BAB III
METODE PENELITIAN

3.1 Objek Penelitian

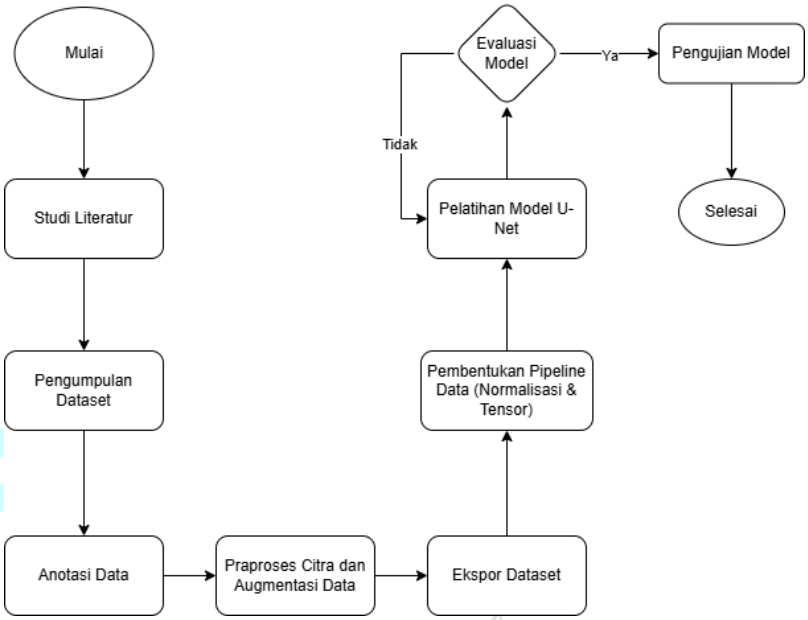
Objek penelitian ini mengambil studi kasus citra rokok yang diambil dari berbagai kondisi lingkungan, seperti ruangan terang, ruangan gelap, posisi tangan berbeda, serta latar belakang bervariasi. Seluruh citra digunakan sebagai data untuk proses segmentasi berbasis *U-Net*. Setiap gambar berisi objek rokok yang memiliki ukuran kecil, bentuk yang tipis, dan warna yang sering kali menyerupai latar belakang, sehingga memerlukan pendekatan segmentasi piksel untuk memperoleh hasil deteksi yang akurat. Citra rokok tersebut kemudian dianotasi menjadi *mask* (label piksel) sebagai dasar pelatihan model. Contoh objek penelitian bisa dilihat pada Gambar 3.1.



Gambar 3. 1 Contoh Objek

3.2 Prosedur Penelitian

Prosedur penelitian disusun untuk memberikan alur kerja yang runtut sejak pengumpulan data hingga pengujian model. Langkah-langkah utama yang dilakukan dapat dilihat pada Gambar 3.2:



Gambar 3. 2 Alur Diagram Prosedur Penelitian

Gambar 3.2 merupakan Alur diagram pada penelitian yang dapat diuraikan sebagai berikut:

1. Studi Literatur: Tahap ini dilakukan untuk mengumpulkan landasan teori terkait segmentasi citra, arsitektur *U-Net*, pengolahan citra digital, serta penelitian terdahulu mengenai deteksi objek kecil seperti rokok. Studi literatur membantu peneliti memahami metode terbaik yang dapat diterapkan dalam penelitian ini.
2. Pengumpulan *Dataset*: *Dataset* dikumpulkan melalui *Dataset* publik yang tersedia seperti pada platform *kaggle* dan juga *google images*.
3. Anotasi Data: Setiap citra kemudian diunggah ke *Roboflow* dan dilakukan anotasi menggunakan *Semantic Segmentation* (Poligon).
4. Praproses Citra dan Augmentasi Data: Untuk memperkaya keragaman fitur pada data latih, citra yang telah dianotasi diberikan perlakuan praproses dasar dan augmentasi pada platform *Roboflow*. Teknik augmentasi yang digunakan meliputi horizontal *flip*, *vertical flip*, rotasi (-15° hingga +15°), penyesuaian *brightness* dan *contrast*, serta *zoom*.
5. *Export* Data: *Dataset* yang telah selesai melalui tahapan anotasi dan augmentasi kemudian diekspor atau diunduh langsung ke dalam format *Semantic Segmentation Masks*. Pada format ini, label *Dataset* telah berbentuk citra *binary mask* (hitam-putih) yang siap digunakan.

