

# 1\_Align\_RGB

June 20, 2018

This code uses the Python Open CV library to align red, green and blue band images in the format tif and combines them to produce an aligned RGB image in the format JPG. In addition, this code transfers the GPS information from the image in the reference band to the aligned RGB image.

Extensions asociated to each name file refers to a diferente band as is detailed below

- '\*\_1.tif' = Blue Band
- '\*\_2.tif' = Green Band
- '\*\_3.tif' = Red Band
- '\*\_4.tif' = Near Infrared Band
- '\*\_5.tif' = Rededge Band

```
In [2]: ##Preamble
        %matplotlib inline
        import cv2
        import numpy as np
        import matplotlib.pyplot as plt
        import os
        import glob
        import traceback
        import sys
        from PIL import Image, ImageFilter
        import PIL
        import matplotlib.pyplot as plt

        import datetime
        now = datetime.datetime.now()

        print(now)
```

2018-05-29 17:53:44.373389

```
In [3]: ##Setting directory
        dir_set='/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001'
        num_set=dir_set[-2:]
        ##Creating lists and listing blue band images
        filenames=glob.glob(dir_set+'/*.tif')
```

```
filenamesBLUE=[]
filenamesGREEN=[]
filenamesRED=[]
for i in range(len(filenames)):
    if filenames[i][83]=='1':
        filenamesBLUE.append(filenames[i])

filenamesBLUE.sort()
filenamesBLUE
```

[illegible]







```

'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0182_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0183_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0184_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0185_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0186_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0187_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0188_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0189_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0190_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0191_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0192_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0193_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0194_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0195_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0196_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0197_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0198_1.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0199_1.t

```

In [4]: *##Listing green band images*

```

filenamesGREEN=[]

for i in range(len(filenames)):
    if filenames[i][83]=='2':
        filenamesGREEN.append(filenames[i])

filenamesGREEN.sort()
filenamesGREEN

```

Out [4]:

```

['/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0000_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0001_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0002_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0003_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0004_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0005_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0006_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0007_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0008_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0009_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0010_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0011_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0012_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0013_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0014_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0015_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0016_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0017_2.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0018_2.t

```











[illegible]

[illegible]





```

'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0192_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0193_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0194_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0195_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0196_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0197_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0198_3.t
'/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001/IMG_0199_3.t

```

In [6]: *##Creating folder to save aligned images*

```

if not os.path.exists(dir_set+'_RGB_ALIGNED/'):
    os.makedirs(dir_set+'_RGB_ALIGNED/')

```

In [7]: *##Preparing to align images*

```

#Define motion model
warp_mode = cv2.MOTION_TRANSLATION  #(Options: MOTION_EUCLIDEAN, MOTION_AFFINE and MOTION_HOMOGRAPHY)
# Set the warp matrix to identity.
if warp_mode == cv2.MOTION_HOMOGRAPHY:
    warp_matrix = np.eye(3, 3, dtype=np.float32)
else:
    warp_matrix = np.eye(2, 3, dtype=np.float32)

```

In [8]: *##Set the stopping criteria for the algorithm.*

```

criteria = (cv2.TERM_CRITERIA_EPS | cv2.TERM_CRITERIA_COUNT, 100000, 1e-5)

```

In [9]: *dir\_rgb=dir\_set+'\_RGB\_ALIGNED/'*

```

print(dir_rgb)

```

```

/tidchile/collaborative/udd/ids/agro/AGRIFRUT_10-02/MICASENSE/0000SET/001_RGB_ALIGNED/

```

In [10]: *##Aligning images and saving as .JPG*

```

#Creating log file. List images which were not aligned
log = open(dir_rgb + "error_align.log", "w")

for i in range(len(filenameesRED)):
    try:
        print("Processing: " + filenameesBLUE[i][74:82])
        im_blue      = cv2.imread(filenameesBLUE[i],0);
        im_green     = cv2.imread(filenameesGREEN[i],0);
        im_red       = cv2.imread(filenameesRED[i],0);

        sz = im_blue.shape
        print (sz)
        height = int(sz[0])
        width = sz[1]
    except:
        log.write(filenameesBLUE[i] + " not aligned\n")

