

BAB IV

HASIL PERANCANGAN DAN PENGUJIAN ALAT

Pada bab berikut ini akan dibahas mengenai langkah-langkah pengujian serta hasil yang didapatkan dari uji coba Rancang Bangun Sistem Elektronik Tilang Ganjil Genap Berbasis Raspberry Pi Menggunakan Algoritma Yolo.

4.1 Hasil Perancangan

Pada bagian ini akan membahas hasil dari penerapan teori menjadi sebuah sistem yang dapat digunakan dengan baik. Rangkaian yang terdiri dari *Raspberry* terkoneksi dengan *webcam camera*. Untuk *main power* didapatkan dari *power bank*. Untuk koneksi internet dari *hotspot wifi*.

Berikut merupakan hasil perancangan yang dapat dilihat pada gambar 4.1 dibawah ini.



Gambar 4.1 Hasil Perancangan Camera Card Box

4.2 Pengoperasian alat

Berikut merupakan cara-cara untuk mengaktifkan sistem:

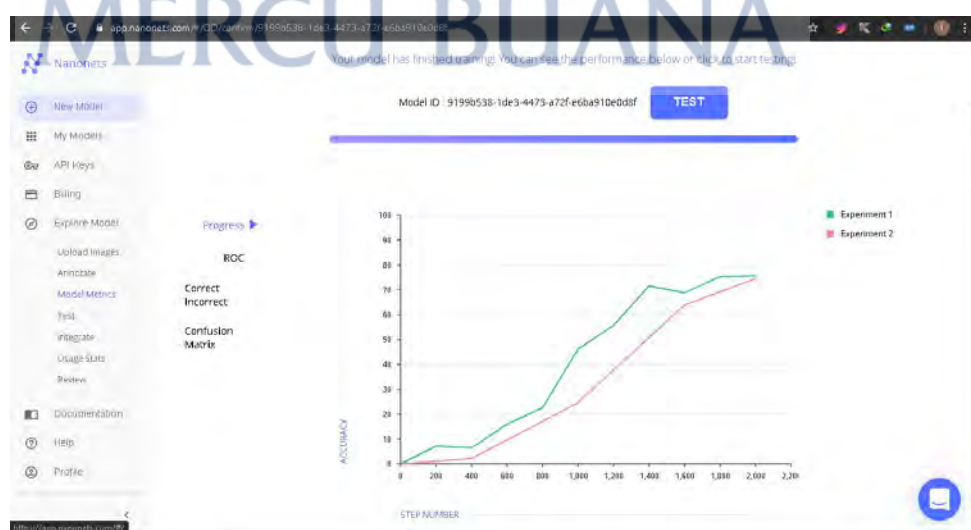
1. Menghubungkan alat dengan sumber tegangan (*power bank* 10.000 mAH).
2. Mengaktifkan *hotspot* wifi.
3. Me-remote *raspberry pi* yang sudah terkoneksi dengan jaringan wifi.
4. *Usb camera* yang tersambung pada *raspberry pi* otomatis akan mengirimkan gambar pada server.
5. Server mampu mendeteksi gambar, apabila terdeteksi maka gambar akan *terbounding*.

4.3 Pengujian Alat

Pada pengujian alat, akan menampilkan *progress* dari *training dataset*, ROC (*Receiver operating characteristics*), prediksi gambar benar dan salah serta *confusion matrix* pada nanonets.

4.3.1 Progress datasets

Progress ini menunjukkan seberapa banyak step yang dilakukan berbanding dengan tingkat akurasi untuk mendeteksi. Yang ditunjukkan pada gambar 4.2 dibawah ini.

