

DeepTour

Deep Learning for Tourism

New Pitch !

Flyers are the most common way to get information on the landmark you're visiting but you never know what to do with it afterwards

Wikipedia is weightless and more detailed but has less links to the actual landmark (Where is what ? What is what ?)

None of those are interactive

DeepTour provides an interactive solution with the best of both worlds

Our project

Using small context dependent CNNs, we provide an interactive experience for specific landmarks

- The application is built to be scalable and lightweight:
 - A data pipeline built on Google Cloud platform makes it extremely easy to add new models to the library
 - The landmark specific approach allows smaller training sets and models
 - Even devices with low camera resolution can run the app
- It respects your privacy:
 - The models size allows to run them offline without latency
 - The camera feed is not recorded



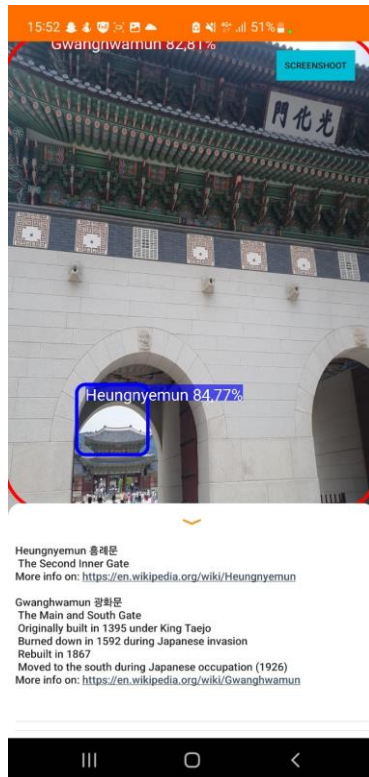
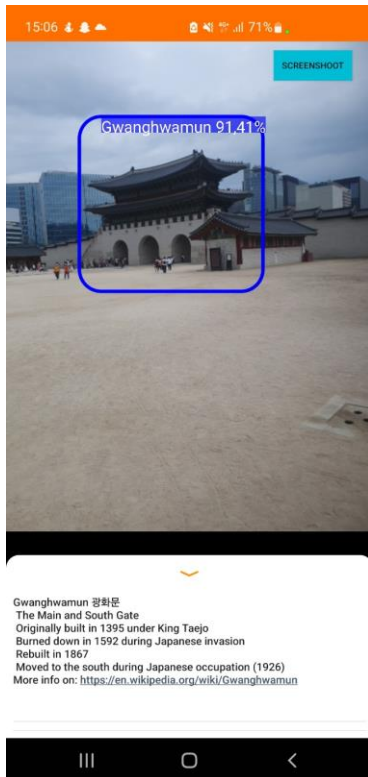
What's new

- Changed the context
 - Midterm remarks focused on the lack of context
 - What's the added value of object detection in this environment ?
- Changed the model
 - Prototype model (Yolo v3) compression did not work
 - Reduced similar models were still too heavy (10s latency, 80Mb)
- One solution to both problems
 - Localized models, specialized on one landmark
 - Implemented models:
 - Gyeongbokgung
 - Yeouido

Use case and demo

- Choose your model at the entrance of the landmark
- Walk around the area, looking for detectable objects/buildings to learn about the area
- Pause to read the quick description and click the link to learn even more







SCREENSHOOT

Geunjeongmun 80,47%



Gyeongbokgung

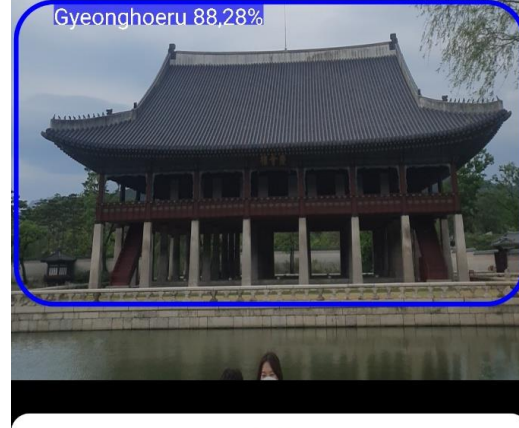
Geunjeongmun 근정문
The Third Inner Gate
More info on: <https://en.wikipedia.org/wiki/Geunjeongmun>