6. Pembentukan *Pipeline* Data: Tahap ini merupakan praproses secara komputasi untuk menyiapkan citra asli dan citra *mask* sebelum diproses oleh model.
7. Pelatihan Model *U-Net*: Model *U-Net* dilatih menggunakan *Dataset* yang telah melalui tahap *pipeline*. Proses pelatihan ini bertujuan agar model mampu mengenali konteks dan fitur semantik dari rokok hingga ke tingkat piksel, sehingga segmentasi untuk objek berukuran kecil tersebut dapat dilakukan secara akurat.
8. Evaluasi Model: Model hasil pelatihan dievaluasi menggunakan matriks *Intersection over Union (IoU)*, *Dice Coefficient*, dan *Accuracy*.
    a. Apabila hasil evaluasi menunjukkan akurasi rendah, maka proses kembali ke tahap pelatihan model *U-Net* untuk dilakukan penyesuaian parameter atau pengulangan pelatihan.
    b. Proses pelatihan dan evaluasi ini dilakukan secara iteratif hingga diperoleh performa model yang optimal.
9. Pengujian Model: Setelah model mencapai hasil evaluasi yang baik, model selanjutnya masuk ke tahap pengujian model menggunakan data uji (*testing* data). Tahap ini bertujuan untuk mengetahui kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

3.2.1 Studi Literatur

Tahap ini dilakukan untuk memperoleh pemahaman mendalam mengenai teori yang berkaitan dengan deteksi objek, segmentasi citra, *deep learning*, arsitektur *U-Net*, serta metode evaluasi yang digunakan pada penelitian sejenis. Literatur yang dipelajari meliputi jurnal internasional, jurnal nasional, buku, artikel akademik, serta penelitian sebelumnya mengenai deteksi objek kecil, segmentasi piksel, dan implementasi *U-Net* untuk berbagai domain. Studi ini membantu menyusun landasan teoritis serta menentukan metode yang paling relevan untuk deteksi rokok.

3.2.2 Pengumpulan Data

Dataset yang digunakan dalam penelitian ini diperoleh melalui pengumpulan data sekunder dari sumber publik, yaitu repositori *Kaggle* dan *Google Images*. Total data yang dikumpulkan berjumlah 1.680 citra yang mencakup berbagai variasi

pencahayaan, sudut pandang, latar belakang, dan jarak objek. Penggunaan *Dataset* publik dengan jumlah tersebut bertujuan agar model *U-Net* dapat mengenali fitur objek secara lebih luas dan memiliki kemampuan generalisasi yang baik. Seluruh citra kemudian diunggah ke platform *Roboflow* untuk diproses lebih lanjut melalui tahap anotasi dan augmentasi guna mendukung sistem deteksi *real-time* yang akurat pada kondisi nyata.

3.2.3 Anotasi Data

Tahap anotasi dilakukan untuk menandai area rokok pada setiap citra. Proses anotasi dikerjakan secara manual menggunakan *platform Roboflow* dengan metode *Semantic Segmentation*. Setiap objek rokok digambar menggunakan *polygon mask* yang mengikuti bentuk asli objek sehingga menghasilkan label yang akurat.

Anotasi ini bertujuan memberikan *ground truth mask* yang akurat agar model *U-Net* dapat mempelajari bentuk rokok secara detail, termasuk garis tepi yang tipis. Hasil anotasi ini kemudian diekspor ke dalam format *Semantic Segmentation Mask*. Format tersebut menghasilkan citra label biner (*Binary Mask*) di mana piksel bernilai 1 merepresentasikan kelas objek rokok dan piksel bernilai 0 merepresentasikan latar belakang *(background).*

