**LAPORAN**

**PELABELAN DATA**

**UNTUK TRAINING MACHINE LEARNING PENDETEKSI API**

OLEH:

SAHRUL

SABILA HADINISA

**1. PENDAHULUAN**

Training dataset dalam machine learning merupakan kumpulan data aktual yang digunakan untuk melatih program atau algoritma machine learning agar dapat melakukan tindakan tertentu. Jumlah training dataset yang dibutuhkan agar program atau algoritma machine learning dapat bekerja dengan efektif harus dalam jumlah yang besar.

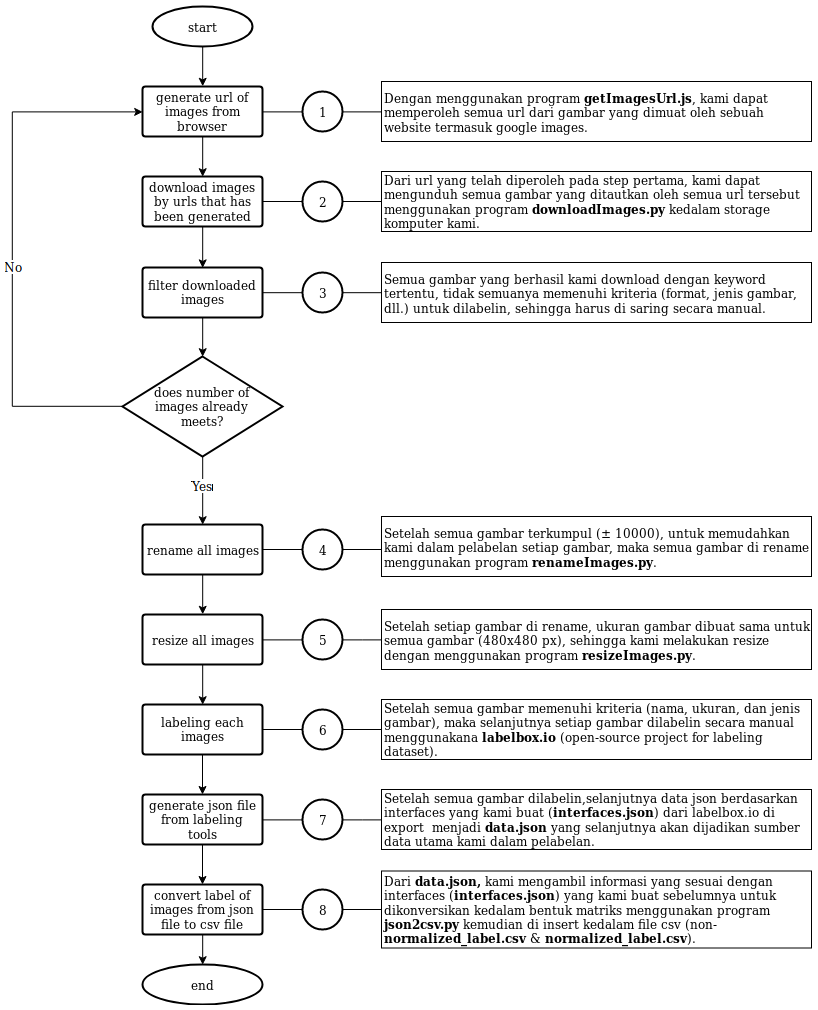
Untuk memudahkan pengerjaan, dibutuhkan program yang dapat menangani tugas-tugas yang repetitif untuk setiap data. Beberapa pekerjaan repetitif yang akan ditangani oleh program seperti mengunduh data dari internet dan pengolahan data itu sendiri. Karena training dataset yang dibutuhkan berupa gambar, maka pengolahan data yang dimaksud adalah seperti resize, rename, dan beberapa proses lainnya.

Proses utama dalam proyek ini adalah mengidentifikasi letak objek api dalam setiap gambar dari training dataset. Proses ini dilakukan secara manual dengan menggunakan situs pelabelan Labelbox.io kemudian data keluarannya diolah menjadi label setiap gambar dengan sebuah program. Proses yang dilakukan dalam proyek ini hanya sampai pada tahap pelabelan setiap data setelah letak objek api diidentifikasi.

Label setiap gambar memuat informasi tentang gambar dan lokasi objek api pada setiap gambar. Hasil akhir dari informasi tersebut berupa matriks yang memuat informasi dari setiap gambar tersebut.

**2. HASIL DAN PEMBAHASAN**

**2.1. Flow Chart**



**2.2. Deskripsi Program**

Tabel 2.1. Deskripsi setiap program yang digunakan.