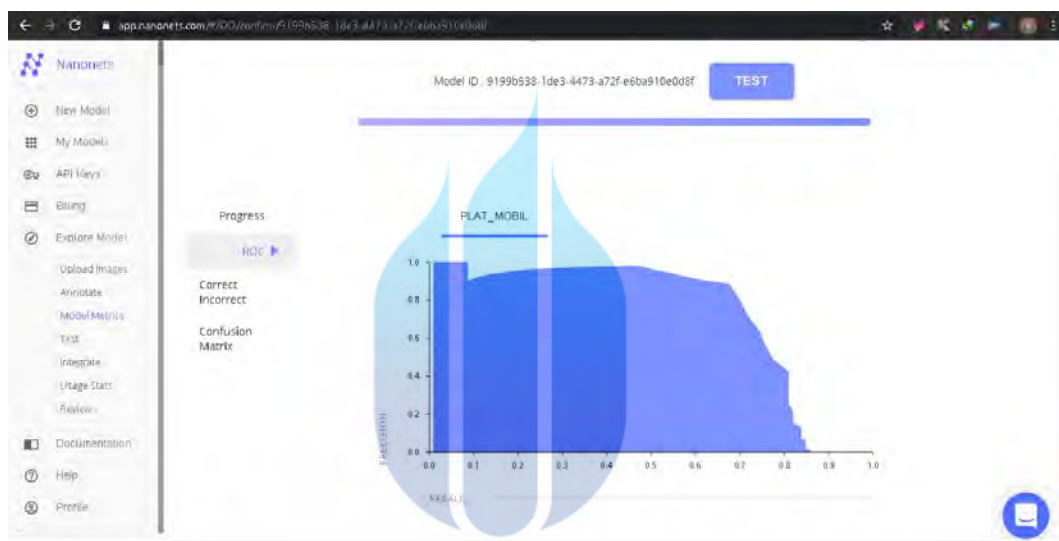


Gambar 4.2 Progress Training Datasets.

Seperti terlihat pada gambar eksperimen 1 dan 2, semakin banyak step yang dijalankan maka tingkat akurasi semakin tinggi. Puncak tertinggi di dapat setelah *me-running step* selama 2000 *step* yang mendapat tingkat akurasi sebesar 75,67%.

4.3.2 ROC (Receiver Operating Characteristics)

Nilai ROC adalah semacam alat ukur *performance* untuk *classification problem* dalam menentukan *threshold* dari suatu model. Berikut merupakan hasil ROC pada dataset setelah di *training* yang terdapat pada gambar dibawah ini.



Gambar 4.3 Nilai ROC.

Kurva ROC sendiri dibuat berdasarkan nilai yang telah didapatkan pada perhitungan *confusion matrix*, yaitu antara *false positive rate* dengan *true positive rate*. Dimana:

1. *False positive rate* (FPR) = *False positive* / (*false positive* + *true negative*).
2. *True positive rate* (TPR) = *True positive* / (*true positive* + *false negative*).

Untuk membaca nilai kurva, kinerja algoritma klasifikasi adalah :

1. Buruk, jika kurva yang dihasilkan mendekati garis baseline atau garis yang melintang dari titik 0,0.
2. Baik, jika kurva mendekati titik 0,1.

Yang mana nilai *precision* dan *recall* ROC akan ditampilkan pada tabel dibawah ini.

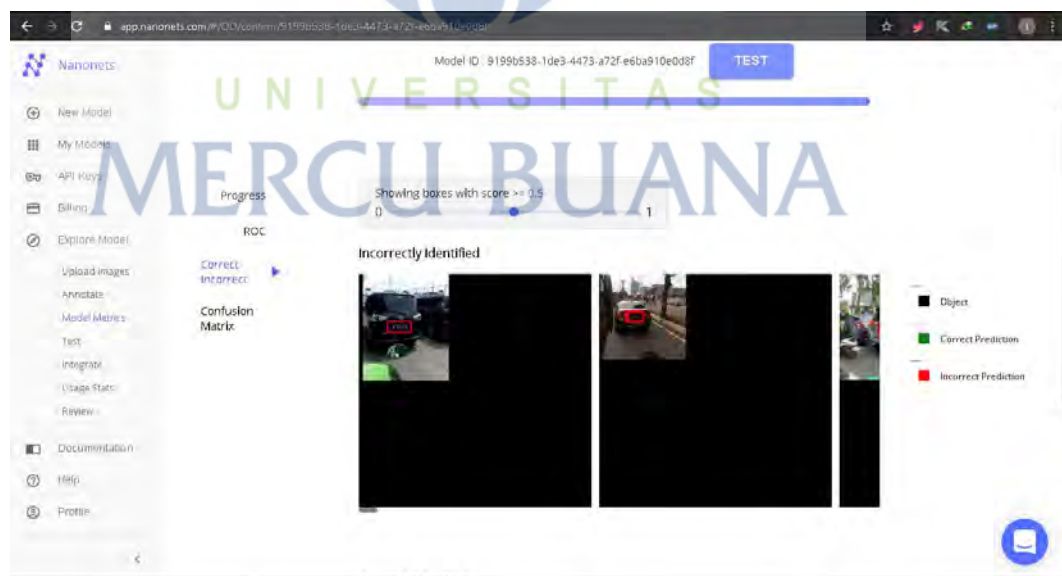
Tabel 4.1 Nilai Precision, Recall dan Score.