3.2.4 Praproses Citra dan Augmentasi Data

Sebelum *Dataset* diperbanyak, citra yang telah dianotasi terlebih dahulu melalui tahap praproses awal (*preprocessing*) pada platform *Roboflow*. Praproses tingkat dasar ini berfungsi untuk menstandarisasi format visual gambar sebelum diproses lebih lanjut. Tahapan ini mencakup:

1. Koreksi Orientasi *(Auto-Orient):* Bertujuan untuk memastikan seluruh citra berada pada posisi tegak yang seragam, serta memperbaiki citra yang miring atau terbalik pada saat pengambilan gambar
2. Penyesuaian Dimensi (*Resize*) dan *Padding*: Seluruh citra diseragamkan ukurannya menjadi resolusi 512 × 512 piksel menggunakan teknik *fit (black edges)*. Teknik ini menambahkan area kosong berwarna hitam *(padding)* pada citra yang memiliki rasio aspek non-persegi. Hal ini dilakukan untuk mencegah terjadinya distorsi spasial, sehingga bentuk asli rokok tidak memanjang atau memendek secara tidak wajar.

Setelah tahap praproses awal selesai, citra latih diperbanyak melalui proses augmentasi data. Tahap ini bertujuan untuk meningkatkan keragaman (*variance*) citra pelatihan sekaligus mengurangi risiko *overfitting* pada model segmentasi berbasis *U-Net*. Pada tugas *semantic segmentation*, augmentasi tidak hanya membantu model mengenali keberadaan objek, tetapi juga memperkuat kemampuannya dalam mempelajari bentuk, kontur, dan batas tepi objek rokok secara lebih akurat hingga tingkat piksel. Proses ini menerapkan transformasi spasial dan piksel yang tetap menjaga konsistensi letak antara citra asli dan *binary mask*. Teknik augmentasi yang diterapkan meliputi:

1. *Flip Horizontal:*
   Digunakan untuk menghasilkan variasi orientasi objek rokok secara kiri–kanan, sehingga model dapat mengenali objek dari arah yang berbeda.
2. *Flip Vertical:*
   Digunakan untuk memperkaya variasi posisi objek rokok secara atas–bawah pada citra pelatihan.
3. Rotasi:
   Diterapkan untuk merepresentasikan kemiringan objek rokok pada proses pengambilan gambar. Rentang sudut rotasi dibatasi agar tidak mengubah bentuk dan struktur utama objek pada *mask* segmentasi.
4. Penyesuaian kontras:
   Digunakan untuk mensimulasikan perbedaan kondisi pencahayaan yang dapat memengaruhi tampilan visual objek rokok.
5. Penyesuaian kecerahan:
   Digunakan untuk merepresentasikan variasi intensitas cahaya pada lingkungan pengambilan citra.

Pemilihan tingkat augmentasi dalam penelitian ini didasarkan pada pertimbangan dari beberapa penelitian sebelumnya. Hu *et al.* (2019) menjelaskan bahwa penambahan data melalui augmentasi dapat meningkatkan kinerja model *deep learning* hingga batas tertentu, dan menyebutkan bahwa augmentasi data hingga dua sampai tiga kali lipat (2–3x) dari *Dataset* awal sudah cukup efektif. Setelah jumlah augmentasi melebihi rentang tersebut, variasi citra yang dihasilkan cenderung tidak jauh berbeda dari data sebelumnya. Akibatnya, model tidak

memperoleh informasi baru yang signifikan, sehingga peningkatan kinerja yang dihasilkan menjadi sangat terbatas. Di sisi lain, proses pelatihan membutuhkan waktu yang lebih lama serta penggunaan sumber daya komputasi yang lebih besar, sehingga penambahan augmentasi dalam jumlah berlebihan menjadi kurang efisien.

Selain itu, penelitian yang dilakukan oleh (Farhan, 2024) menunjukkan bahwa peningkatan performa model lebih dipengaruhi oleh penggunaan kombinasi teknik augmentasi yang tepat dibandingkan dengan memperbanyak data secara berlebihan. Dalam penelitiannya, penggunaan beberapa teknik augmentasi dengan intensitas yang wajar sudah mampu meningkatkan variasi data pelatihan tanpa menghasilkan citra yang kurang realistis. Temuan ini juga sejalan dengan penelitian yang lebih baru, seperti (Xu *et al.*, 2023) dan (Z. Wang *et al.*, 2021), yang menyatakan bahwa augmentasi dengan intensitas berlebihan cenderung menghasilkan data yang bersifat redundan dan tidak selalu memberikan peningkatan kinerja yang sebanding.

