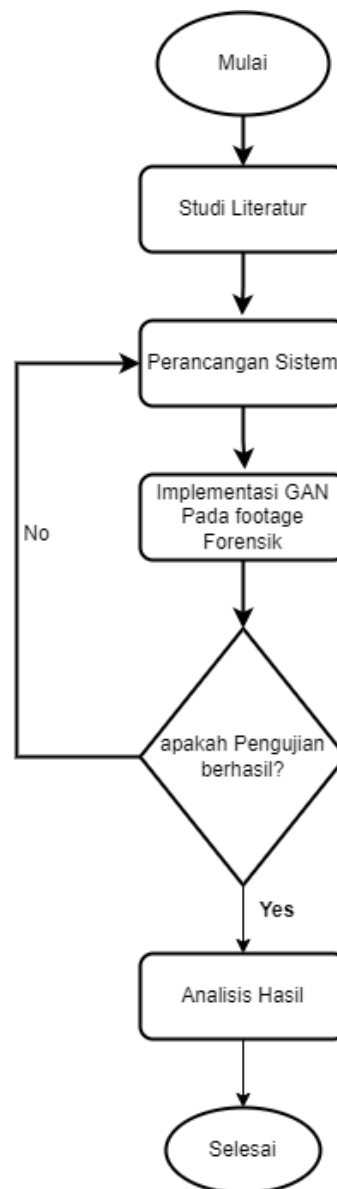


## BAB III METODOLOGI PENELITIAN

### 3.1 Alur Penelitian

Alur penelitian yang dilakukan pada penelitian ini dapat dilihat pada Gambar 3.1.



**Gambar 3. 1 Alur Penelitian**

### 3.2 Studi Literature

Studi literatur adalah kegiatan penting dalam dunia penelitian yang dilakukan dengan cara melakukan survei terhadap artikel ilmiah, buku, disertasi, konferensi, dan materi terbitan lainnya. Tujuannya adalah untuk memberikan ringkasan, deskripsi, dan evaluasi kritis dari suatu topik, masalah, atau bidang penelitian. Studi literature yang dilakukan oleh penulis meliputi pengambilan data berupa citra digital yang memiliki kerusakan Gaussian Blur dari kasus pencurian yang terekam CCTV berfokus kepada wajah dan detail pada citra, serta jurnal-jurnal dan video pembelajaran dari YouTube tentang penelitian sejenis.

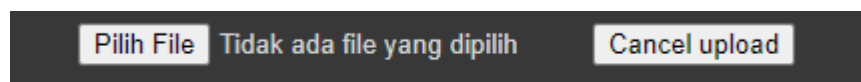
### 3.3 Perancangan Dan Cara Kerja Sistem

Dalam langkah pertama perancangan sistem, telah dilakukan pengumpulan data berupa citra forensik digital yang akan dijadikan sampel. Pada penelitian ini peneliti menggunakan Google Colab sebagai Tools pembuatan sistem. Sebelum memulai dilakukan Penginstalan *software* yang dibutuhkan yaitu Python 3, GPU, Kemudian mengkloning repositori, menyiapkan lingkungan (menginstal library, menginstal software yang diperlukan, konfigurasi editor kode), dan mengunduh model terlatih.

```
!wget
https://github.com/TencentARC/GFPGAN/releases/download/v1.3.0/GFPGANv1.3.pth -P experiments/pretrained_models
```

Selanjutnya dilakukan Upload Citra yaitu citra yang berkualitas rendah/buram. Inputan akan dimasukkan ke dalam folder GFPGAN pada bagian inputs

```
upload_folder = 'inputs/upload'
```



Gambar 3. 2 Input Citra

Jika citra yang ingin direkonstruksi hanya satu maka langsung saja unggah citra tersebut, namun jika menginginkan unggah folder maka diperlukan menghubungkan Google colab dengan Drive pastikan seluruh gambar yang ingin direkonstruksi berada pada drive, setelah terhubung tarik folder dari drive pindahkan ke 'input' folder GFPGAN, ganti bagian 'inputs/upload' menyesuaikan nama folder yang sudah dimasukan dan jalankan kode pemrograman selanjutnya.

```
from google.colab import drive
drive.mount('/content/drive')
```