No	Precision	Recall	Score
1	0,666	0,019	0,949
2	0,966	0,285	0,880
3	0,918	0,323	0,872
4	0,934	0,592	0,759
5	0,751	0,754	0,621

Seperti terlihat pada tabel diatas, selama pengujian tingkat ROC awalnya cukup baik dengan *score* hingga 0,949 tapi terus menurun hingga *score* 0,621.

4.3.3 Incorrect and Correct Prediction

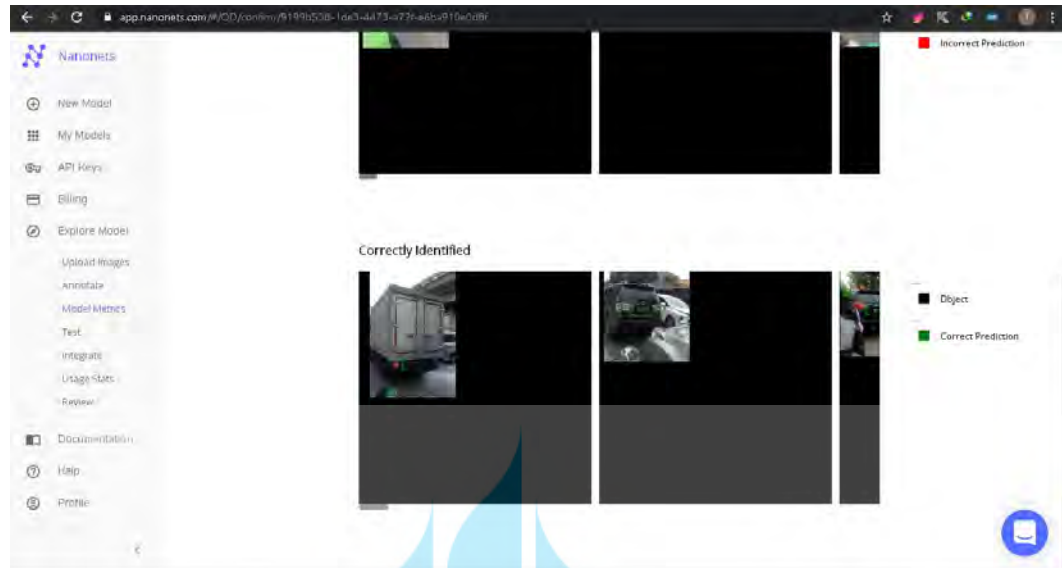
Dibawah ini merupakan *incorrect and correct prediction with showing boxes with score ≥ 0.5* .



Gambar 4.4 Incorrect Prediction

Pada gambar diatas menunjukkan *incorrect prediction* dimana boxes memiliki nilai lebih dari 0.5 atau tidak terdeteksi atau tidak terprediksi sama sekali.

Dibawah ini merupakan gambar dari *correct prediction*.