Selain itu, penelitian yang dilakukan oleh Farhan (2024) menunjukkan bahwa peningkatan performa model lebih dipengaruhi oleh penggunaan kombinasi teknik augmentasi yang tepat dibandingkan dengan memperbanyak data secara berlebihan. Dalam penelitiannya, penggunaan beberapa teknik augmentasi dengan intensitas yang wajar sudah mampu meningkatkan variasi data pelatihan tanpa menghasilkan citra yang kurang realistis. Temuan ini juga sejalan dengan penelitian yang lebih baru, seperti Xu *et al*.(2023) dan Z. Wang *et al*. (2021), yang menyatakan bahwa augmentasi dengan intensitas berlebihan cenderung menghasilkan data yang bersifat redundan dan tidak selalu memberikan peningkatan kinerja yang sebanding.

Berdasarkan pertimbangan tersebut, tingkat augmentasi pada penelitian ini diterapkan secara moderat, yaitu dengan memperbanyak data pelatihan hingga sekitar dua sampai tiga kali dari jumlah *Dataset* awal, guna menjaga keseimbangan antara variasi data yang cukup, ketepatan segmentasi objek rokok hingga tingkat piksel, serta efisiensi proses pelatihan model *U-Net*.

3.2.5 Ekspor *Dataset* dan Pembagian *Dataset*

Dataset yang telah melalui tahap praproses dan augmentasi kemudian diekspor dari platform *Roboflow*. Untuk memenuhi kebutuhan *input* model *U-Net*,

pengeksporan dilakukan secara langsung menggunakan format *Semantic Segmentation Masks*. Melalui format ini, sistem secara otomatis menghasilkan luaran berupa pasangan citra asli (RGB) dan citra label berwujud *binary mask* (hitam-putih). Pada citra *binary mask* tersebut, piksel yang bernilai 1 merepresentasikan area kelas objek (rokok), sedangkan piksel bernilai 0 merepresentasikan area latar belakang (*background*). Format ini mengeliminasi kebutuhan konversi titik koordinat poligon secara manual, sehingga *Dataset* siap untuk diproses ke dalam tahap komputasi *pipeline*.

Pada proses ekspor ini, *Roboflow* juga secara otomatis membagi keseluruhan *Dataset* ke dalam tiga *subset*, yaitu data latih (*training*) sebesar 80%, data validasi (*validation*) sebesar 20%. Proporsi pembagian ini dipertahankan selama proses pemodelan berlangsung. Tujuannya adalah untuk memastikan model *U-Net* memiliki jumlah data pelatihan yang memadai untuk mempelajari pola visual objek, sekaligus menyediakan data validasi dan data uji yang cukup untuk mengukur serta mengevaluasi performa dan tingkat generalisasi model secara objektif.

3.2.6 Pembentukan *Pipeline* Data (Normalisasi dan *Tensor*)

Setelah *Dataset* diekspor, tahap selanjutnya adalah pembentukan *pipeline* data secara komputasi. Tahap ini sangat penting untuk memastikan citra asli dan label (*mask*) siap diproses oleh arsitektur *U-Net* melalui struktur data yang efisien. Proses ini dilakukan menggunakan *library TensorFlow* dan mencakup tiga langkah utama:

1. Normalisasi Nilai Piksel

   Secara standar, nilai piksel pada citra digital direkam dalam bentuk bilangan bulat dari 0 sampai 255. Untuk mempermudah kerja algoritma *U-Net*, rentang nilai ini dikonversi menjadi angka desimal antara 0,0 sampai 1,0. Proses konversi dilakukan cukup dengan membagi setiap piksel dengan angka batas maksimumnya, yaitu 255 menggunakan tipe data float32. Langkah ini diterapkan agar perhitungan di dalam model tidak harus memproses deretan angka yang terlalu besar. Dengan skala angka yang lebih ringkas, kinerja model saat proses pelatihan menjadi jauh lebih stabil dan optimal..
2. Konversi ke Format *Tensor*

Pasangan citra dan *mask* dikonversi ke dalam struktur data *Tensor* agar dapat diproses secara paralel oleh unit komputasi grafis (GPU). Pada tahap ini juga dilakukan penetapan dimensi (*set shape*) secara eksplisit sebesar 512×512 piksel pada setiap data untuk memastikan konsistensi *input shape* sebelum masuk ke dalam lapisan pertama model *U-Net*.

