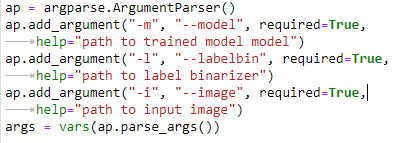


Pertama kita mengimpor paket yang diperlukan

  
Lalu parsing argumen dengan kode diatas, seperti yang tertera diatas, ada tiga argumen baris perintah yang kita perlu parsing, diantaranya:

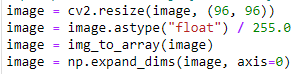
--model : yang akan menunjukkan ke file direktori dimana model kita di training sebelumnya

--labelbin : yang akan menunjukkan ke file direktori binarizer label berada.

--image : yang akan menunjukkan file direktori dari gambar yang dimasukkan.



Disini kita memuat gambar masukkan, dan membuat salinan / copy dari gambar masukkan tersebut untuk ditampilkan Kembali nantinya bersamaan dengan hasil klasifikasi

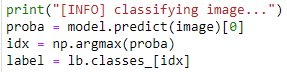


Kemudian kita melakukan *preprocess* gambar dengan cara yang sama persis seperti yang kita lakukan pada saat training / pelatihan.

Selanjutnya, kita memuat/load model + label binarizer dan kemudian mengklasifikasikan gambar:



Untuk mengklasifikasikan gambar, kita membutuhkan model dan label binarizer di memori, sehingga kita perlu memuatnya terlebih dahulu dengan kode diatas

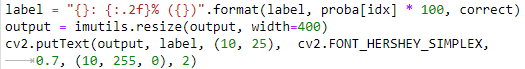


Selanjutnya, kita mengklasifikasikan gambar dan membuat label



Kita mengekstrak nama Thermal dari nama file dan membandingkannya dengan label . Variabel yang benar akan menjadi "correct" atau "incorrect" berdasarkan tahap ini

Kedua baris ini yang akan membuat asumsi bahwa gambar yang kita input memiliki nama file yang berisi label sebenarnya.



Dari kode diatas akan menjalankan langkah-langkah berikut:

Label = Menambahkan persentase probabilitas dan teks "correct" / "incorrect"

Output = Mengubah ukuran gambar output agar sesuai dengan layar kita

Cv2.putText = Membuat teks label pada gambar keluaran



Terakhir menampilkan gambar setelah proses pengklasifikasian selesai. Maka hasil keluaran / output dari gambar masukkan terbagi menjadi 2 sesuai dengan kasus yang dibutuhkan seperti pada gambar berikut :