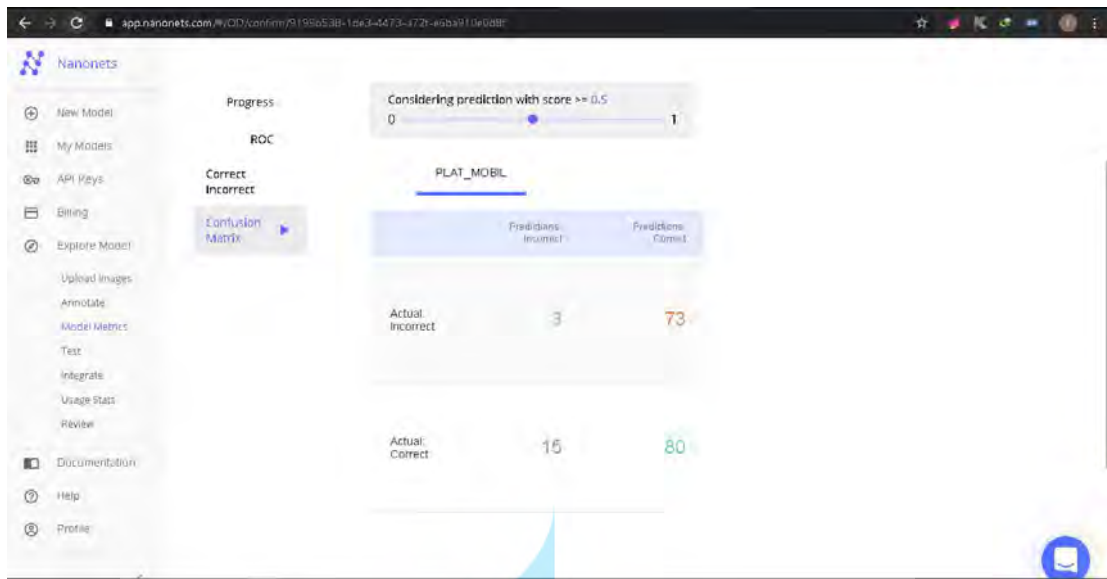


Gambar 4.5 Correct Prediction.

Terlihat pada gambar diatas apabila prediksi berhasil maka ditandai dengan box berwarna hijau, dimana ini menandakan bahwa gambar terdeteksi dan nilai boxes 0.5 atau sesuai dengan keinginan.

4.3.4 Confusion Matrix

Pada dasarnya *confusion matrix* mengandung informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem dengan hasil klasifikasi yang seharusnya. Hasil dari confusion matrix dapat dilihat pada gambar dibawah ini.



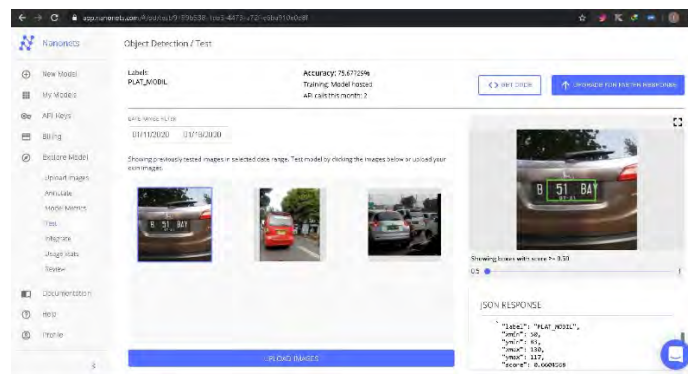
Gambar 4.6 Confusion Matrix

Pada confusion matrix terlihat bawah dari 2 experiment menunjukkan nilai:

1. Eksperimen pertama : *prediction incorrect* dengan nilai 8 dan *prediction correct* dengan nilai 58.
2. Eksperimen kedua : *prediction incorrect* sebesar 7 dan *prediction correct* sebesar 82.