```

```

im_aligned = np.zeros((height,width,3), dtype=np.uint8 )
im_aligned[:, :,2] = im_red[:, :]

#Blue (Aligning Blue)
(cc, warp_matrix) = cv2.findTransformECC (im_red[:, :], im_blue[:, :],warp_matrix,

im_aligned[:, :,0] = cv2.warpAffine(im_blue[:, :], warp_matrix, (width, height),

#Green (Aligning Green)
(cc, warp_matrix) = cv2.findTransformECC (im_red[:, :], im_green[:, :],warp_matrix,

im_aligned[:, :,1] = cv2.warpAffine(im_green[:, :], warp_matrix, (width, height),
cv2.imwrite(dir_rgb + filenamesBLUE[i][74:82]+'_RGB.JPG',im_aligned)
except Exception:
    print("Error processing: " + filenamesBLUE[i][74:82])
    log.write("Error processing: " + filenamesBLUE[i][74:82] + '\n')
    continue

```

```

Processing: IMG_0000
(960, 1280)
Processing: IMG_0001
(960, 1280)
Processing: IMG_0002
(960, 1280)
Processing: IMG_0003
(960, 1280)
Processing: IMG_0004
(960, 1280)
Processing: IMG_0005
(960, 1280)
Processing: IMG_0006
(960, 1280)
Processing: IMG_0007
(960, 1280)
Processing: IMG_0008
(960, 1280)
Processing: IMG_0009
(960, 1280)
Processing: IMG_0010
(960, 1280)
Processing: IMG_0011
(960, 1280)
Processing: IMG_0012
(960, 1280)
Processing: IMG_0013
(960, 1280)
Processing: IMG_0014

```



(960, 1280)  
Processing: IMG\_0015  
(960, 1280)  
Processing: IMG\_0016  
(960, 1280)  
Processing: IMG\_0017  
(960, 1280)  
Processing: IMG\_0018  
(960, 1280)  
Processing: IMG\_0019  
(960, 1280)  
Processing: IMG\_0020  
(960, 1280)  
Processing: IMG\_0021  
(960, 1280)  
Processing: IMG\_0022  
(960, 1280)  
Processing: IMG\_0023  
(960, 1280)  
Processing: IMG\_0024  
(960, 1280)  
Processing: IMG\_0025  
(960, 1280)  
Processing: IMG\_0026  
(960, 1280)  
Processing: IMG\_0027  
(960, 1280)  
Processing: IMG\_0028  
(960, 1280)  
Processing: IMG\_0029  
(960, 1280)  
Processing: IMG\_0030  
(960, 1280)  
Processing: IMG\_0031  
(960, 1280)  
Processing: IMG\_0032  
(960, 1280)  
Processing: IMG\_0033  
(960, 1280)  
Processing: IMG\_0034  
(960, 1280)  
Processing: IMG\_0035  
(960, 1280)  
Processing: IMG\_0036  
(960, 1280)  
Processing: IMG\_0037  
(960, 1280)  
Processing: IMG\_0038

(960, 1280)  
Processing: IMG\_0039  
(960, 1280)  
Processing: IMG\_0040  
(960, 1280)  
Processing: IMG\_0041  
(960, 1280)  
Processing: IMG\_0042  
(960, 1280)  
Processing: IMG\_0043  
(960, 1280)  
Processing: IMG\_0044  
(960, 1280)  
Processing: IMG\_0045  
(960, 1280)  
Processing: IMG\_0046  
(960, 1280)  
Processing: IMG\_0047  
(960, 1280)  
Processing: IMG\_0048  
(960, 1280)  
Processing: IMG\_0049  
(960, 1280)  
Processing: IMG\_0050  
(960, 1280)  
Processing: IMG\_0051  
(960, 1280)  
Processing: IMG\_0052  
(960, 1280)  
Processing: IMG\_0053  
(960, 1280)  
Processing: IMG\_0054  
(960, 1280)  
Processing: IMG\_0055  
(960, 1280)  
Processing: IMG\_0056  
(960, 1280)  
Processing: IMG\_0057  
(960, 1280)  
Processing: IMG\_0058  
(960, 1280)  
Processing: IMG\_0059  
(960, 1280)  
Processing: IMG\_0060  
(960, 1280)  
Processing: IMG\_0061  
(960, 1280)  
Processing: IMG\_0062

