

# Loading Time Project

I. Creating Loading time **detector** and **classifier**.

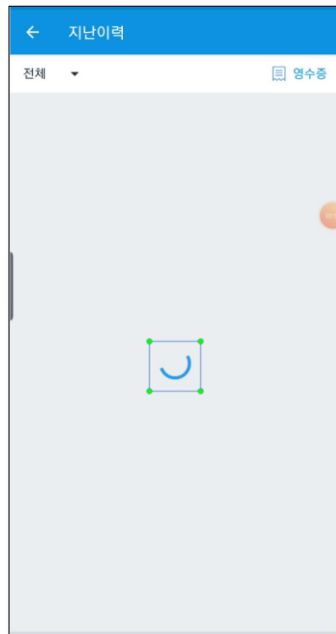
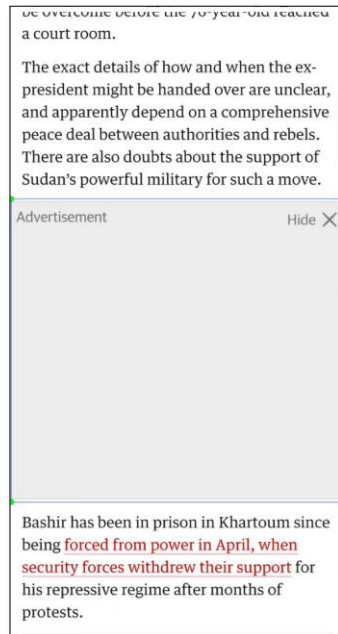
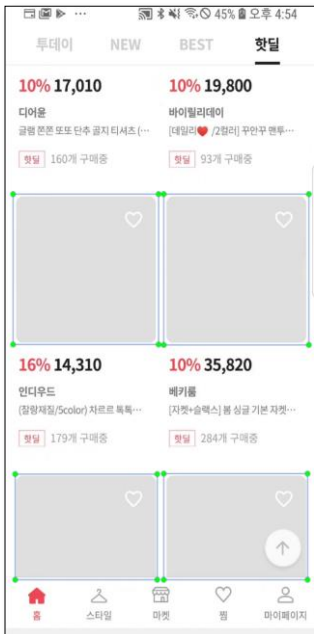
II. Creating Loading time API.

Step 1 > Collecting dataset: Take screenshots using **150** mobile apps

Step 2 > Labeling dataset using **LabelImg** tool (link: <https://github.com/tzutalin/labelImg>)

*class names* = [loading]

Loading image examples:



# Loading Time Project

Creating Loading time detector and classifier.

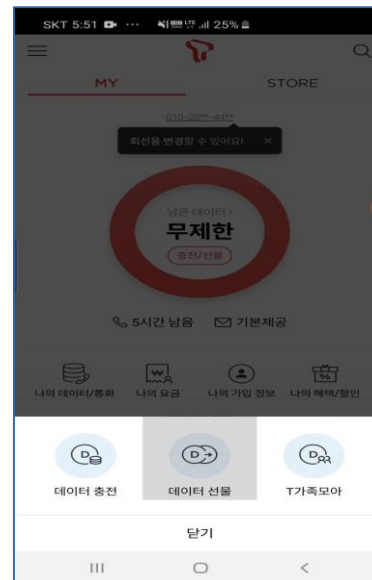
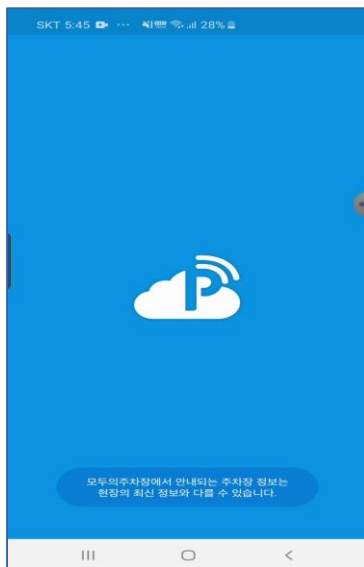
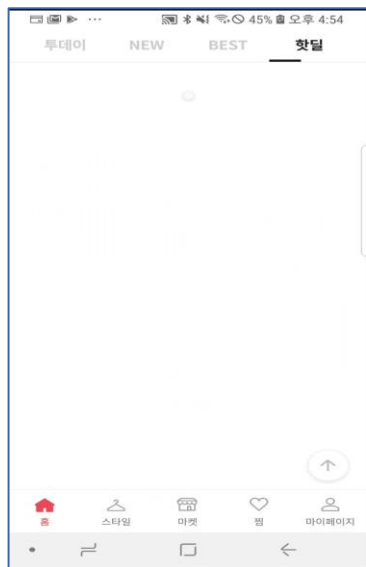
Creating Loading time API.