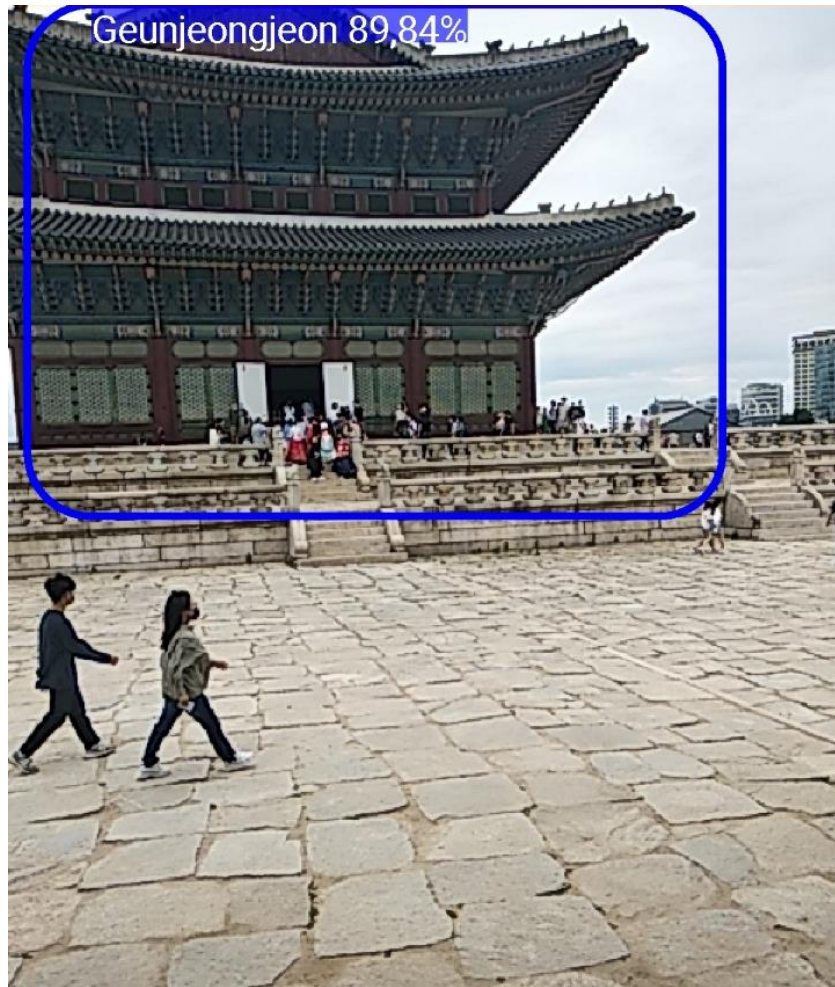
15:22 63%

SCREENSHOOT

Gyeonghoeru 88,28%



Gyeonghoeru 경회루
Royal Banquet Hall
Originally built in 1412 under King Taejong
Burned down in 1592 during Japanese invasion
Rebuilt in 1867
48 stone pillars
More info on: <https://en.wikipedia.org/wiki/Gyeonghoeru>



