

Implementation notes

For detection ‘mmdetection’ framework was chosen. The instance of RTMDet architecture is used.

For API FastAPI framework is used via HTTP protocol.

Running examples

Running via uvicorn.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\user> cd desktop/apple_det/detection_app
PS C:\Users\user\desktop\apple_det\detection_app> uvicorn main:app --reload
INFO: Will watch for changes in these directories: ['C:\\Users\\user\\desktop\\apple_det\\detection_app']
INFO: Uvicorn running on http://127.0.0.1:8000 (Press CTRL+C to quit)
INFO: Started reloader process [23620] using StatReload
INFO: Started server process [31584]
INFO: Waiting for application startup.
INFO: Application startup complete.
```

Via Docker.

```
user@DESKTOP-T6QV8P4:/mnt/c/Users/user/desktop/apple_det/detection_app$ docker build -t apple_detection .
[+] Building 553.4s (19/19) FINISHED
=> [internal] load build definition from Dockerfile
=> transferring dockerfile: 976B
=> [internal] load .dockerignore
=> transferring context: 2B
=> [internal] load metadata for docker.io/library/python:3.8-slim
=> [internal] load build context
=> transferring context: 4.23kB
=> [ 1/14] FROM docker.io/library/python:3.8-slim@sha256:9a1e8d68615dd54b15889d57ae9232d6e696e7ad11353660e0f320f66d002f9b
=> CACHED [ 2/14] WORKDIR /app
=> [ 3/14] COPY . /app
=> [ 4/14] RUN apt-get update && apt-get install -y libgl1-mesa-glx
=> [ 5/14] RUN apt-get update && apt-get install -y libglu1-mesa-dev
=> [ 6/14] RUN pip install --upgrade pip
=> [ 7/14] RUN pip install torch==2.0
=> [ 8/14] RUN pip install fastapi
=> [ 9/14] RUN pip install uvicorn
=> [10/14] RUN pip install -U openmim
=> [11/14] RUN pip install python-multipart
=> [12/14] RUN mim install mmdet
=> [13/14] RUN mim install "mim==2.0.0"
=> [14/14] RUN mim install mmdet
=> exporting to image
=> exporting layers
=> writing image sha256:de6797ca2439c3627ea9563d6afecdc2ed09087131fd3acc4d43eddcf221411
=> naming to docker.io/library/apple_detection

What's Next?
View a summary of image vulnerabilities and recommendations → docker scout quickview
user@DESKTOP-T6QV8P4:/mnt/c/Users/user/desktop/apple_det/detection_app$ docker run -d --name mycontainer -p 80:80 apple_detection
81c718dcaa40a76912fe29adaca3a711b1e6a174024f07909d562b55834c1e02
```

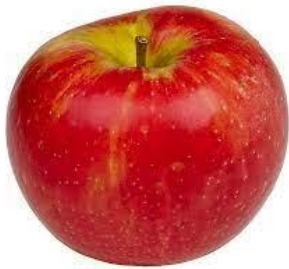
Welcome to the Start Page of Apple Detection Application

Choose a file:

To test detection functionality three examples were used.

1. One apple detection.

Initial picture:



Result:

Welcome to the Start Page of Apple Detection Application

Choose a file:



2. No apples detection.

Initial picture:



Result:

Welcome to the Start Page of Apple Detection Application

Choose a file:

There is no APPLE! At least I don't see it...

3. Many apples detection

Initial picture:



Result:

Welcome to the Start Page of Apple Detection Application



It's obvious that algorithm should be tuned to distinguish separate apples.