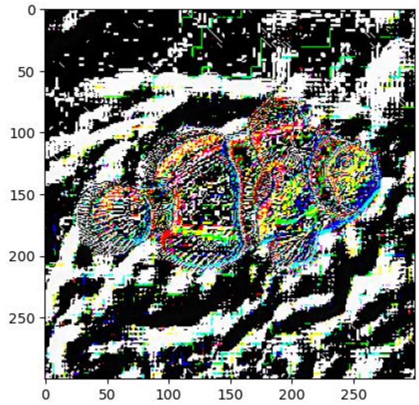


Figure 6 - Subtracted Resized Image (OpenCV - Tensorflow)

Result for the experiment no.2:

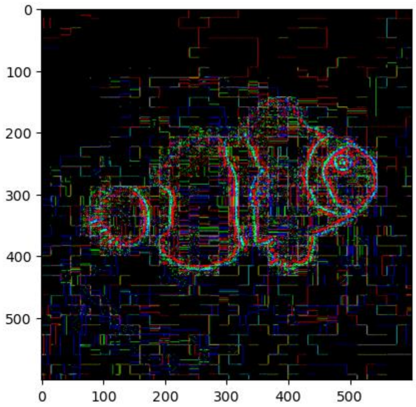


Figure 7 - Substracted Image Between OpenCV and Pillow

Result for the experiment no.3:

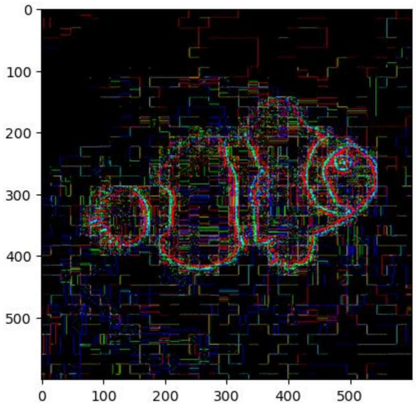


Figure 8 - Substracted Image Between Pillow and Tensorflow