4.3.5 Hasil Test

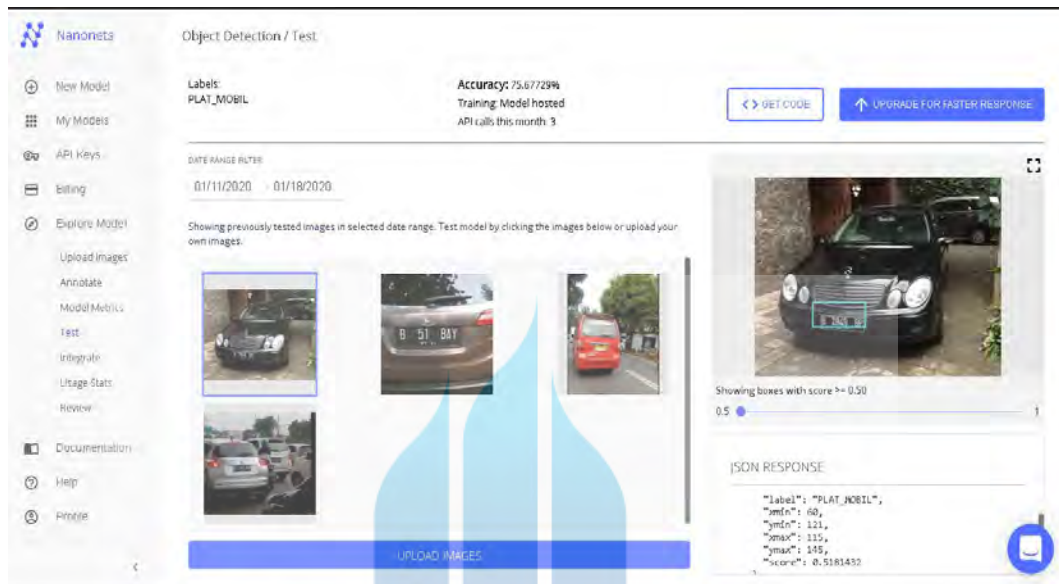
Dibawah ini merupakan hasil prediksi beserta code dari gambar-gambar yang dijadikan model test



Gambar 4.7 Test Pertama

pada gambar diatas, *score* didapat 0,680. Gambar dapat terdeteksi tetapi tidak mendeteksi keseluruhan plat (dengan *showing boxes score* ≥ 0.50).

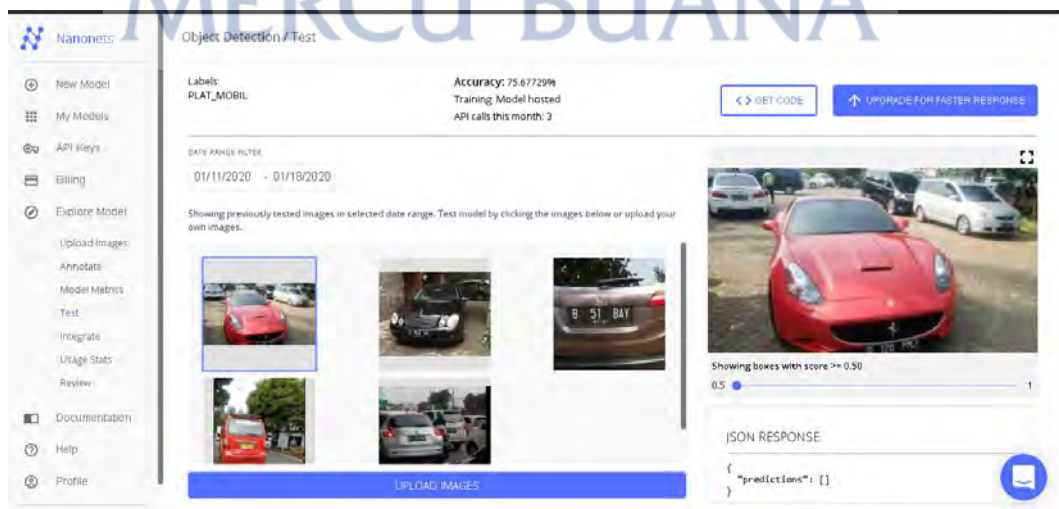
hasil test kedua dapat dilihat pada gambar dibawah ini.



Gambar 4.8 Test Kedua

Pada gambar kedua, *prediction box* mendapatkan nilai 0,518 dimana ini mendekati nilai *showing box* yang di set sebesar 0.50.