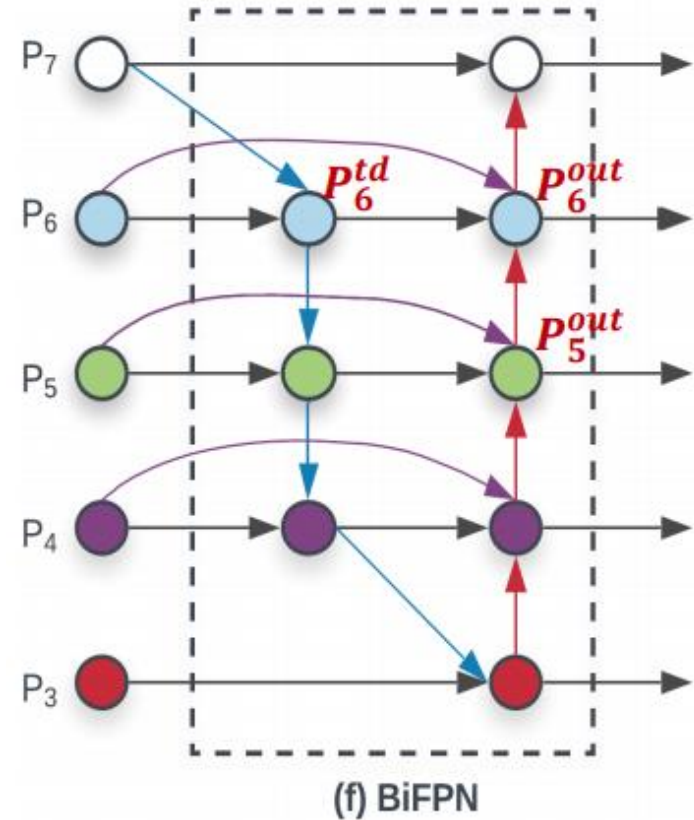
Technical review

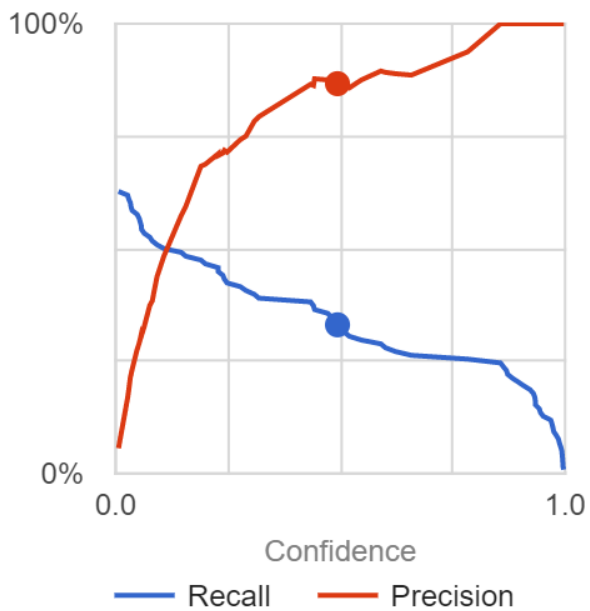
YOLOv3 did not work on the phone due to large latency

YOLOv3 Tiny had the same issue

Solution – EfficientDet (based on EfficientNet)

- Utilizes compound scaling - scaling each dimension by a constant ratio
- Apply bidirectional feature pyramid network (BiFPN)
- Learnable weights to determine the importance of different input features





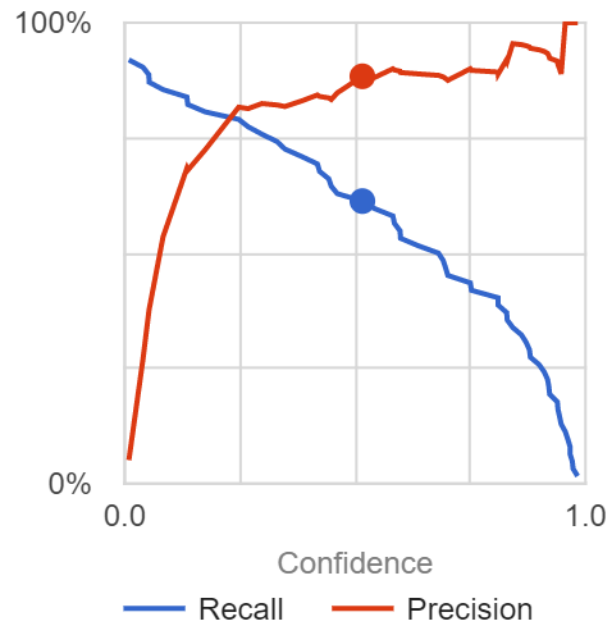
All labels

Total images	284
Test items	33
Total objects	118
Object to image avg	3.58
Precision ?	86.67%
Recall ?	33.05%

Gyeongbokgung

All labels

Total images	220
Test items	28
Total objects	62
Object to image avg	2.21
Precision ?	88.37%
Recall ?	61.29%