3. Pengaturan *Batching* dan *Prefetching*

   Pada tahap ini, *Dataset* dibagi menjadi beberapa kelompok (*batch*) dengan ukuran tertentu agar sesuai dengan sumber daya GPU yang tersedia. Peneliti juga menerapkan mekanisme *prefetching* untuk menciptakan jalur distribusi data yang lebih cepat. Sistem akan mengambil dan menyiapkan data selanjutnya secara otomatis sebelum proses pelatihan pada kelompok data sebelumnya selesai. Hal ini dilakukan untuk menghindari hambatan (*bottleneck*) pada aliran data, yang pada akhirnya mempercepat keseluruhan waktu komputasi pelatihan model *U-Net*.

Secara rangkaian persiapan ini mengubah *Dataset* yang berawal berupa kumpulan data mentah menjadi struktur data yang siap diolah secara sistematis. Standarisasi format dan optimalisasi alur data ini menjadi fondasi penting agar arsitektur *U-Net* dapat fokus mempelajari fitur visual objek tanpa terganggu oleh kendala teknis pada data masukan, sehingga proses pelatihan menghasilkan performa yang lebih konsisten.

3.2.7 Pelatihan Model *U-Net*

Model dibangun menggunakan *framework TensorFlow* dan *Keras*. Arsitektur yang digunakan adalah *U-Net* dengan *EfficientNet-B0* sebagai *backbone* (*encoder*) yang telah dilatih sebelumnya (*pre-trained*) pada *Dataset ImageNet*. Penggunaan *backbone* ini bertujuan untuk mempercepat proses konvergensi dan meningkatkan akurasi ekstraksi fitur pada objek rokok. Pelatihan dilakukan menggunakan unit GPU agar proses komputasi berjalan lebih cepat dan efisien.

Untuk mendukung komputasi yang optimal, proses pelatihan model memanfaatkan platform *Kaggle Notebook*. Penggunaan platform ini bertujuan untuk mengakses unit GPU berperforma tinggi, yaitu NVIDIA Tesla P100 dengan VRAM sebesar 16GB, sehingga proses iterasi pada *Dataset* dapat berjalan lebih stabil.

Tabel 3. 1 Perangkat dan Spesifikasi yang digunakan untuk proses pelatihan

Perangkat	**Spesifikasi**
Platform Pelatihan	*Kaggle Notebook (Cloud Based)*
GPU (*Cloud*)	NVIDIA Tesla P100-PCIE-16GB
RAM (*Cloud*)	29 GB
Perangkat Lokal	Laptop Lenovo LOQ (Proses Pengujian Sistem)
CPU Lokal	Intel Core i7-13650HX
RAM	28 GB DDR5
Storage	512 GB NVMe SSD
Framework	*TensorFlow-Keras*

Dan juga dengan Parameter pelatihan meliputi:

1. *Loss function: Loss Function: Focal Loss + Dice Loss*

   Fungsi *Loss* yang digunakan merupakan kombinasi antara *Focal Loss* dan *Dice Loss* dengan bobot masing-masing 0,4 dan 0,6. *Focal Loss* digunakan karena mampu menangani masalah ketidakseimbangan kelas (*class imbalance*), yaitu kondisi di mana piksel objek rokok jauh lebih sedikit dibandingkan piksel latar belakang. Sementara itu, *Dice Loss* berfungsi untuk memaksimalkan tumpang tindih (*overlap*) antara hasil prediksi model dengan *ground truth*.