(960, 1280)  
Processing: IMG\_0063  
(960, 1280)  
Processing: IMG\_0064  
(960, 1280)  
Processing: IMG\_0065  
(960, 1280)  
Processing: IMG\_0066  
(960, 1280)  
Processing: IMG\_0067  
(960, 1280)  
Processing: IMG\_0068  
(960, 1280)  
Processing: IMG\_0069  
(960, 1280)  
Processing: IMG\_0070  
(960, 1280)  
Processing: IMG\_0071  
(960, 1280)  
Processing: IMG\_0072  
(960, 1280)  
Processing: IMG\_0073  
(960, 1280)  
Processing: IMG\_0074  
(960, 1280)  
Processing: IMG\_0075  
(960, 1280)  
Processing: IMG\_0076  
(960, 1280)  
Processing: IMG\_0077  
(960, 1280)  
Processing: IMG\_0078  
(960, 1280)  
Processing: IMG\_0079  
(960, 1280)  
Processing: IMG\_0080  
(960, 1280)  
Processing: IMG\_0081  
(960, 1280)  
Processing: IMG\_0082  
(960, 1280)  
Processing: IMG\_0083  
(960, 1280)  
Processing: IMG\_0084  
(960, 1280)  
Processing: IMG\_0085  
(960, 1280)  
Processing: IMG\_0086

(960, 1280)  
Processing: IMG\_0087  
(960, 1280)  
Processing: IMG\_0088  
(960, 1280)  
Processing: IMG\_0089  
(960, 1280)  
Processing: IMG\_0090  
(960, 1280)  
Processing: IMG\_0091  
(960, 1280)  
Processing: IMG\_0092  
(960, 1280)  
Processing: IMG\_0093  
(960, 1280)  
Processing: IMG\_0094  
(960, 1280)  
Processing: IMG\_0095  
(960, 1280)  
Processing: IMG\_0096  
(960, 1280)  
Processing: IMG\_0097  
(960, 1280)  
Processing: IMG\_0098  
(960, 1280)  
Processing: IMG\_0099  
(960, 1280)  
Processing: IMG\_0100  
(960, 1280)  
Processing: IMG\_0101  
(960, 1280)  
Processing: IMG\_0102  
(960, 1280)  
Processing: IMG\_0103  
(960, 1280)  
Processing: IMG\_0104  
(960, 1280)  
Processing: IMG\_0105  
(960, 1280)  
Processing: IMG\_0106  
(960, 1280)  
Processing: IMG\_0107  
(960, 1280)  
Processing: IMG\_0108  
(960, 1280)  
Processing: IMG\_0109  
(960, 1280)  
Processing: IMG\_0110

(960, 1280)  
Processing: IMG\_0111  
(960, 1280)  
Processing: IMG\_0112  
(960, 1280)  
Processing: IMG\_0113  
(960, 1280)  
Processing: IMG\_0114  
(960, 1280)  
Processing: IMG\_0115  
(960, 1280)  
Processing: IMG\_0116  
(960, 1280)  
Processing: IMG\_0117  
(960, 1280)  
Processing: IMG\_0118  
(960, 1280)  
Processing: IMG\_0119  
(960, 1280)  
Processing: IMG\_0120  
(960, 1280)  
Processing: IMG\_0121  
(960, 1280)  
Processing: IMG\_0122  
(960, 1280)  
Processing: IMG\_0123  
(960, 1280)  
Processing: IMG\_0124  
(960, 1280)  
Processing: IMG\_0125  
(960, 1280)  
Processing: IMG\_0126  
(960, 1280)  
Processing: IMG\_0127  
(960, 1280)  
Processing: IMG\_0128  
(960, 1280)  
Processing: IMG\_0129  
(960, 1280)  
Processing: IMG\_0130  
(960, 1280)  
Processing: IMG\_0131  
(960, 1280)  
Processing: IMG\_0132  
(960, 1280)  
Processing: IMG\_0133  
(960, 1280)  
Processing: IMG\_0134

(960, 1280)  
Processing: IMG\_0135  
(960, 1280)  
Processing: IMG\_0136  
(960, 1280)  
Processing: IMG\_0137  
(960, 1280)  
Processing: IMG\_0138  
(960, 1280)  
Processing: IMG\_0139  
(960, 1280)  
Processing: IMG\_0140  
(960, 1280)  
Processing: IMG\_0141  
(960, 1280)  
Processing: IMG\_0142  
(960, 1280)  
Processing: IMG\_0143  
(960, 1280)  
Processing: IMG\_0144  
(960, 1280)  
Processing: IMG\_0145  
(960, 1280)  
Processing: IMG\_0146  
(960, 1280)  
Processing: IMG\_0147  
(960, 1280)  
Processing: IMG\_0148  
(960, 1280)  
Processing: IMG\_0149  
(960, 1280)  
Processing: IMG\_0150  
(960, 1280)  
Processing: IMG\_0151  
(960, 1280)  
Processing: IMG\_0152  
(960, 1280)  
Processing: IMG\_0153  
(960, 1280)  
Processing: IMG\_0154  
(960, 1280)  
Processing: IMG\_0155  
(960, 1280)  
Processing: IMG\_0156  
(960, 1280)  
Processing: IMG\_0157  
(960, 1280)  
Processing: IMG\_0158