Yeouido

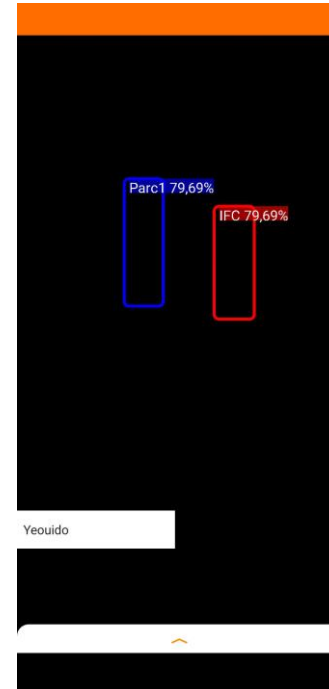
	Tasks	7 April	14 April	21 April	28 April	2 May	9 May	16 May	23 May	30 May	6 June	
	Search a dataset to train	YL, ND										Done
	Process dataset			YL, AP		Delayed						Done
Deep Learning (Python)	Build/Adapt a Neural Network model			AP, ND								Done
	Train the model				AP							Done
	Adapt the model for smartphone					AP, ND		Delayed				Done
App dev (Android, Java)	Setup retrieval of video flux			YL, AP								Done
	First UI design				YL, ND							Done
	Retrieve coordinate of the monument in the screen				YL, ND							Done
	Setup basic content generation (Layout on Android)				YL, ND							Done
	Intermediate presentation				Preparation							Done
	Feedback modifications from Intermediate report						YL, ND, AP					Done
	Final UI design						YL, ND					Done
	Expand model						YL, ND, AP					Done
	Evaluation/Test in real condition							YL, AP				Done
Optional	Participative training set								ND, AP			Cancelled
	360 Camera feed								AP, YL			Cancelled
	Screen guidance / recommandation								YL, ND			Cancelled
	Separate models (add-ons) for different Landmarks									YL, ND, AP		NEW (Done)
	User study							ND, AP				Done
	Final presentation									Preparation		Done

Notes and Discussion

- Roles and contributions
 - Data: Norman
 - ML: Andrei
 - Android: Yohan
 - Data and ML are upstream processes of the app development
 - Git used to store some data initially but the application itself has been mainly pushed by Yohan
- Application tested on 3 android phones
 - Different computational power and camera resolution (Galaxy S10+, Samsung A50, Redmi 5+)
 - Different android OS (8.1 to 12)
- Compatibility issues arose but the app is functional overall

Annexe

- Compatibility issues
- Screenshot authorizations



```
protected void setFragment() {  
    String cameraId = chooseCamera();  
  
    Fragment fragment;  
    if (useCamera2API) {  
        CameraConnectionFragment camera2Fragment =  
            CameraConnectionFragment.newInstance(  
                new CameraConnectionFragment.ConnectionCallback() {  
                    @Override  
                    public void onPreviewSizeChosen(final Size size, final int rotation) {  
                        previewHeight = size.getHeight();  
                        previewWidth = size.getWidth();  
                        CameraActivity.this.onPreviewSizeChosen(size, rotation);  
                    }  
                },  
                imageListener: this,  
                getLayoutId(),  
                getDesiredPreviewFrameSize());  
    }
```



```
@Override
public void onItemClick(AdapterView<?> parent, View view, int position, long id){
    String text = parent.getItemAtPosition(position).toString();
    if (position == 0){
        TF_OD_API_MODEL_FILE = "detect0.tflite";
    }
    else {
        TF_OD_API_MODEL_FILE ="detect1.tflite";
    };
    spinner_id = position;
    setFragment();
}
```

```
runOnUiThread(  
    new Runnable() {  
        @Override  
        public void run() {  
            labelNames= findViewById(R.id.label_info);  
            labelNames.setText(finalString);  
            showFrameInfo(previewWidth + "x" + previewHeight);  
            showCropInfo(cropCopyBitmap.getWidth() + "x" + cropCopyBitmap.getHeight());  
            showInference( inferenceTime: lastProcessingTimeMs + "ms");  
        }  
    });
```