2. *Optimizer: Adam*

   Optimizer Adam digunakan karena memiliki kemampuan adaptif dalam menyesuaikan nilai *learning rate* selama proses pelatihan. Adam menggabungkan kelebihan metode momentum dan *adaptive learning rate*, sehingga mampu mempercepat konvergensi model dan menjaga kestabilan proses pembelajaran, terutama pada *Dataset* dengan jumlah data yang terbatas.

3. *Batch size:* 8-16

   Ukuran *batch* ditetapkan pada rentang 8-16 citra per iterasi. Pemilihan *batch size* yang relatif kecil dilakukan untuk menyesuaikan dengan keterbatasan memori serta menjaga kestabilan pelatihan. Selain itu, *batch size* kecil memungkinkan model untuk lebih sering memperbarui bobot sehingga dapat belajar pola visual secara lebih detail.

4. *Epoch:* 50-120

   Jumlah *Epoch* ditentukan pada rentang 50 hingga 120, yang memungkinkan model melakukan proses pembelajaran secara bertahap hingga mencapai performa optimal. Rentang ini dipilih agar model memiliki cukup waktu untuk mempelajari pola visual rokok tanpa mengalami pelatihan berlebihan.

5. *Early stopping* dan *Cosine Annealing Scheduler*

   Mekanisme *Early Stopping* diterapkan dengan parameter *patience* sebesar 25 *Epoch*, di mana pelatihan akan dihentikan otomatis jika nilai *Dice Coefficient* pada data validasi tidak lagi menunjukkan peningkatan. Selain itu, digunakan *Cosine Annealing Scheduler* untuk menurunkan nilai *learning rate* secara berkala hingga mencapai batas minimum. Hal ini memungkinkan model untuk melakukan penyesuaian bobot dengan lebih halus pada tahap akhir pelatihan untuk mencapai akurasi yang lebih presisi.

Dengan konfigurasi parameter tersebut, model *U-Net* diharapkan mampu mempelajari pola visual objek rokok hingga tingkat piksel secara optimal melalui pasangan citra asli dan citra *mask*, serta menghasilkan segmentasi yang lebih akurat dan stabil.

3.2.8 Evaluasi Model

Evaluasi dilakukan untuk menilai performa model dalam melakukan segmentasi. Beberapa matriks yang digunakan antara lain:

1. *Intersection over Union* (*IoU*):

   IoU digunakan untuk mengukur tingkat tumpang tindih antara *mask* hasil prediksi model dan *mask ground truth*. Nilai *IoU* dihitung berdasarkan perbandingan antara area irisan dan area gabungan dari kedua *mask* tersebut. Matriks ini memberikan gambaran seberapa baik model memetakan area objek secara keseluruhan, sehingga sangat relevan untuk menilai ketepatan segmentasi objek rokok yang memiliki ukuran kecil dan bentuk memanjang.

2. *Dice Coefficient:*

   Dice Coefficient digunakan untuk mengukur kesesuaian antara *mask* prediksi dan *mask ground truth* dengan penekanan yang lebih besar pada area objek. Metrik ini sensitif terhadap kesalahan pada objek kecil, sehingga mampu memberikan informasi tambahan mengenai kualitas segmentasi

pada rokok yang memiliki luas area relatif kecil dibandingkan latar belakang.

3. *Accuracy:*

   Accuracy digunakan untuk mengukur tingkat ketepatan prediksi model secara keseluruhan pada tingkat piksel. Nilai *Accuracy* menunjukkan perbandingan antara jumlah piksel yang diklasifikasikan dengan benar terhadap seluruh piksel pada citra. Metrik ini memberikan gambaran umum mengenai kemampuan model *U-Net* dalam membedakan antara area objek rokok dan latar belakang.

4. *Loss training dan validation:*

   Nilai *Loss* pada data pelatihan dan validasi digunakan untuk memantau proses pembelajaran model. Perbandingan antara *Loss training* dan *Loss validation* membantu dalam mendeteksi adanya *overfitting* atau *underfitting* selama pelatihan. Penurunan *Loss* yang stabil menunjukkan bahwa model belajar secara konsisten.