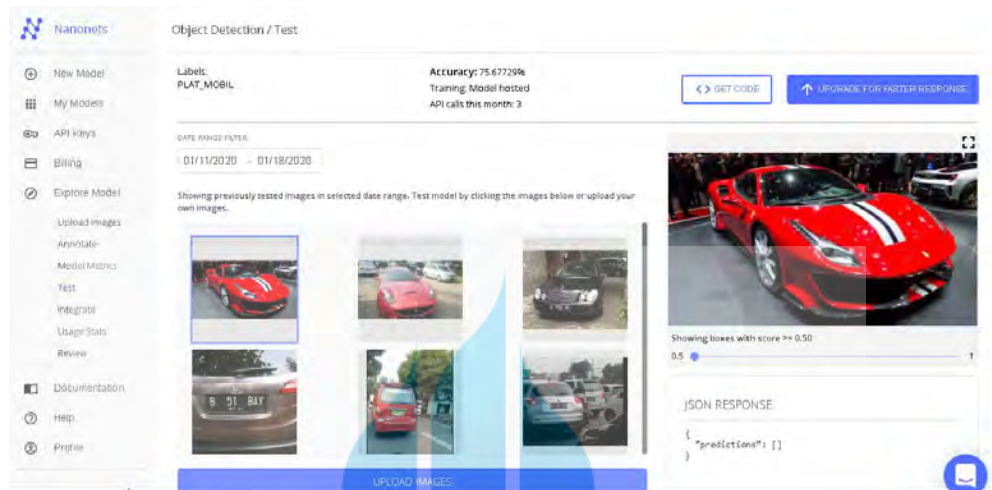
Test ketiga dapat dilihat pada gambar dibawah ini.



Gambar 4.9 Test Ketiga

Pada gambar test ketiga, *showing box* tidak dapat mendeteksi plat mobil sama sekali.

Test keempat menggunakan gambar mobil yang tidak memiliki plat dapat dilihat pada gambar dibawah ini.

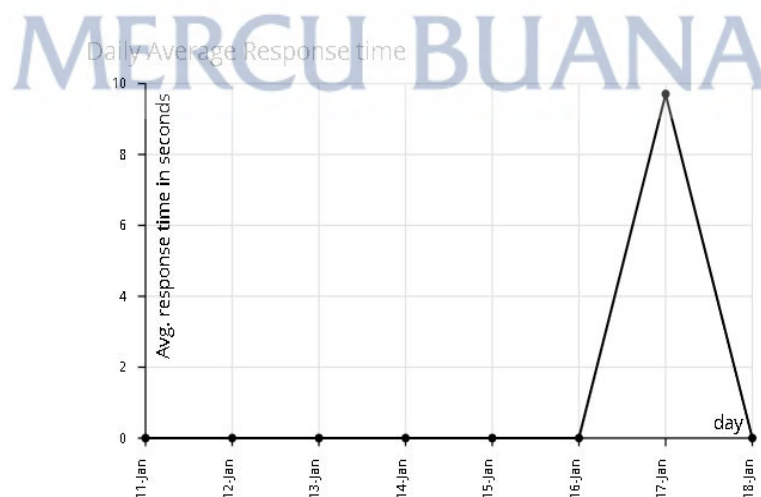


Gambar 4.10 Test Keempat

Pada test keempat membuktikan bahwa dataset hanya dapat mendeteksi gambar mobil yang memiliki plat mobil sesuai dengan label yang sudah ditentukan.