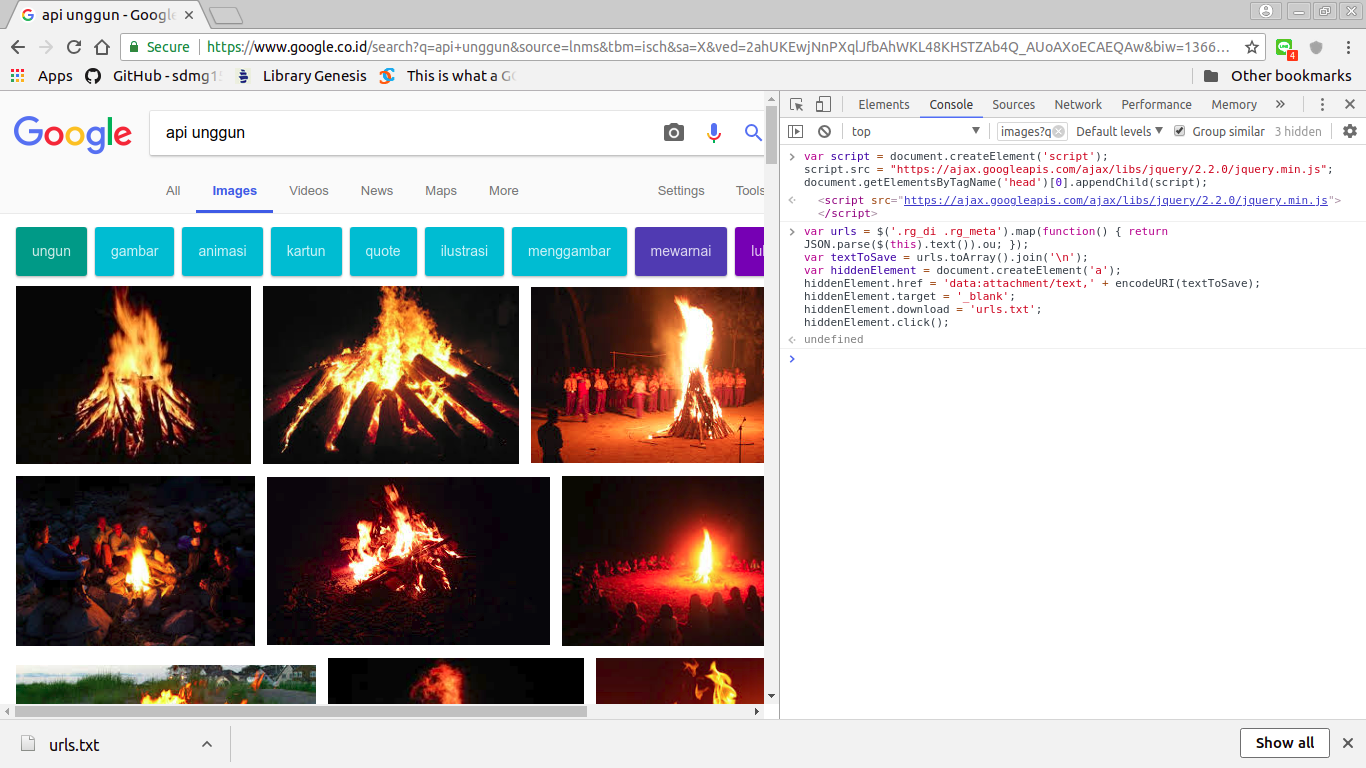
|  |  |  |
| --- | --- | --- |
| Program / Data / Tools | Fungsional | Fungsi Utama |
| getImagesUrl.js | 1. Melakukan manipulasi objek pada browser dengan menggunakan jquery framework. 2. Menyaring semua url dari gambar untuk diunduh sebagai file. | Memperoleh semua url dari gambar yang dimuat oleh website (google images). |
| downloadImages.py | 1. Mengubah file urls.txt menjadi python list dengan elemen setiap url itu sendiri. 2. Melakukan request pada setiap elemen pada python list untuk mengunduh gambar yang ditautkan pada url tersebut. | Mengunduh setiap gambar yang ditautkan yang ditautkan oleh setiap url. |
| renameImages.py | 1. Membaca setiap file dalam sebuah directory yang akan di-rename. 2. Melakukan rename pada setiap images secara incremental (by index) dan menulisnya kembali kedalam sebuah directory baru. | Mengubah nama setiap gambar agar lebih mudah diidentifikasi. |
| resizeImages.py | 1. Membaca setiap file dalam sebuah directory yang akan di-resize. 2. Menulis kembali file yang telah di-resize ke dalam directory baru. | Mengubah ukuran setiap gambar menjadi 480x480 px agar mudah diidentifikasi. |
| json2csv.py | 1. Membaca data json dan di-assign kedalam sebuah variabel baru. 2. Membaca setiap file dalam sebuah directory yang images-nya ditautkan oleh data json. 3. Menentukan size dimensi dari setiap gambar yang dimuat untuk keperluan normalisasi data. 4. Menentukan model gambar, titik (x, y), lebar, dan panjang dari objek api yang terdapat dalam sebuah gambar. 5. Data setiap gambar disimpan dalam dua buah array (normalisasi dan tidak dinormalisasi). 6. Menggambar kembali lokasi objek api pada sebuah gambar dan disimpan sebagai gambar baru dalam sebuah directory. 7. Melakukan pengurutan nama gambar agar sesuai dengan urutan pada directory. 8. Membuat file csv yang merupakan label dari setiap gambar. | Mengonversi data json menjadi file csv (label). |
| data.json | 1. Menyimpan seluruh data terkait pelabelan setiap gambar, mulai dari nama images titik setiap x,y, model gambar, dll. | Sumber data utama dalam penentuan label sebuah gambar. |
| interfaces.json | 1. Memuat data yang dibutuhkan dalam proses pelabelan. | Interface pada labelbox.io agar data json yang dihasilkan sesuai dengan kriteria. |
| urls.txt | 1. Memuat url setiap gambar yang ditautkan dalam website. | Memuat url setiap gambar yang ditautkan dalam website. |
| Labelbox.io |  | Tools untuk pelabelan dataset. |

**2.3. Tools**

Bahasa yang digunakan adalah Python dan Javascript, serta pertukaran data antara Labelbox.io (stius pelabelan) dengan program menggunakan JavaScript Object Notation (JSON). Python digunakan karena memiliki sangat banyak package (paket) atau module (modul) yang dibutuhkan dalam pengolahan data sedangkan Javascript sangat firiendly (ramah) terhadap layanan web dan browser. Paket python yang digunakan merupakan paket yang umum digunakan dalam Python sehingga sangat mudah didapatkan pada situs resmi setiap paket tersebut.