5. Visualisasi hasil prediksi segmentasi:

   Selain matriks numerik, evaluasi juga dilakukan melalui visualisasi hasil segmentasi. *Mask* hasil prediksi dibandingkan secara langsung dengan citra asli dan *mask ground truth* untuk menilai ketepatan bentuk, posisi, dan batas objek secara visual. Visualisasi ini penting untuk memastikan bahwa hasil segmentasi tidak hanya baik secara angka, tetapi juga masuk akal secara visual.

Matriks tersebut digunakan untuk memeriksa ketepatan prediksi *mask*, terutama pada objek kecil dan tipis seperti rokok. Model terbaik dipilih berdasarkan nilai evaluasi tertinggi.

3.2.9 Pengujian Model

Tahapan pengujian dilakukan setelah model *U-Net* mendapatkan hasil evaluasi metrik yang optimal pada proses pelatihan. Tujuan utama dari tahap ini adalah untuk memvalidasi kemampuan model dalam melakukan segmentasi objek rokok pada sekumpulan data baru yang belum pernah dilihat sebelumnya (*unseen* data). Pengujian dilakukan menggunakan data uji (*testing* data) yang telah disiapkan dengan mencakup berbagai variasi kondisi lingkungan nyata, seperti

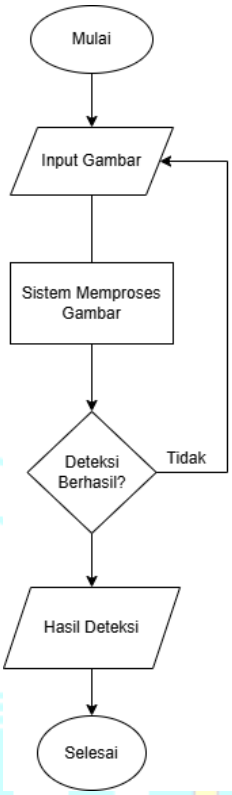
perbedaan intensitas cahaya, kompleksitas latar belakang, keberagaman bentuk fisik objek, hingga sudut pengambilan gambar yang berbeda.

Proses pengujian ini dilakukan dengan membandingkan hasil prediksi *mask* dari model terhadap *ground truth* atau label asli yang telah ditentukan. Melalui tahapan ini, peneliti dapat mengamati sejauh mana model mampu mempertahankan konsistensi dan akurasi segmentasinya terhadap perubahan skenario visual di lapangan. Hasil pengujian ini akan menjadi dasar validasi akhir untuk memastikan bahwa model *U-Net* yang dikembangkan telah memenuhi kriteria keberhasilan dalam mengenali area objek rokok secara tepat.

3.2.10 Implementasi Sistem

Implementasi sistem merupakan tahap penerapan model *U-Net* yang telah dilatih ke dalam sebuah antarmuka aplikasi berbasis *website*. Aplikasi ini dirancang sebagai sistem deteksi rokok otomatis yang memungkinkan pengguna mengunggah gambar atau menggunakan kamera *webcam* secara *real-time* untuk mendeteksi keberadaan rokok dalam citra menggunakan segmentasi berbasis model *UNet + EfficientNet-B0.*