4.3.6 Hasil Grafik Response, Image Processed, Labelwise and Confidence

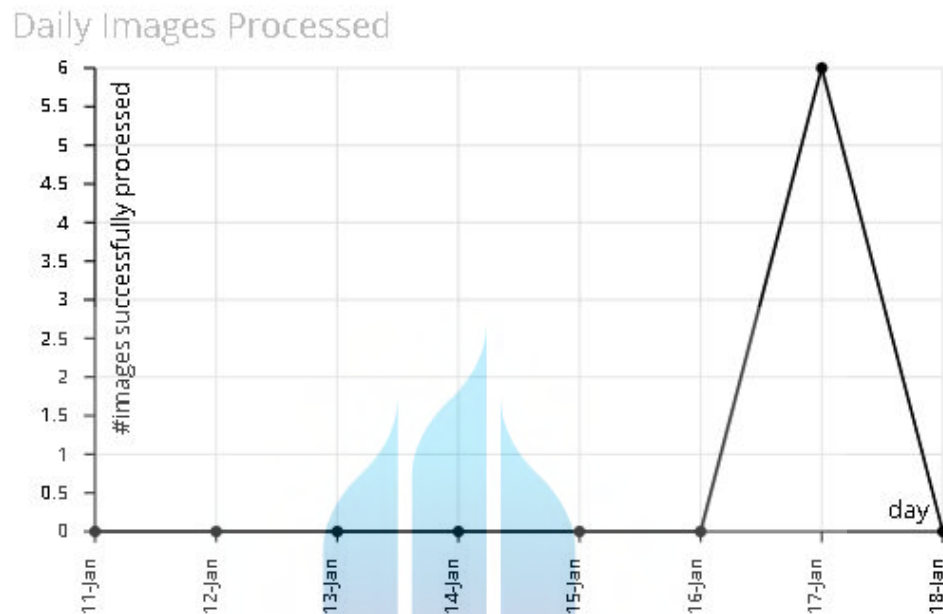
Dibawah ini merupakan gambar dari *daily average response time*.



Gambar 4.11 Grafik Daily Average Response Time.

Dimana dapat dilihat bahwa rata-rata respon mendekati 10 detik.

Daily image processed dapat dilihat pada gambar dibawah ini.



Gambar 4.12 Daily Image Processed.

Dimana ini menunjukkan bahwa perkiraan gambar sukses terproses sebesar

6.

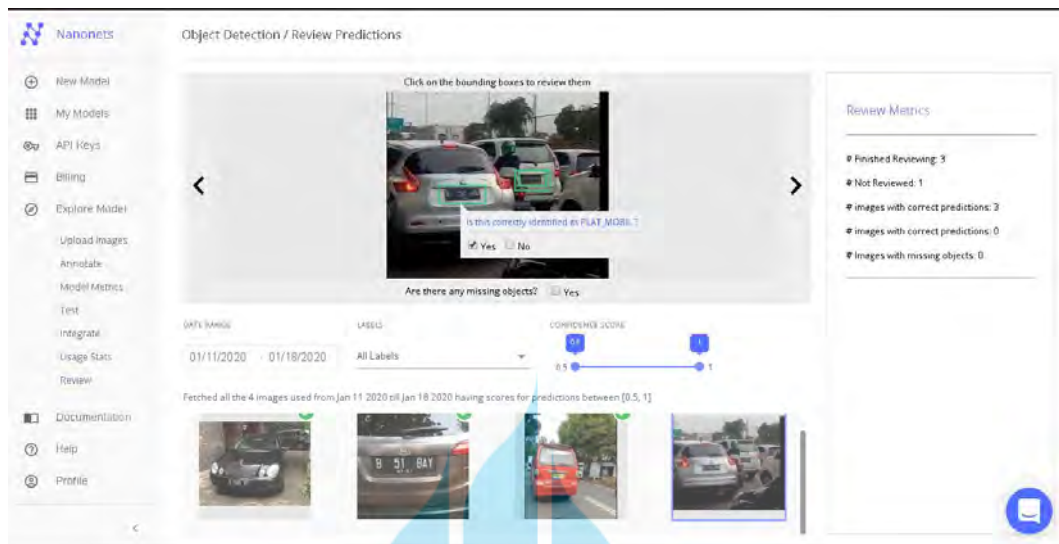
Gambar dibawah ini menunjukkan *daily labelwise average confidence*.



Gambar 4.13 Daily Labelwise Average Confidence

4.3.7 Review

Dibawah ini menunjukkan hasil review dari program ini.



Gambar 4.14 Review

Pada nanonets API ini dari 4 kali test, 2 diantaranya mendeteksi dengan baik. 1 gambar melewati *showing box* yang ditentukan dan 1 gambar tidak dapat mendeteksi keseluruhan plat mobil.

UNIVERSITAS
MERCU BUANA