Tahap selanjutnya dalam penelitian ini memerlukan penggunaan model untuk meningkatkan resolusi citra. Pemilihan model dilakukan berdasarkan versi GANs yang dapat menentukan faktor *upsample* akhir secara optimal. Dalam konteks ini, peneliti memilih Model GFPGAN, sebuah jenis GANs yang telah dirancang khusus untuk menghasilkan citra wajah berkualitas tinggi dari citra wajah berkualitas rendah. Model ini telah melalui pelatihan menggunakan dataset besar yang mencakup citra wajah berkualitas tinggi dan citra wajah berkualitas rendah.

Dalam penelitian ini, pelatihan manual tidak dilakukan. Keputusan ini didasarkan pada keterbatasan data yang tersedia dan keterbatasan waktu pelatihan dan keterbatasan spesifikasi perangkat keras. Oleh karena itu, peneliti memilih untuk menggunakan model GFPGAN yang telah diunggah oleh Xintao Wang di GitHub pada tanggal 11 Juni 2021 (<https://github.com/TencentARC/GFPGAN>).

Selanjutnya, data citra yang telah dikumpulkan akan diolah menggunakan model generator GFPGAN, dan kemudian model generator tersebut akan diuji performanya. Dalam mengatasi lingkungan sekitar (*environment*) pada citra non-wajah, digunakan model dari (<https://github.com/xinntao/Real-ESRGAN>), sementara untuk menekankan pada wajah, digunakan versi 1.3 model dari [https://github.com/TencentARC/GFPGAN#european\\_castle-](https://github.com/TencentARC/GFPGAN#european_castle-)

[model-zoo](#). Proses ini bertujuan untuk melakukan perbaikan pada citra agar menjadi lebih baik [29].

Lalu masuk ketahap rekontruksi citra berdasarkan kode dibawah ini diketahui bahwa kode menjalankan skrip python `inference_gfpgan.py`, menentukan input citra atau folder, `-o result` menentukan folder output dimana hasil restorasi akan disimpan. `-v 1.3` menentukan versi model GFPGAN, `-s 2` menentukan skala akhir *upsample*, `bg_upsampler realesrgan` menentukan metode *upsample* untuk lingkungan sekitar (*enviromtent*).

```
python inference_gfpgan.py -i inputs/upload -o results -v 1.3 -s 2 --
bg_upsampler realesrgan
```

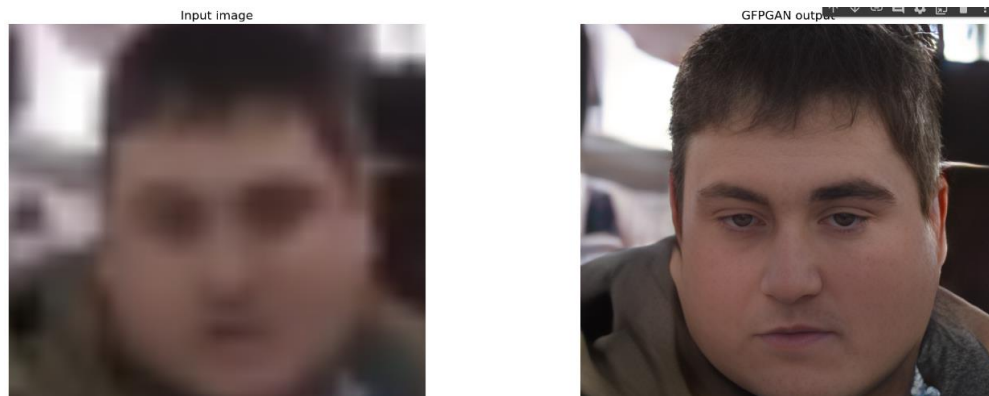
Selanjutnya menampilkan hasil rekontruksi citra yang telah di potong bagian wajah saja atau menampilkan hasil secara keseluruhan citra.

```
def display(img1, img2):
    fig = plt.figure(figsize=(25, 10))
    ax1 = fig.add_subplot(1, 2, 1)
    plt.title('Input image', fontsize=16)
    ax1.axis('off')
    ax2 = fig.add_subplot(1, 2, 2)
    plt.title('GFPGAN output', fontsize=16)
    ax2.axis('off')
    ax1.imshow(img1)
    ax2.imshow(img2)

# (Crop Faces)
    input_folder = 'results/cropped_faces'
    result_folder = 'results/restored_faces'

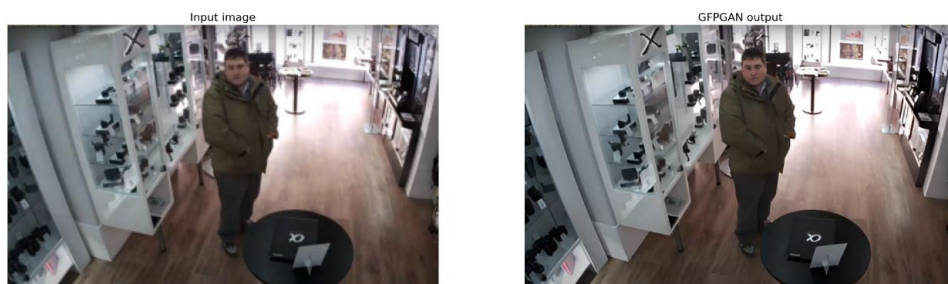
# (Whole Images)
    input_folder = 'inputs/upload'
result_folder = 'results/restored_imgs'
```