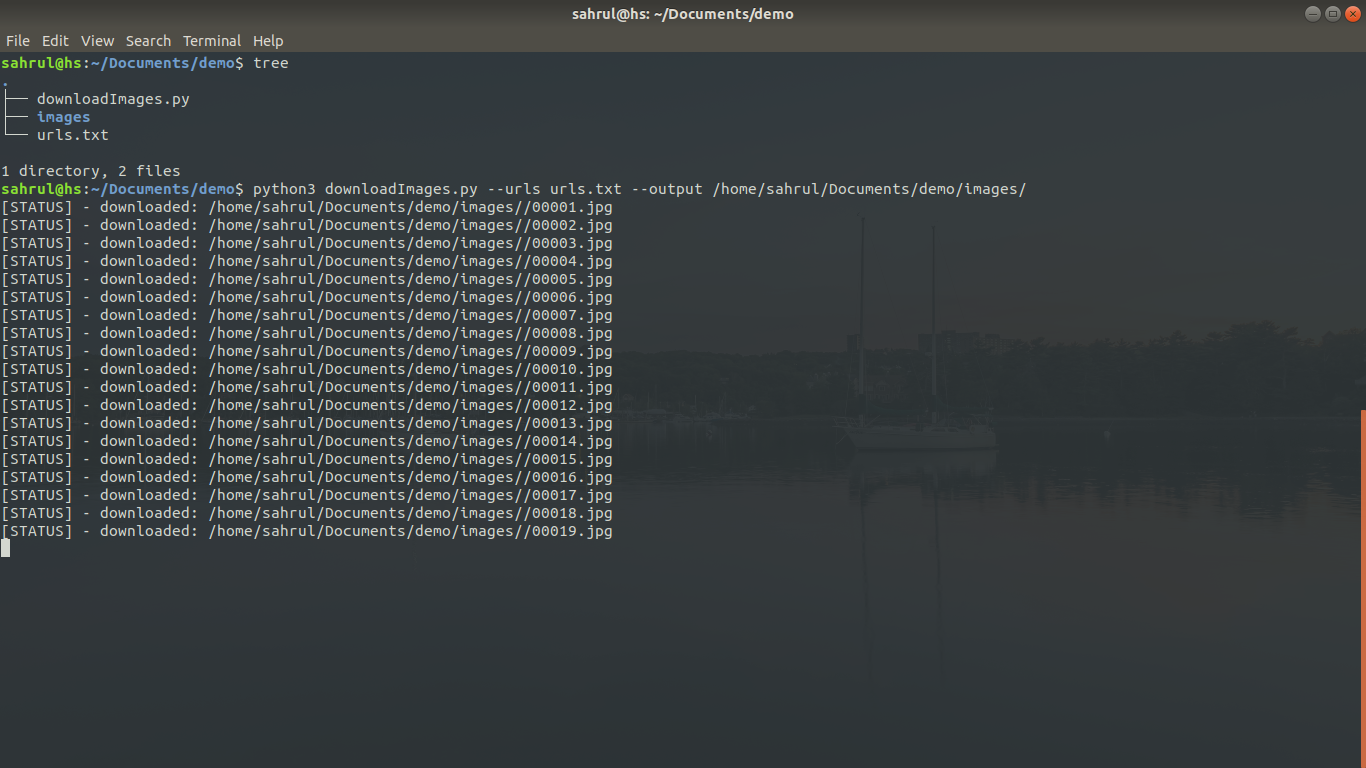
**2.4. Demo Program**

2.4.1. getImagesUrl.js



Note:

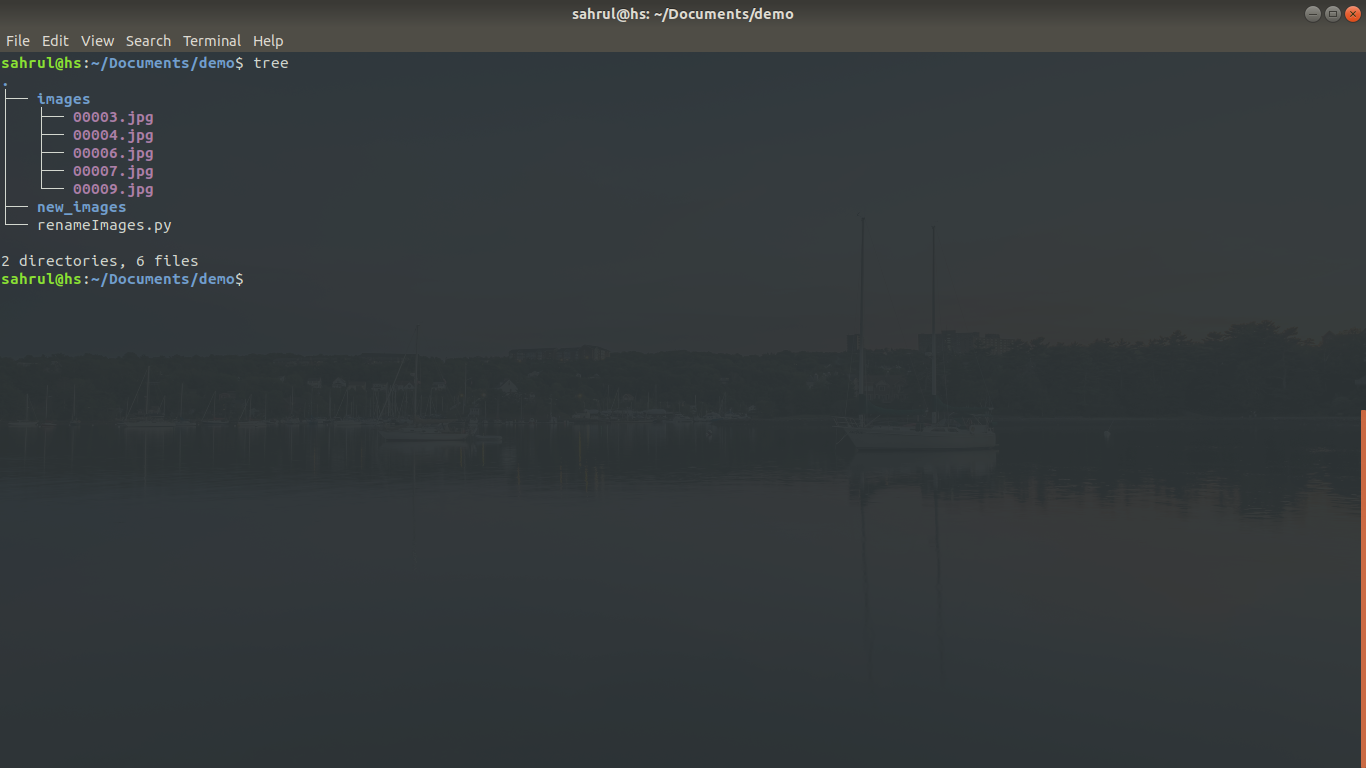
1. Program file **getImagesUrl.py** (petunjuk penggunaan lengkap ada di file tersebut).
2. Untuk mendapatkan lebih banyak gambar, tunggu browser untuk memuat seluruh gambar yang ada pada situs tersebut.
3. Memastikan pada proses terakhir, **urls.txt** berhasil diunduh.

