**Wheat Head Detection**

**Devendra Kumar Sahu**

TCR-Innovation Internship Final Project 1 (Wheat Head Detection)

**Source Code :-**

import os,ast

import pandas as pd

import numpy as np

from matplotlib import pyplot as plt

import matplotlib.patches as patches

import seaborn as sns

!ls global-wheat-detection

BASE\_DIR="global-wheat-detection"

%%time

train\_df = pd.read\_csv(os.path.join(BASE\_DIR, 'train.csv'))

sample\_sub\_df = pd.read\_csv(os.path.join(BASE\_DIR, 'sample\_submission.csv'))

train\_df.head()

sample\_sub\_df.head()

print(f'Shape of training data: {train\_df.shape}')

print(f'Shape of given test data: {sample\_sub\_df.shape}')

print(f'# of unique images: {train\_df["image\_id"].nunique()}')

print(f'Unique heights and widths: {train\_df["width"].unique()}, {train\_df["height"].unique()}')

print(f'Minimum number of wheat heads: {max(train\_df["image\_id"].value\_counts())}')

print(f'Minimum number of wheat heads: {len(train\_df)/train\_df["image\_id"].nunique()}')

print(f'Total number of images: {len(os.listdir(os.path.join(BASE\_DIR, "train")))}')

sns.distplot(train\_df['image\_id'].value\_counts(), kde=False)

plt.xlabel('# of wheat heads')

plt.ylabel('# of images')

plt.title('# of wheat heads vs. # of images')

plt.show()

train\_df[['x\_min','y\_min', 'width', 'height']] = pd.DataFrame([ast.literal\_eval(x) for x in train\_df.bbox.tolist()], index= train\_df.index)

train\_df = train\_df[['image\_id', 'bbox', 'source', 'x\_min', 'y\_min', 'width', 'height']]

train\_df

sns.distplot(train\_df['width'] \* train\_df['height'], kde=False)

plt.xlabel('Area of bbox')

plt.ylabel('# of images')

plt.title('Area of bbox vs. # of images')

plt.show()

fig, ax = plt.subplots(nrows=2, ncols=4, figsize=(20, 10))

count=0

for row in ax:

for col in row:

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", train\_df["image\_id"].unique()[count])}.jpg')

col.grid(False)

col.set\_xticks([])

col.set\_yticks([])

col.imshow(img)

count += 1

plt.show()

fig, ax = plt.subplots(nrows=2, ncols=4, figsize=(20, 10))

count=0

for row in ax:

for col in row:

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", train\_df["image\_id"].unique()[-count])}.jpg')

col.grid(False)

col.set\_xticks([])

col.set\_yticks([])

col.imshow(img)

count += 1

plt.show()

def get\_bbox(image\_id, df, col, color='white'):

bboxes = df[df['image\_id'] == image\_id]

for i in range(len(bboxes)):

rect = patches.Rectangle(

(bboxes['x\_min'].iloc[i], bboxes['y\_min'].iloc[i]),

bboxes['width'].iloc[i],

bboxes['height'].iloc[i],

linewidth=2,

edgecolor=color,

facecolor='none')

col.add\_patch(rect)

fig, ax = plt.subplots(nrows=2, ncols=2, figsize=(20, 20))

count=0

for row in ax:

for col in row:

img\_id = train\_df["image\_id"].unique()[count]

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", img\_id)}.jpg')

col.grid(False)

col.set\_xticks([])

col.set\_yticks([])

get\_bbox(img\_id, train\_df, col, color='red')

col.imshow(img)

count += 1

plt.show()

fig, ax = plt.subplots(nrows=2, ncols=2, figsize=(20, 20))

count=0

for row in ax:

for col in row:

img\_id = train\_df["image\_id"].unique()[-count]

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", img\_id)}.jpg')

col.grid(False)

col.set\_xticks([])

col.set\_yticks([])

get\_bbox(img\_id, train\_df, col,color='red')

col.imshow(img)

count += 1

plt.show()

image\_id = (train\_df['image\_id'].value\_counts() == 116).index[0]

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", image\_id)}.jpg')

fig, ax = plt.subplots(1, figsize=(12, 12))

ax.grid(False)

ax.set\_xticks([])

ax.set\_yticks([])

ax.axis('off')

get\_bbox(image\_id, train\_df, ax, color='red')

ax.imshow(img)

plt.plot()

all\_images = os.listdir(os.path.join(BASE\_DIR, 'train'))

all\_images = set([x[:-4] for x in all\_images])

images\_with\_bbox = set(list(train\_df['image\_id']))

images\_without\_bbox = list(all\_images - images\_with\_bbox)

print(f'Total number of images without wheat heads: {len(images\_without\_bbox)}')

fig, ax = plt.subplots(nrows=2, ncols=4, figsize=(20, 10))

count=20

for row in ax:

for col in row:

img\_id = images\_without\_bbox[count]

img = plt.imread(f'{os.path.join(BASE\_DIR, "train", img\_id)}.jpg')

col.grid(False)

col.set\_xticks([])

