# About

This tool is developed for wildlife detection. Currently, the tool is supported by Python3, Pytorch and Detectron2.

# Installation

To use the tool, please install the Detectron2 following the installation instructions.

You also need to install the following packages:

- opencv: pip install opencv
- ensemble-boxes: pip install ensemble-boxes
- scikit-imag: pip install scikit-imag
- imanticse: pip install imantics
- Pillow: pip install Pillow

# Usage

When using this tool run 'python main.py -h' or 'python main.py --help' to get the help information.

Currently the following parameters can be used:

| | |
|---|---|
| --ImgPath | The path of the image file. |
| --DirPath | The path of the image dir. |
| --Classification_Usage | If classification model will be used to help detection. |
| --WBF_Usage | If Weighted-Boxes-Fusion will be used to help detection. |
| --Output_Dir | Setup the path of output results you want to save. |
| --Draw_Detection | If generating images with detected animals. |

You must offer a valid image path or the path of a folder with images, currently we only support '.jpg' and '.JPG' files.

Example commands:

Basic running:

```
python main.py ––DirPath /Volumes/LIN/ANIMAL_DETECTION/DEMO_DATA
```

Running with draw detection output:

```
python main.py ––DirPath /Volumes/LIN/ANIMAL_DETECTION/DEMO_DATA ––
Draw_Detection True
```

Running with classification and WBF:

```
python main.py ––DirPath /Volumes/LIN/ANIMAL_DETECTION/DEMO_DATA ––
Draw_Detection True ––Classification_Usage True ––WBF_Usage True
```

Running with specific output folder:

```
python main.py ––DirPath /Volumes/LIN/ANIMAL_DETECTION/DEMO_DATA ––
Output_Dir /Volumes/LIN/ANIMAL_DETECTION/DEMO_OUTPUT/OUTPUT_TEST
```

# Output

## 1. Normal output json file

The default output file is ‘detection.json’ file:


```
{
  "Armadillo7": [
    {
      "id": 1,
      "bbox": [
        [
          0.2148,
          0.6778,
          0.4016,
          0.7958
        ]
      ],
      "class": "Armadillo",
      "conf": 0.9924
    }
  ],
  "Armadillo2": [
    {
      "id": 1,
      "bbox": [
        [
          0.2422,
          0.6505,
          0.4714,
          0.9097
        ]
```

```
        ],
        "class": "Armadillo",
        "conf": 0.9932
      }
   ]
}
```


## 2. CSV file with Detection result:

The program will also generate a csv file called ‘detection.csv’ with detection result:

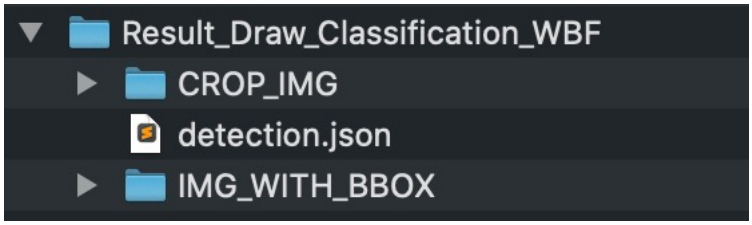
| | A | B | C | D | E |
|---|---|---|---|---|---|
| 1 | ImageName | bbox | class | conf | |
| 2 | Armadillo2 | [[0.2422, 0.6505, 0.4714, 0.9097]] | Armadillo | 0.9932 | |
| 3 | Armadillo6 | [[0.225, 0.5111, 0.6781, 0.8583]] | Armadillo | 0.9963 | |
| 4 | Bird12 | [[0.4828, 0.1292, 0.6953, 0.3806]] | Bird | 0.9969 | |
| 5 | Bird12 | [[0.332, 0.3611, 0.482, 0.7319]] | Bird | 0.9957 | |
| 6 | Bird25 | [[0.5982, 0.6594, 0.7461, 0.864]] | Bird | 0.9976 | |
| 7 | Deer14 | [[0.1388, 0.2814, 0.3187, 0.97]] | Deer | 0.9984 | |
| 8 | Deer14 | [[0.4338, 0.4878, 0.6325, 0.7824]] | Deer | 0.9971 | |
| 9 | Deer18 | [[0.0176, 0.1732, 0.7188, 0.9896]] | Deer | 0.997 | |
| 10 | | | | | |

## 3. Image with Detection result:

If ‘--Draw_Detection’ is set to ‘True’, there will be a folder called ‘IMG_WITH_BBOX’ with the images with detected bounding boxes on it.

## 4. Intermedia result (cropped images) for classification model:

If '--Classification_Usage is set to 'True', there will be a folder called 'CROP_IMG' with the cropped images detected by the detection model.




Example Cropped Image: