Experiment Report for Bird Nest Detection Using Detectron2

Md Saddam Hossain
Date: 2019.12.20

Train Phase: 01

- Bird nest train, test, val datasets size: 3419, 377, and 372 samples with less negative samples
- Training Parameters:
  - INPUT.MIN_SIZE_TRAIN 1200
  - INPUT.MAX_SIZE_TRAIN 2000
- FILTER_EMPTY_ANNOTATIONS option
  - FILTER_EMPTY_ANNOTATIONS False

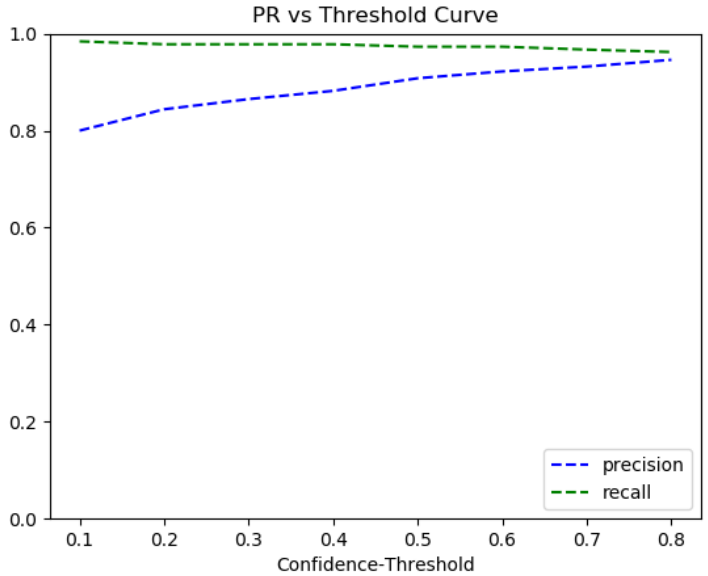
Phase: 01

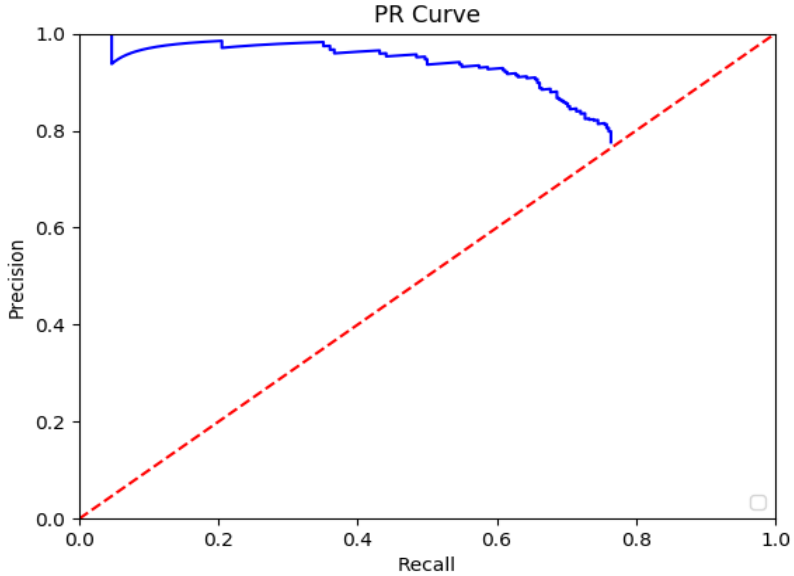
- After finishing initial training with 3419 samples along with above parameters,
- Evaluated the model with ***eiso_mondai*** test datasets and default confidence-threshold is 0.18,
- Model generates the following performances

    - Precision : 0.883
    - Recall : 0.989
    - AP :
    - AP01 : 96.5614
    - AP75 :

Phase: 01

- PR, PR vs Confidence-threshold curve for initial training phase.





Performance Observations and Design Planning for the next phase

- After completing initial training, It seems Precision is bit low compared to Recall. So we have a new challenge to increase the precision without reducing the Recall.
- We did a plan for increasing the precision value which was increase the dataset size. Specially, in negative samples. It seems that number of negative samples was less compared to positive samples.
- So initially, we collected, nearly +500 negative samples from google street view in Japan. And then start training for the next phase.
- And with using parameters, for training the model more time consuming. It takes nearly double training time compared to parameter-less training.
- So, for our next training phase, we did not use any parameters.

Train without parameters

- Parameterless training with initial datasets and using FILTER_EMPTY_ANNOTATIONS False option.
- We found the following performance for default conf-thres 0.18:
    - Precision : 0.718
    - Recall : 0.989
    - AP :
    - AP01 : 98.4789
    - AP75 :

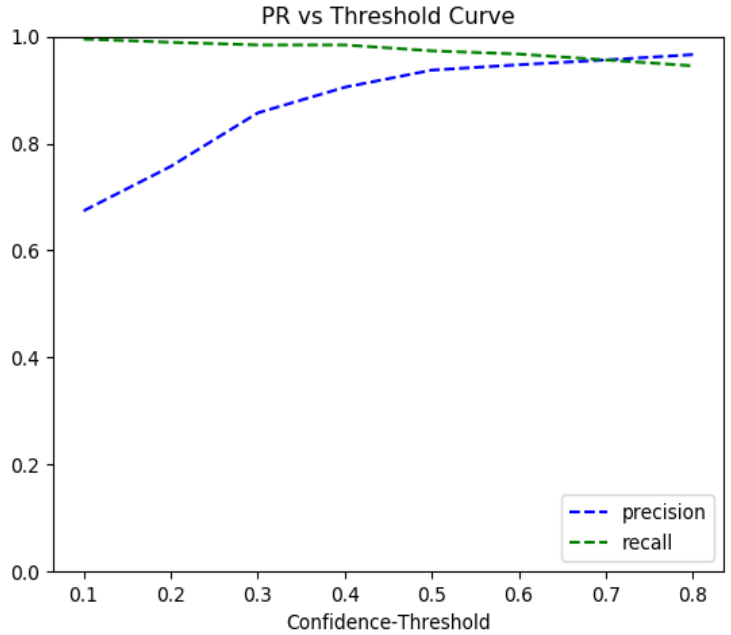
Train Phase: 02 with Additional 500 Negitive Samples

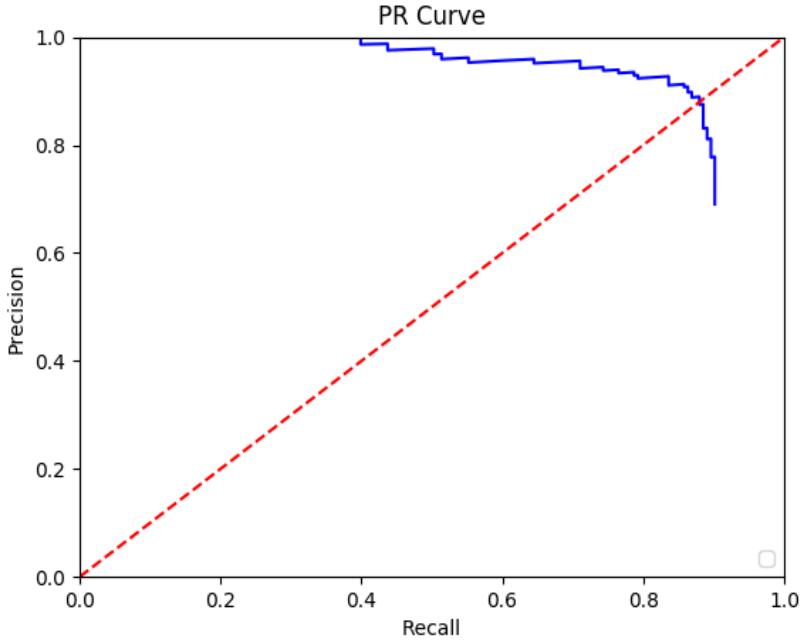
- Since using input size parameter, training is more time consuming. For this reason, we start training without parameter.
- After finishing the 2nd training phase with 3419 samples along with additional +500 negative samples without using any parameters,
- And FILTER_EMPTY_ANNOTATIONS False, model generates the following performances with default conf-thres 0.18

    - Precision : 0.748
    - Recall : 0.989
    - AP :
    - *AP01 :* 98.4031
    - AP75 :

Train Phase: 02

- PR. PR vs Conf-threshold curve for training phase 2.





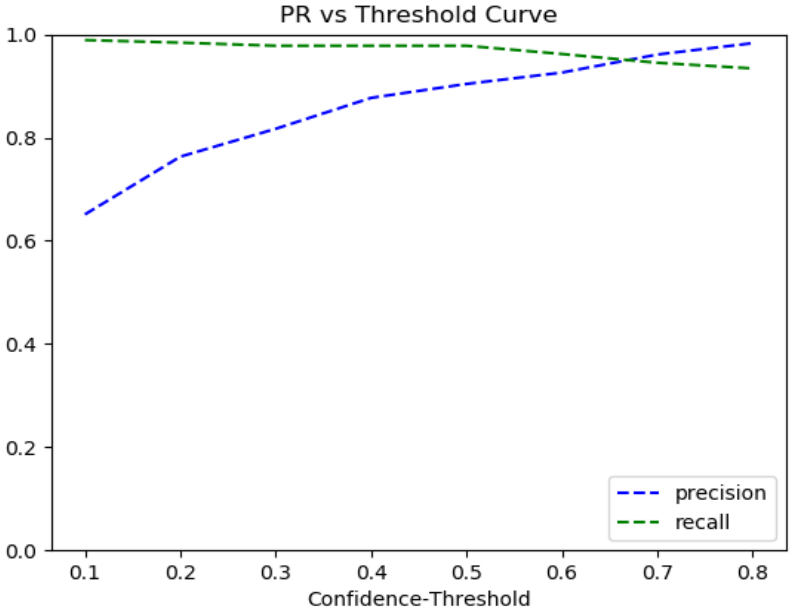
Train Phase: 03 with Additional 750 Negetive Samples

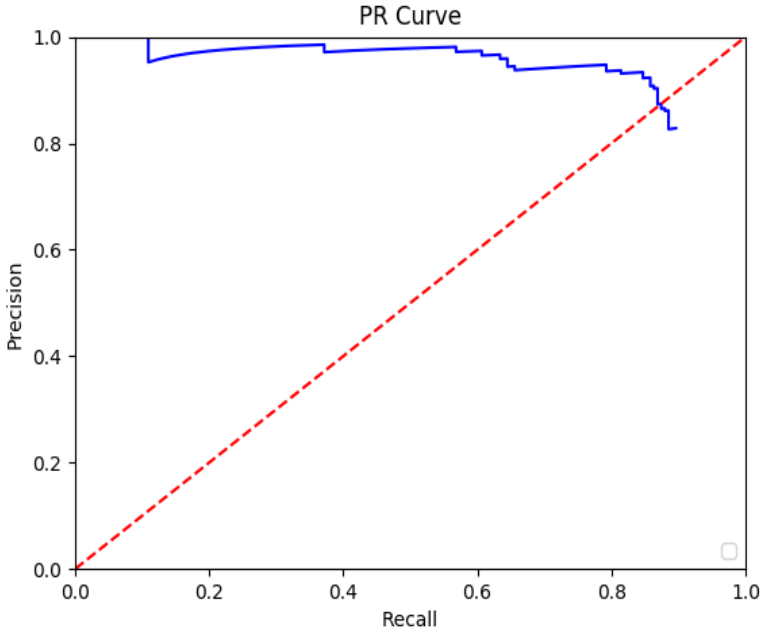
- Observing 2nd training phase it seems that little bit of improvement in AP01 values compared to initial training. So, we decided to add some more additional negative samples.
- After finishing the 3rd training phase with 3419 samples along with additional +750 negative samples without using any parameters
- And FILTER_EMPTY_ANNOTATIONS False, model generates the following performances

    - Precision : 0.767
    - Recall : 0.989
    - AP :
    - AP01 : 98.4659
    - AP75 :

Train Phase: 03

- PR, PR vs Confidence-threshold Curve for training phase 3.