Berikut Hasil dari bagian yang berfokus pada wajah:



**Gambar 3. 3 Perbandingan Fokus Wajah**

Berikut hasil dari keseluruhan citra hasil rekontruksi:



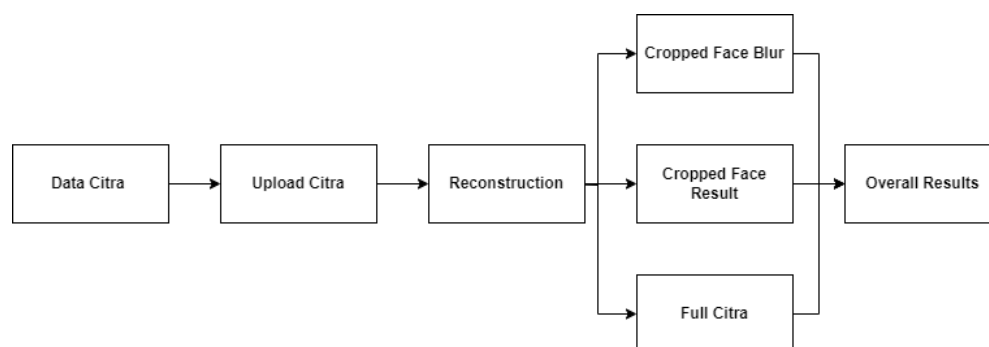
**Gambar 3. 4 Perbandingan Keseluruhan Citra**

Terakhir hasil dari rekontruksi citra dapat disimpan dengan format zip.

```
!ls results
print('Download results')
os.system('zip -r download.zip results')
files.download("download.zip")
```

### 3.4 Implementasi pada Footage Forensik Digital

Implementasi pada footage forensik digital dapat digambarkan seperti pada Gambar 3.2



**Gambar 3. 5 Bagan Implementasi**

Rancangan implementasi pada sistem ini menggambarkan penggunaan citra yang memiliki kualitas rendah sebagai input awal. Selanjutnya, citra tersebut akan diunggah dan mengalami pemrosesan di dalam program yang dijalankan sesuai dengan kode yang telah dikembangkan menggunakan bahasa pemrograman Python. Output yang dihasilkan dari eksekusi kode tersebut akan memperlihatkan perbaikan dalam kualitas citra yang telah direkonstruksi, hasil perbaikan citra dibagi menjadi 3 yaitu: citra dengan bagian wajah saja, citra bagian wajah setelah direkonstruksi, terakhir citra semula dengan bagian penuh yang sudah direkonstruksi. Keseluruhan citra hasil rekonstruksi dapat menjadi satu folder dan di simpan oleh pengguna.

### 3.5 Pengujian

Pengujian tetap dilakukan untuk memastikan bahwa model GFPGAN dapat bekerja dengan baik pada data penelitian, meskipun model tersebut telah dilatih pada dataset yang besar. Pengukuran kinerja juga diperlukan untuk memantau perkembangan kinerja model.