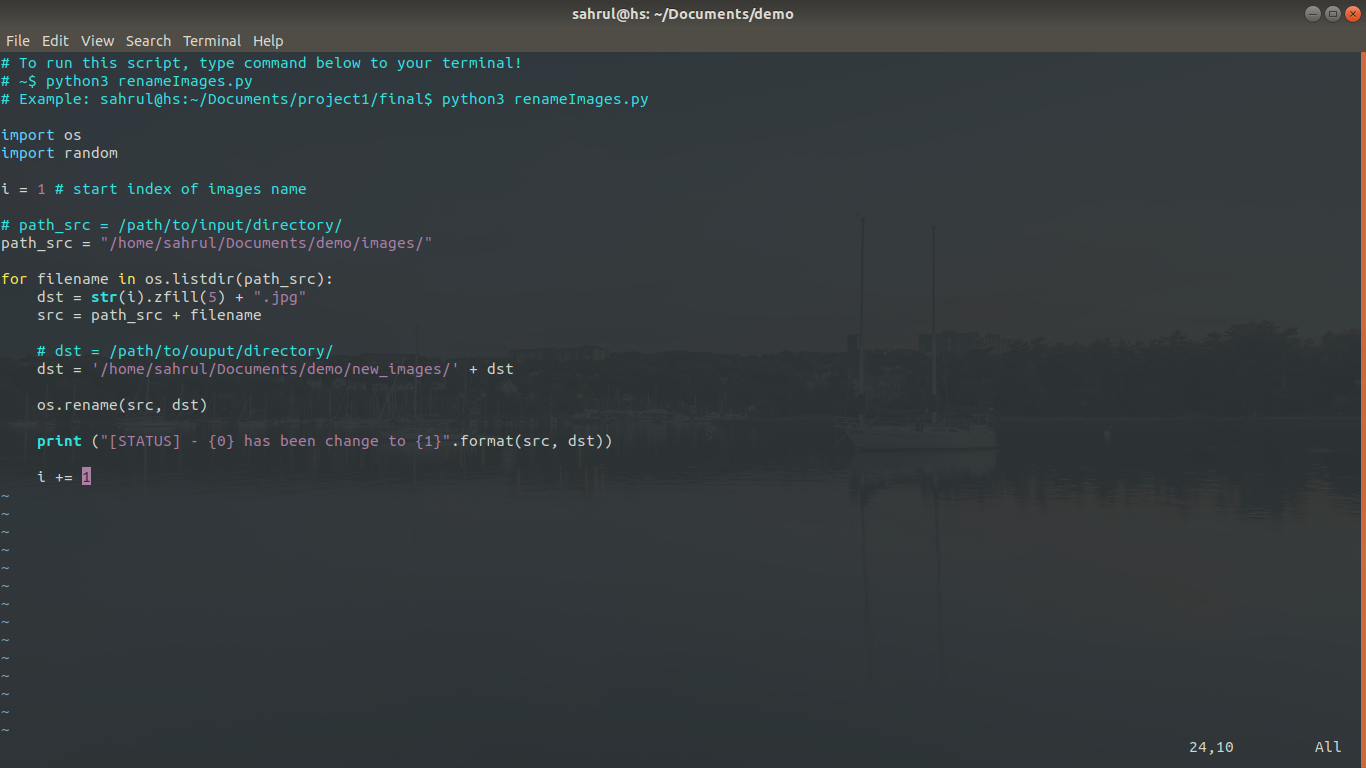
2.4.2. downloadImages.py

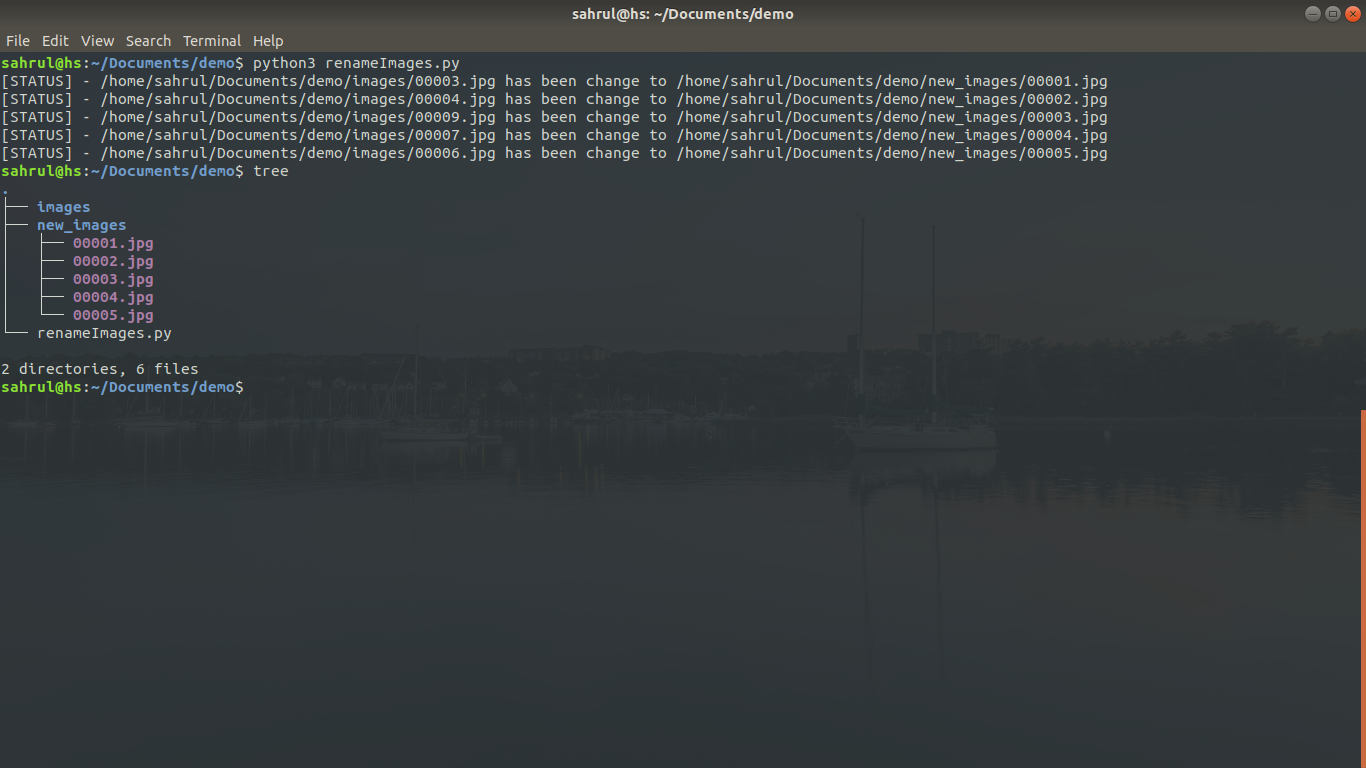
Note:

1. Program file **downloadImages.py** (petunjuk penggunaan lengkap ada di file tersebut)
2. --urls diikuti dengan file .txt yang memuat url setiap gambar.
3. --output diikuti dengan path ke directory dimana gambar akan disimpan.

2.4.3. renameImages.py



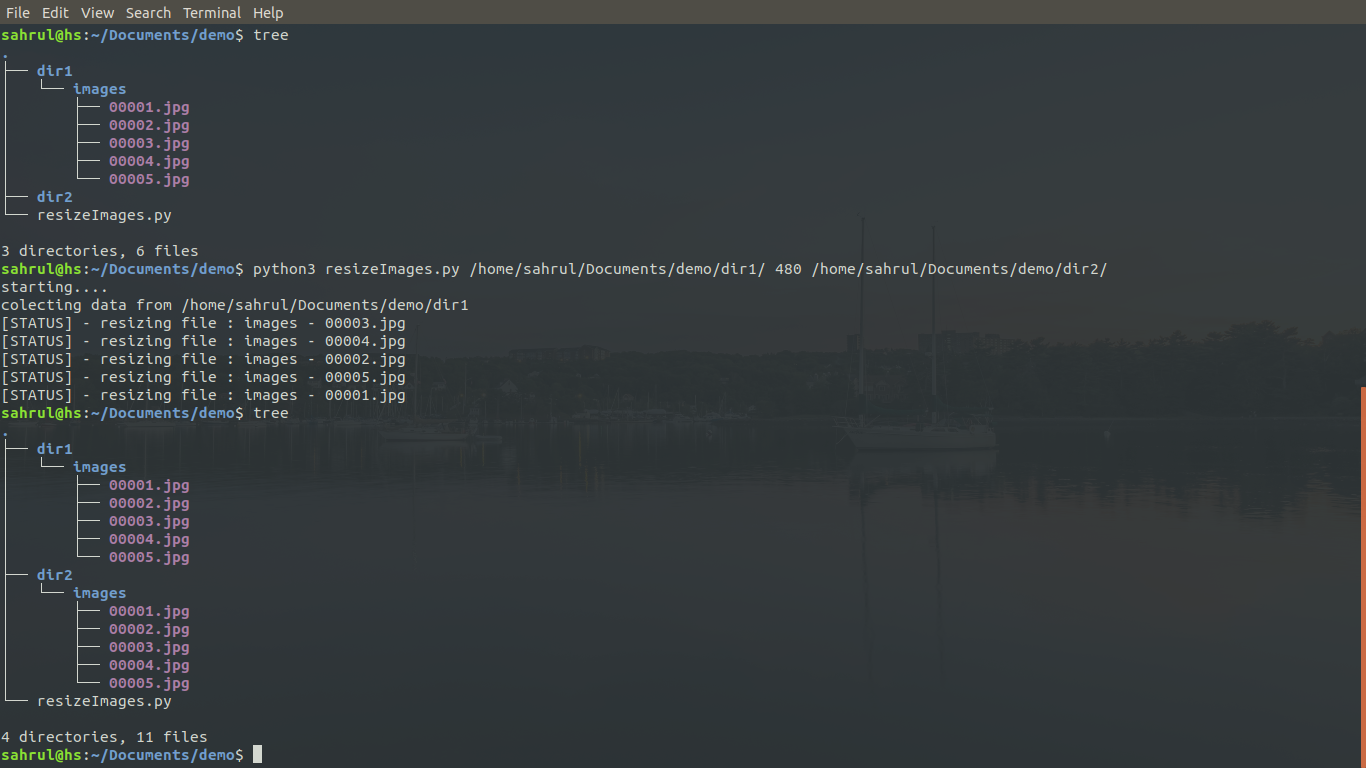


****

Note:

1. Program file **renameImages.py** (petunjuk penggunaan lengkap ada di file tersebut).
2. Memastikan terdapat directory baru untuk menyimpan semua gambar yang telah di rename.

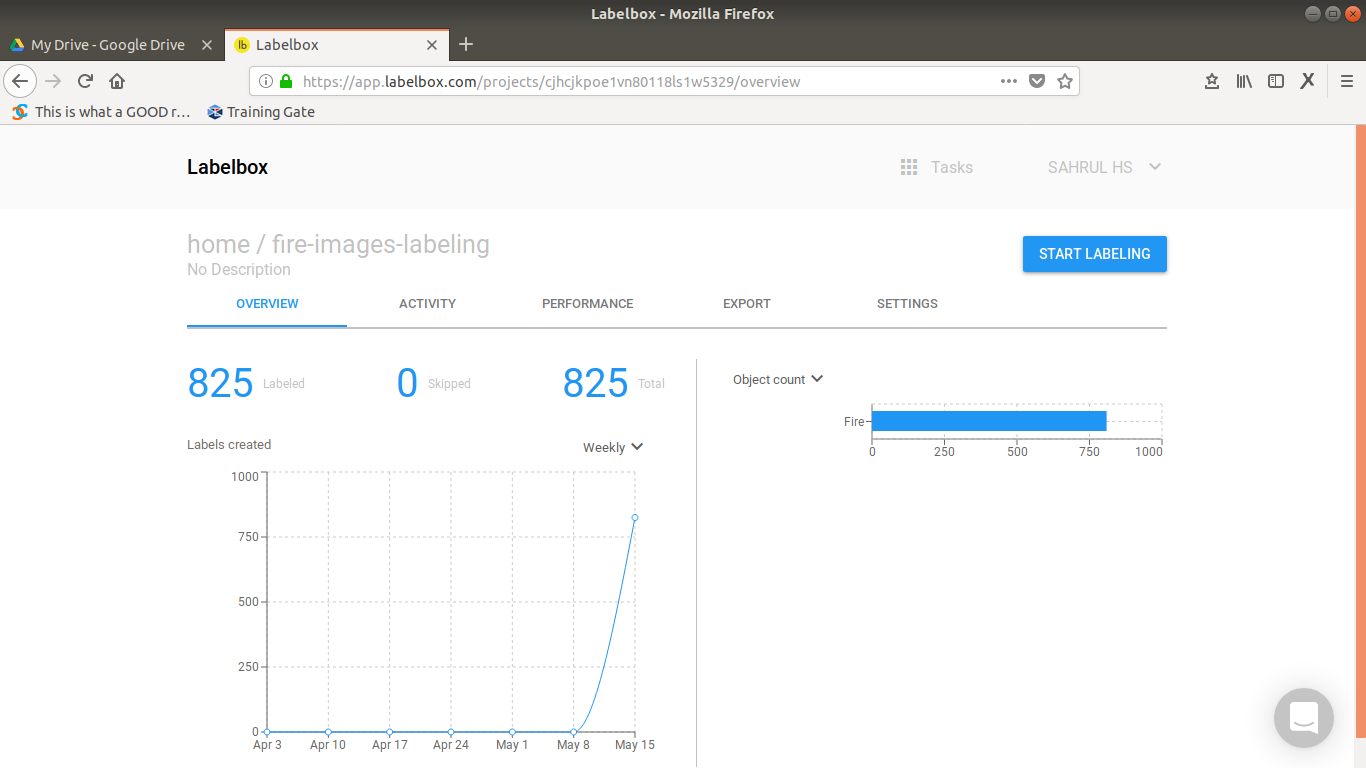
2.4.4. resizeImage.py



Note:

1. Program file **resizeImages.py** (petunjuk lengkap ada di file tersebut).
2. Program ini awalnya didesain untuk membaca beberapa directory dalam directory, sehingga path input directory harus memuat directory yang memuat gambar.
3. Input directory boleh sama dengan output directory, tapi beresiko apabila ingin mengembalikan gambar ke ukuran semula.

2.4.5. Labelbox.io



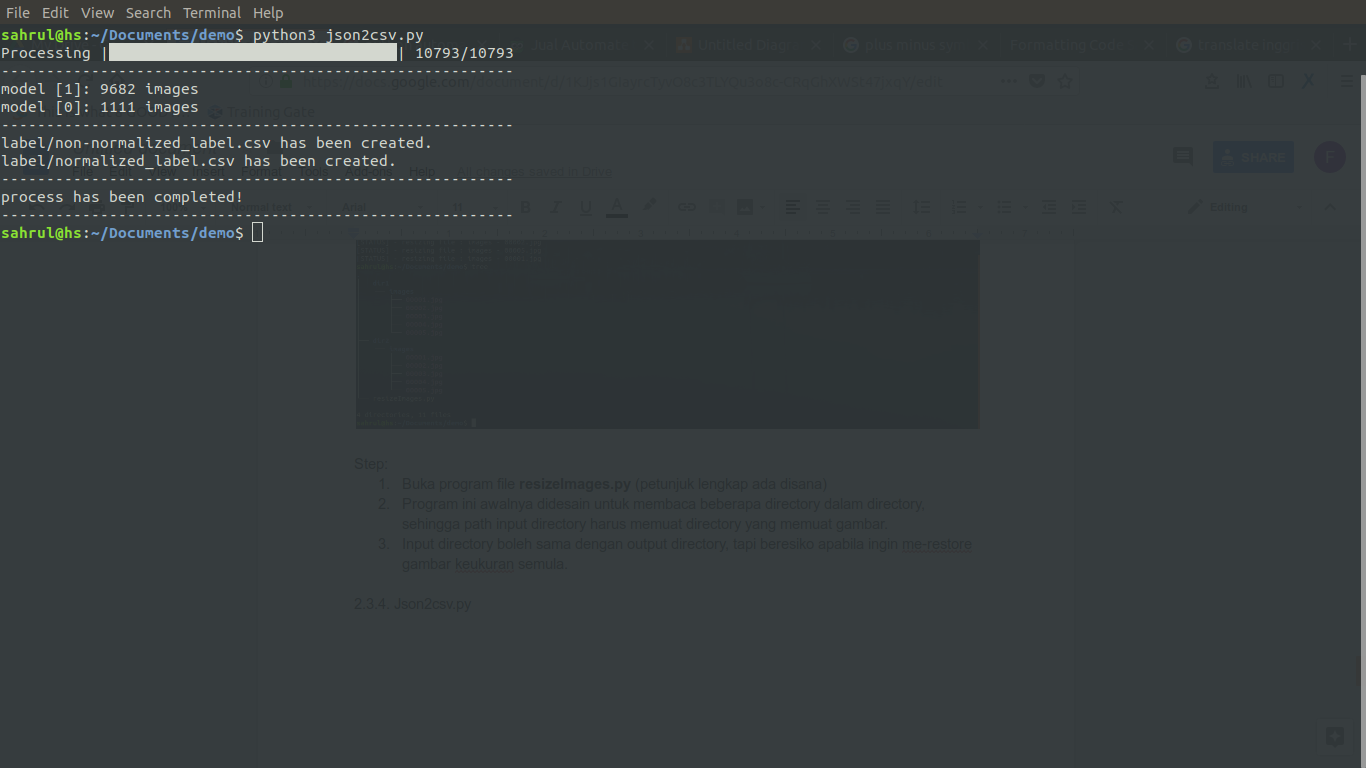
Step:

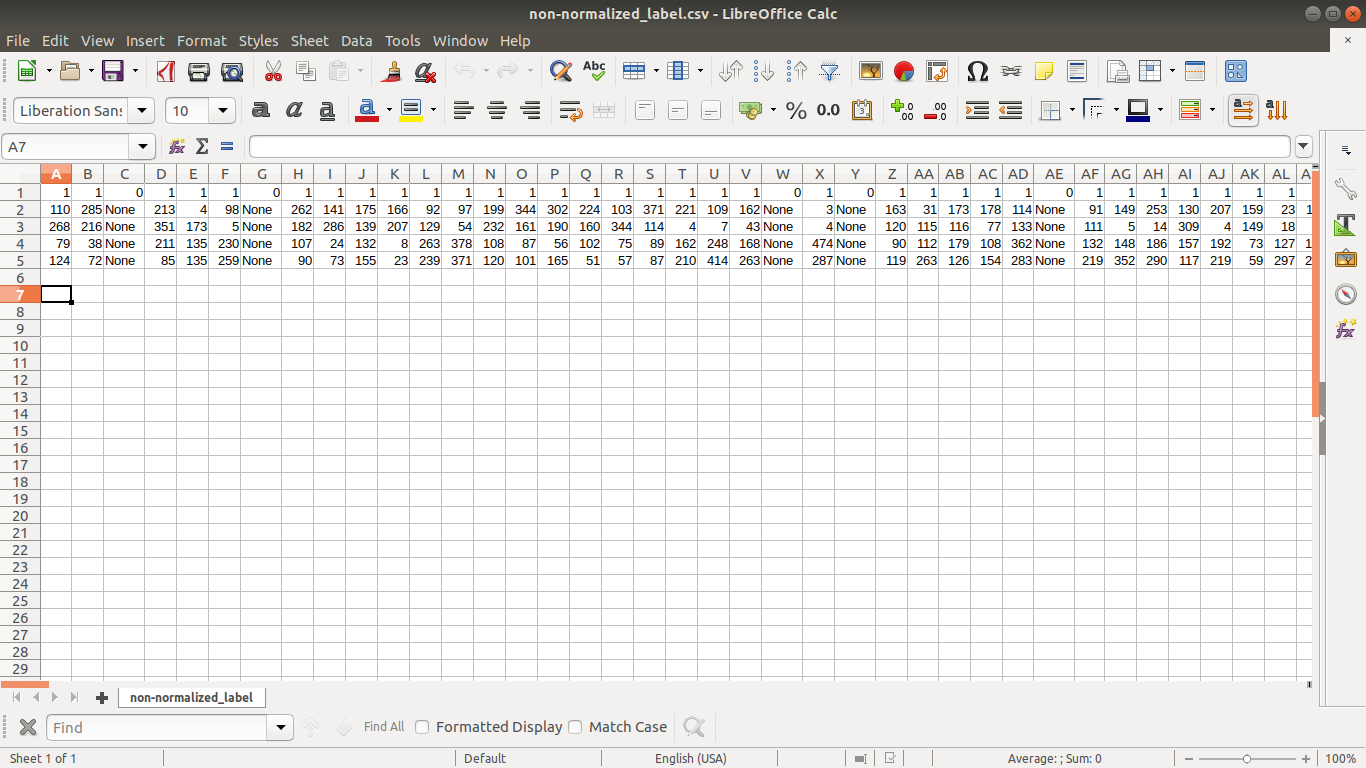
1. Lakukan registrasi pada situs Labelbox.io
2. Mengupload data source (images) yang akan dilakukan pelabelan.
3. Membuat sebuah project dengan meng-attach data source yang telah diupload sebelumnya.
4. Memilih metode pelabelan atau labelling frontend bagian Image Segmentation.
5. Mengubah interfaces pada situs Labelbox.io yang sesuai dengan file **interfaces.json** untuk menyesuaikan dengan kebutuhan.
6. Setelah interfaces dikonfirmasi dan menyelesaikan setup, pelabelan setiap gambar sudah dapat dimulai.
7. Setelah pelabelan semua gambar selesai, data di-export dalam format json.

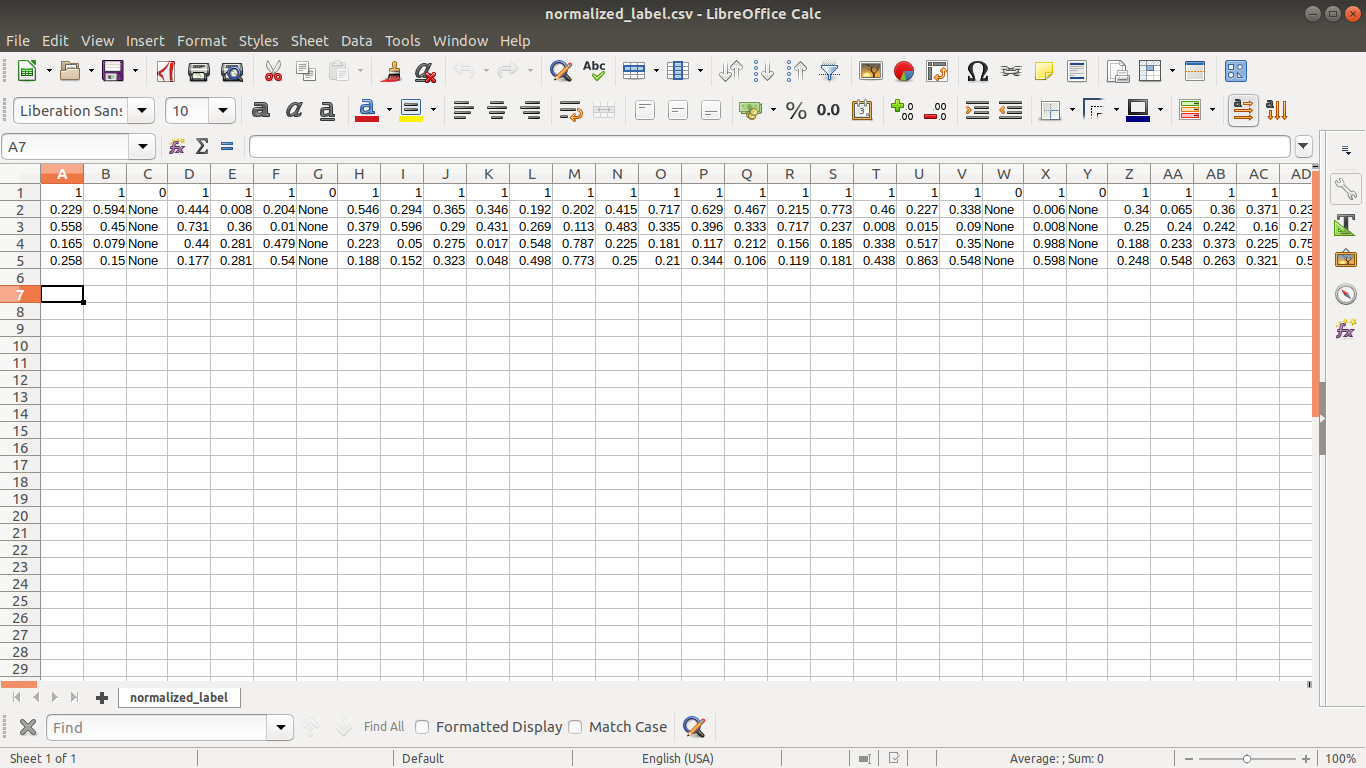
Note:

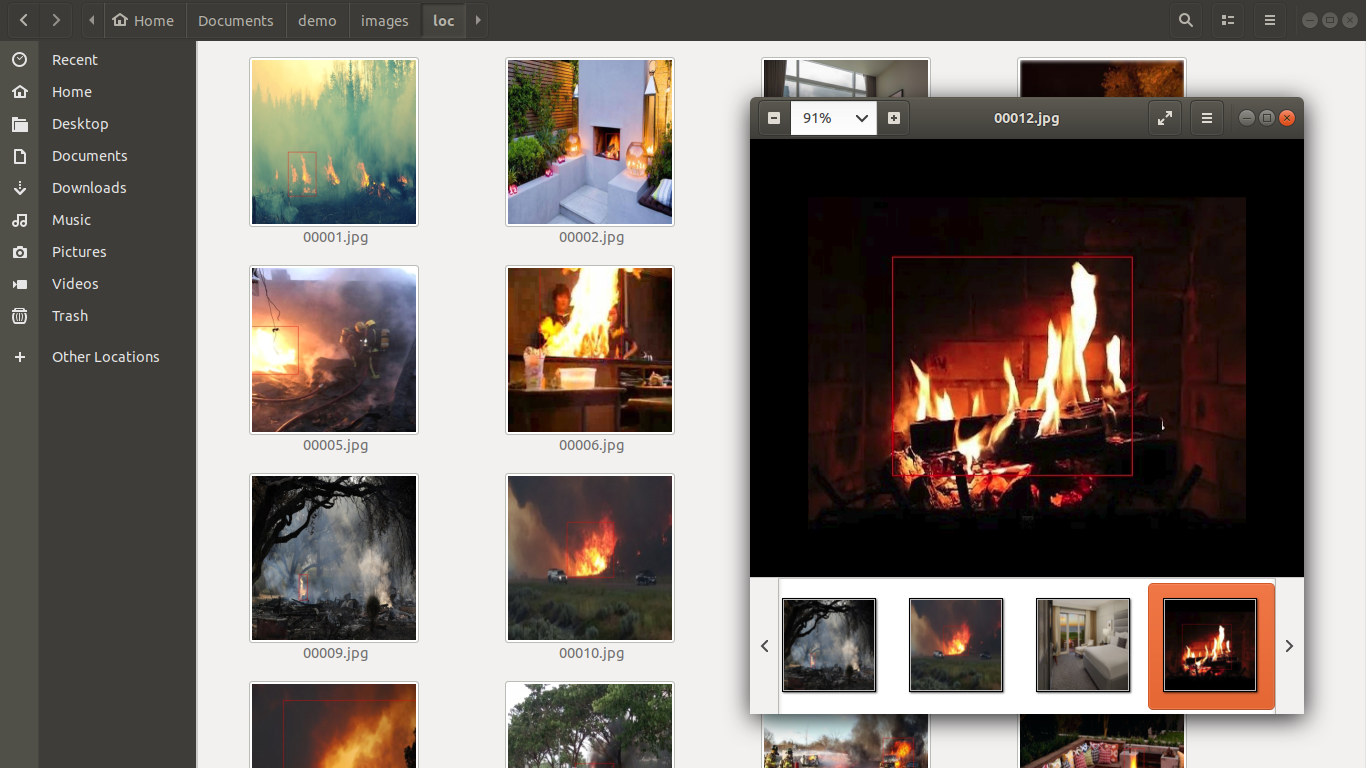
1. Labelbox.io hanya memberi kuota 5000 images untuk dilabelin setiap akun secara gratis.
2. Hasil pelabelan dapat di-review kembali atau diperbaiki kembali.
3. Situs Labelbox.io masih dalam proses pengembangan, sehingga seringkali tidak dapat diakses.

2.4.6. Json2csv.py



****

****

****

Note:

1. Program file **json2csv.py** (petunjuk penggunaan lengkap ada di file tersebut).
2. Memastikan setiap directory pada program tersebut sesuai dengan struktur directory pada komputer.

**3. KESIMPULAN**

Proses yang dilakukan dalam proyek ini mulai dari mengumpulkan dataset, pelabelan data, dan mengonversi label dalam bentuk matriks. Dari seluruh proses tersebut, kami berhasil mengumpulkan 10793 gambar lengkap dengan label setiap gambar dalam waktu kurang lebih 2 minggu.

Dengan selesainya proyek ini, kami dapat mengetahui bahwa beberapa pekerjaan repetitif dapat dilakukan dengan cepat dan akurat menggunakan program pendukung. Penggunaan aplikasi open-source sangat membantu dalam menyelesaikan proyek ini karena kami tidak perlu mengembangkan aplikasi dari awal.