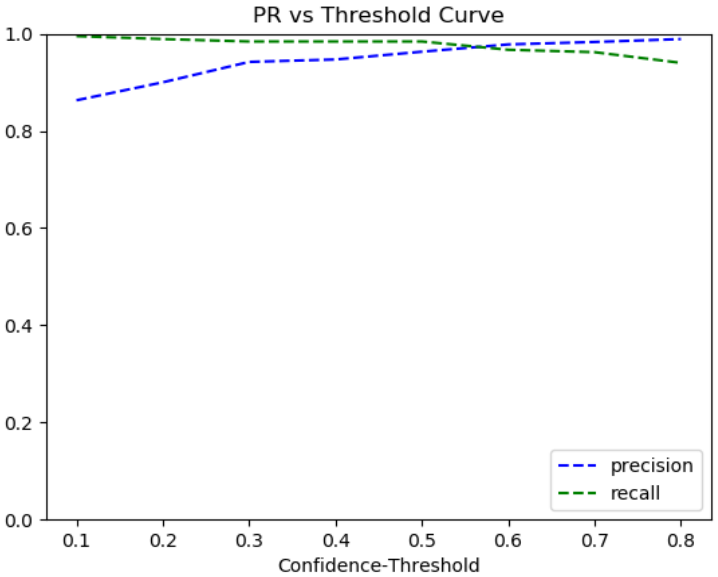
Train Phase: 04 with Additional 1000 Negetive samples

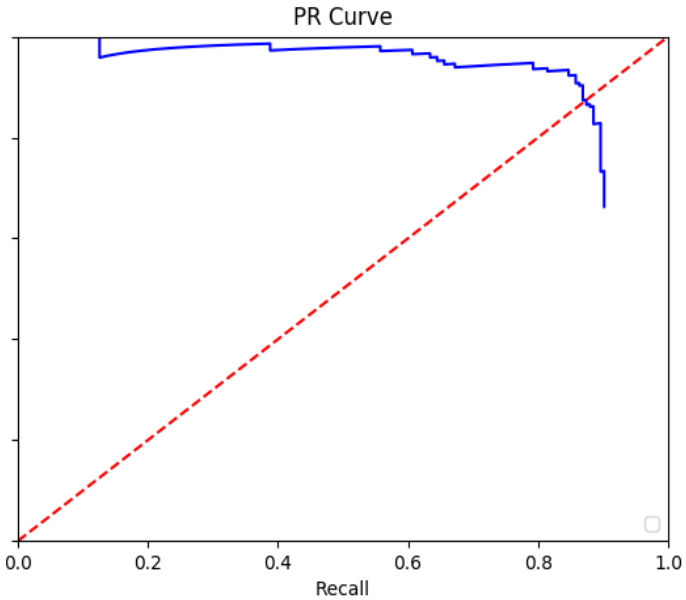
- In 3nd training phase, we observed that it seems that little bit of reducing in AP01 values compared to 2nd training phase. So, we decided to add some more additional negative samples,
- How the model actually reacts? So, we added nearly 1000 negative samples in total.
- After finishing the 4rth training phase with 3419 samples along with additional +1000 negative samples without using any parameters
- And FILTER_EMPTY_ANNOTATIONS False, model generates the following performances

    - Precision : 0.789
    - Recall : 0.984
    - AP :
    - AP01 : 97.7992
    - AP75 :

Phase: 04

- PR, PR vs Confidence-threshold Curve for training phase 4.