Step 1 > Collecting dataset: Take screenshots using **150** mobile apps

Step 2 > Labeling dataset using **Labellmg** tool (link: <https://github.com/tzutalin/labellmg>)

Step 3 > Sorting dataset for classification:

class names = [**loading**, **normal**]



# Loading Time Project : Loading Time API

---

Creating Loading time detector and classifier.

Creating Loading time API.

Step 1 > Collecting dataset: Take screenshots using **150** mobile apps

Step 2 > Labeling dataset using **LabelImg** tool (link: <https://github.com/tzutalin/labelImg>)

Step 3 > Sorting dataset for classification

Step 4 > **Making detection** and **classification model**:

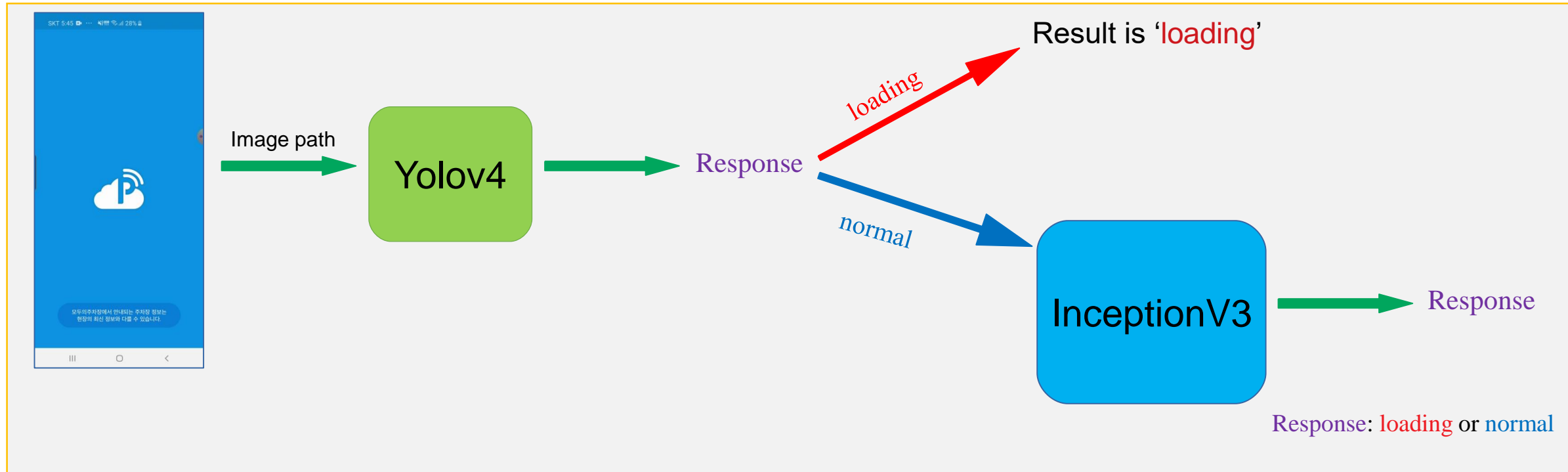
**Detection model**: Training dataset with **Yolov3** and **Yolov4** using Darknet (link: <https://github.com/AlexeyAB/darknet>)

**Classification model**: Training dataset with **Inception v3** pre-trained model (link: [https://tfhub.dev/google/imagenet/inception\\_v3/feature\\_vector/3](https://tfhub.dev/google/imagenet/inception_v3/feature_vector/3))

Step 5 > Creating loading time **API** using **Flask**

# Loading Time Project : Loading Time API

Working process of Loading Time API:



In the above architecture: Image will be sent to Yolov4, if the response from Yolov4 is '**loading**', *result is loading*.

If the response from Yolov4 is '**normal**', *the image* will be sent to **InceptionV3**. **InceptionV3** gives the final result.

Please enter to download video for knowing how **Loading Time API** works:

Video link : [https://github.com/aavuzb/project\\_videos/raw/main/Loading\\_Time\\_API.mp4](https://github.com/aavuzb/project_videos/raw/main/Loading_Time_API.mp4)