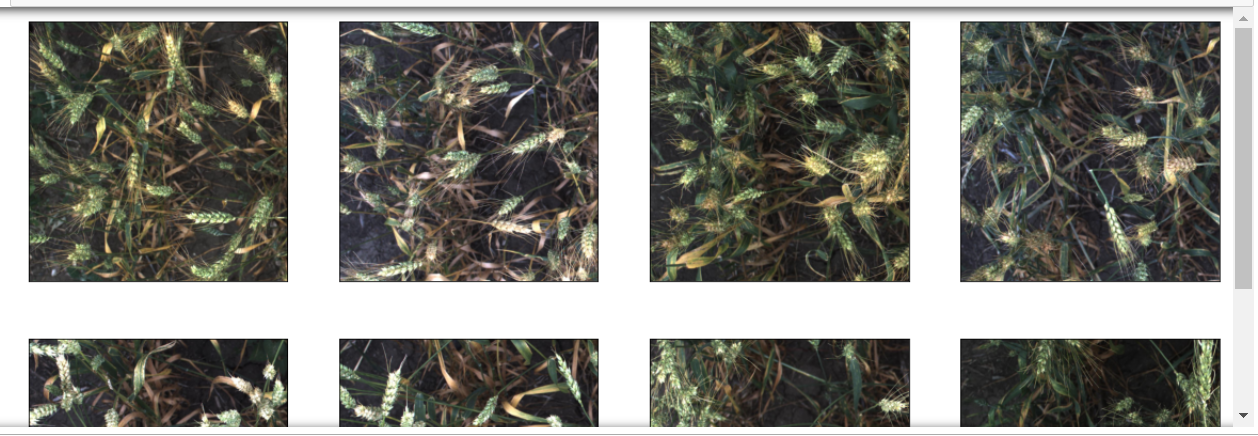
col.set\_yticks([])

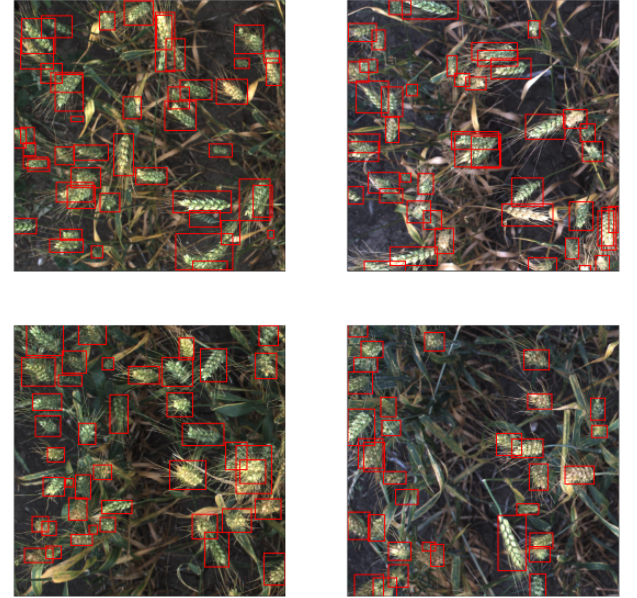
col.imshow(img)

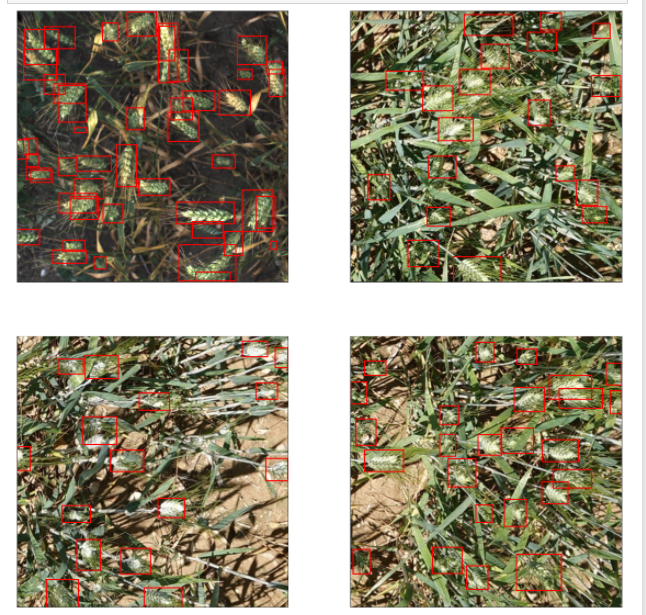
count += 1

plt.show()

**Output:-**

**Wheat Images**

**Wheat Detected Images**

****

****

**No wheat Detected**

****