Train Phase :05 with additional 2890 Negative samples

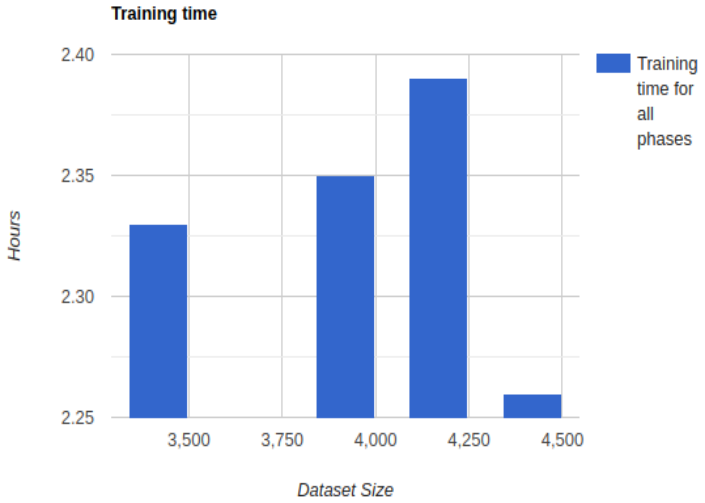
- In the above all training phases datasets were one sided. Positive samples was more more than negative samples. So we decided that we have to make the datasets balance. Then we collected nearly another 2000+ negative samples from google street view.
- So, after finishing the 5rth training phase with 2657 positive samples along with +2890 negative samples without using any parameters,
- And using filter option FILTER_EMPTY_ANNOTATIONS False, model generated the following performances

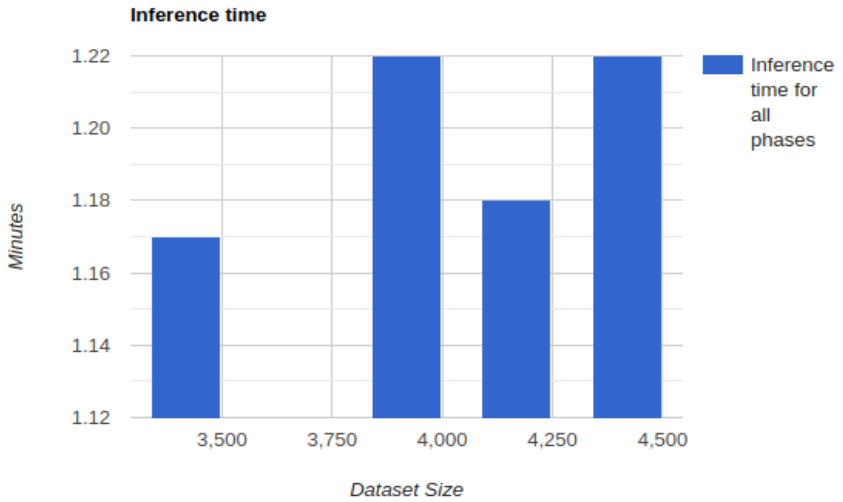
    - Precision : 0.773
    - Recall : 0.984
    - AP :
    - AP01 : 97.9730
    - AP75 :

Explanation of all Phases AP01

- In the above bar diagram, we observed that the more we add negative samples model improved its AP01’s performance. But improvement was a quite small.
- Though it showed a little bit decrease in training phase 3.
- But at final training stage, it improved AP01 value 98.32 to 99.24.

Training Vs Inference Time





Learning Take Aways in Detectron2 Implementation phase in Nest Detection

- I have learnt a bunch of concepts throughout this projects. Specially, object detection metrics and how they computed? More understanding about evaluation metrics.
- Learning more about how fine-tuning applied a model while training and evaluating the model.
- Experienced with preparing a bunch of preprocessing tools for the datasets.

Cont’d...

- And More clarification about version control management tools github. How to manage the task in github and how to write documentation in github wiki pages?
- Another most import thing is to learn is, remote development with docker and integrating it with vscode IDE.
- Learning a bit about docker image, docker container, and how to build a docker image by using dockerfile.