### 3.6 Analisis Hasil

Hasil dari peningkatan kualitas citra digital dievaluasi dan dianalisis. Ini dapat mencakup perbandingan citra digital sebelum dan sesudah, serta evaluasi apakah peningkatan kualitas memberikan manfaat dalam analisis forensik digital, selain membandingkan secara visual peneliti juga menggunakan penilaian menggunakan PSNR dan MSE.

Adapun skala presentasi untuk nilai PSNR dapat dilihat pada Gambar 3.

| Nilai PSNR | Kualitas citra              |
|------------|-----------------------------|
| 60 dB      | Baik, tanpa derau           |
| 50 dB      | Berderau, tetapi tetap baik |
| 40 dB      | Tampak butiran halus        |
| 30 dB      | Tampak banyak derau         |
| 20 dB      | Tidak bisa dipakai          |

**Gambar 3. 6 Skala Persentase PSNR**

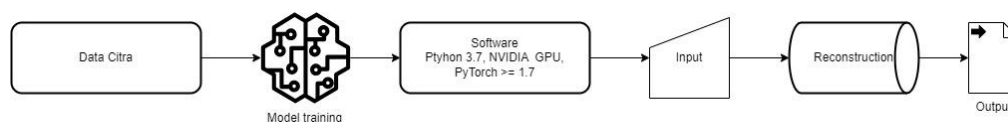
### 3.7 Cara Kerja Image Processing

*Image processing* merupakan metode untuk mengubah citra menjadi bentuk digital dan melakukan beberapa operasi padanya, untuk mendapat citra yang disempurnakan atau untuk mengekstrak beberapa informasi berguna darinya.

Adapaun komponen utamanya adalah *importing* (mengimpor), di mana citra digital diambil melalui *scanning* (pemindaian) atau fotografi digital. Kemudian ada komponen analisis dan manipulasi citra digital, dilakukan dengan menggunakan berbagai aplikasi perangkat lunak khusus, serta yang terakhir yaitu output yang merupakan keluaran[18].

### 3.8 Arsitektur Sistem

Arsitektur sistem pada penelitian ini mengikuti serangkaian langkah langkah yang terstruktur. Flowchart di bawah ini memberikan gambaran tentang bagaimana penelitian ini akan dilaksanakan, dapat dilihat pada gambar 3.3



**Gambar 3. 7 Flowchart Arsitektur Sistem**

Arsitektur sistem yang akan dirancang dimulai dengan menyiapkan dataset atau citra *low quality* sebagai inputan awal, lalu menyiapkan model yang cocok pada penelitian. Tahap selanjutnya adalah penginstalan *software* menyesuaikan kebutuhan model yaitu python 3,7, NVIDIA GPU, Pytorch

>=1.7. Citra tersebut akan mengalami proses di dalam program yang dijalankan sesuai dengan kode yang telah dikembangkan menggunakan bahasa pemrograman Python. kemudian dilakukan rekonstruksi Output yang dihasilkan dari eksekusi kode tersebut akan memperlihatkan perbaikan dalam kualitas citra yang telah direkonstruksi. Dengan demikian, sistem ini berfokus pada pengolahan citra yang awalnya memiliki kualitas rendah, namun melalui serangkaian operasi di dalam program, menghasilkan peningkatan visual pada citra sebagai output akhir.

### **3.9 Spesifikasi Perangkat Keras (Hardware)**

Adapun perangkat keras yang digunakan dalam pembuatan sistem pada penelitian ini adalah sebagai berikut:

- a) 11<sup>th</sup> Gen Intel® Core(TM) i7-11700f @ 2.50Hz 2.50
- b) GPU NVIDIA GeForce RTX 3080
- c) RAM 16.0 GB

### **3.10 Spesifikasi Perangkat Lunak (Software)**

Adapun perangkat Lunak yang digunakan dalam pembuatan sistem pada penelitian ini adalah sebagai berikut:

- a) Sistem Operasi 64bit;
- b) Google Colab
- c) Python 3.7
- d) Pytorch 1.7