(960, 1280)  
Processing: IMG\_0159  
(960, 1280)  
Processing: IMG\_0160  
(960, 1280)  
Processing: IMG\_0161  
(960, 1280)  
Processing: IMG\_0162  
(960, 1280)  
Processing: IMG\_0163  
(960, 1280)  
Processing: IMG\_0164  
(960, 1280)  
Processing: IMG\_0165  
(960, 1280)  
Processing: IMG\_0166  
(960, 1280)  
Processing: IMG\_0167  
(960, 1280)  
Processing: IMG\_0168  
(960, 1280)  
Processing: IMG\_0169  
(960, 1280)  
Processing: IMG\_0170  
(960, 1280)  
Processing: IMG\_0171  
(960, 1280)  
Processing: IMG\_0172  
(960, 1280)  
Processing: IMG\_0173  
(960, 1280)  
Processing: IMG\_0174  
(960, 1280)  
Processing: IMG\_0175  
(960, 1280)  
Processing: IMG\_0176  
(960, 1280)  
Processing: IMG\_0177  
(960, 1280)  
Processing: IMG\_0178  
(960, 1280)  
Processing: IMG\_0179  
(960, 1280)  
Processing: IMG\_0180  
(960, 1280)  
Processing: IMG\_0181  
(960, 1280)  
Processing: IMG\_0182

```

(960, 1280)
Processing: IMG_0183
(960, 1280)
Processing: IMG_0184
(960, 1280)
Processing: IMG_0185
(960, 1280)
Processing: IMG_0186
(960, 1280)
Processing: IMG_0187
(960, 1280)
Processing: IMG_0188
(960, 1280)
Processing: IMG_0189
(960, 1280)
Processing: IMG_0190
(960, 1280)
Processing: IMG_0191
(960, 1280)
Processing: IMG_0192
(960, 1280)
Processing: IMG_0193
(960, 1280)
Processing: IMG_0194
(960, 1280)
Processing: IMG_0195
(960, 1280)
Processing: IMG_0196
(960, 1280)
Processing: IMG_0197
(960, 1280)
Processing: IMG_0198
(960, 1280)
Processing: IMG_0199
(960, 1280)

```

```
In [11]: !ls
```

```

000                                005                010_RGB_ALIGNED
000_IR-REDGE_ALIGNED  005_RGB_ALIGNED  Align_RGB_1.ipynb
000_RGB_ALIGNED      006                Align_RGB_2.ipynb
001                006_RGB_ALIGNED  Align_RGB_3_NIRRE.ipynb
001_RGB_ALIGNED      007                Align_RGB_4_NIRRE.ipynb
002                007_RGB_ALIGNED  IMG_0001_1.tif
002_RGB_ALIGNED      008                Untitled.ipynb
003                008_RGB_ALIGNED  Untitled1.ipynb
003_RGB_ALIGNED      009                diag.dat

```



```
004                                009_RGB_ALIGNED paramlog.dat
004_RGB_ALIGNED                   010
```

```
In [12]: ##Copy information from red images (*.3.tif) with Exiftool. I aligning using as a refer
         #If you align using another band as a reference you have to use the images with
         #the correct extension.
```

```
In [ ]: %%bash -s "$dir_set" "$dir_rgb"
```

```
dir=$(pwd)
```

```
dir_set=$1/
```

```
dir_rgb=$2
```

```
echo $dir_rgb
```

```
ls $dir_rgb*.JPG | xargs -n 1 basename > ${dir_rgb}rgb_files.txt
```

```
for f in `cat ${dir_rgb}rgb_files.txt`;
```

```
do
```

```
aa=$dir_set${f:0:8}_3.tif
```

```
bb=$dir_rgb$f
```

```
echo $aa
```

```
echo $bb
```

```
exiftool -overwrite_original_in_place -r -tagsFromFile $aa -gps:all -ExifIFD:all $bb;
```

```
done
```

```
In [14]: now = datetime.datetime.now()
```

```
print(now)
```

```
2018-05-29 18:29:54.182722
```