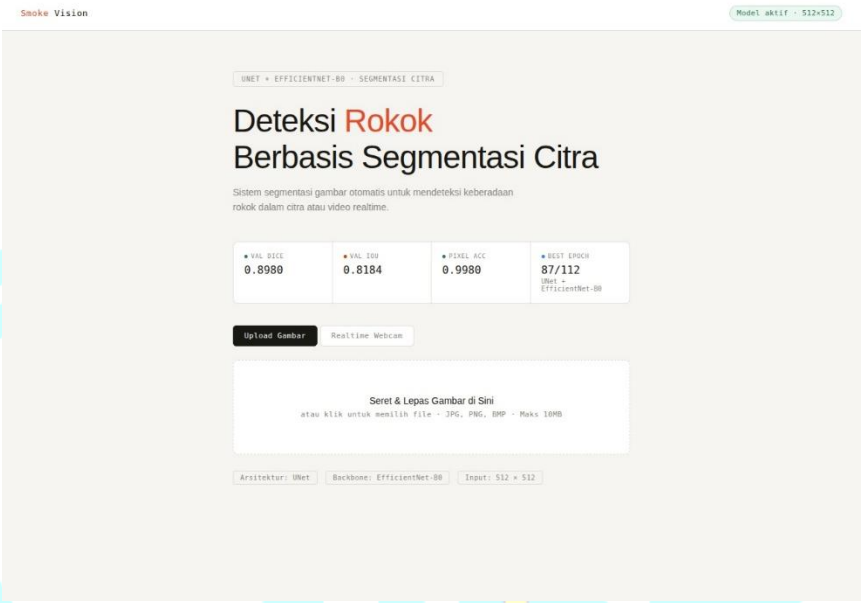
Alur sistem menggambarkan alur kerja keseluruhan aplikasi dari sisi pengguna hingga hasil keluaran sistem. Alur ini mencakup dua mode *input* yang tersedia, yaitu mode unggah gambar (*upload*) dan mode kamera *real-time (webcam)*. Setelah gambar diproses oleh model *U-Net + EfficientNet-B0*, sistem akan menampilkan hasil berupa banner deteksi, gambar *overlay* segmentasi, dan *mask* prediksi. *Flowchart* sistem ditunjukkan pada Gambar 3.3.



Gambar 3. 3 Alur Sistem

1. *Input* Gambar: Pengguna memasukkan gambar ke sistem, baik melalui mode unggah *file* (*drag* & *drop* atau klik) maupun mode kamera *webcam real-time*. Format yang diterima adalah JPG, PNG, dan BMP dengan ukuran maksimal 10MB.
2. Sistem Memproses Gambar: Gambar yang diterima diproses oleh model *U-Net* + *EfficientNet-B0*. Pada tahap ini gambar di-*resize* menjadi 512×512 piksel, dinormalisasi, lalu dimasukkan ke model untuk menghasilkan prediksi *mask* segmentasi pada tingkat piksel.
3. Deteksi Berhasil?: Sistem mengevaluasi *output* model. Jika nilai *confidence* dan *Dice Estimate* memenuhi ambang batas deteksi, sistem menyatakan rokok terdeteksi (Ya). Jika tidak (Tidak), sistem kembali ke tahap *Input* Gambar untuk menerima gambar baru dari pengguna.
4. Hasil Deteksi: Sistem menampilkan hasil berupa banner status (hijau "Tidak Ada Rokok" atau merah "Rokok Terdeteksi"), gambar asli, *overlay* segmentasi, *mask* prediksi hitam-putih, serta metrik evaluasi (*Confidence*, *Dice* Est., waktu inferensi, dan ukuran gambar).

Untuk memberikan gambaran implementasi sistem yang telah dibangun, berikut ditampilkan antarmuka (UI/UX) dari *website* deteksi rokok berbasis segmentasi citra ada pada gambar 3.4.



Gambar 3. 4 Desain Tampilan *Website*

Berdasarkan tampilan pada gambar di atas, halaman utama sistem dirancang dengan konsep minimalis dan modern untuk memberikan pengalaman pengguna yang nyaman dan mudah dipahami. Pada bagian atas terdapat *header* bertuliskan *Smoke Vision* sebagai identitas aplikasi, serta informasi status model yang aktif dengan ukuran *input* 512×512 piksel.

Di bagian tengah halaman ditampilkan judul utama “Deteksi Rokok Berbasis Segmentasi Citra” yang menegaskan fungsi sistem, disertai deskripsi singkat mengenai kemampuan sistem dalam mendeteksi keberadaan rokok pada gambar maupun video secara *real-time*. Tepat di bawahnya terdapat informasi performa model, seperti nilai *Dice*, *IoU*, *Pixel Accuracy*, jumlah *epoch* terbaik, serta arsitektur model yang digunakan yaitu *U-Net + EfficientNet-B0*.

Selanjutnya, pengguna diberikan dua pilihan metode *input*, yaitu tombol *Upload* Gambar dan *Realtime Webcam*, yang memudahkan pengguna dalam memilih jenis deteksi yang diinginkan. Pada bagian bawahnya tersedia area *input* berbentuk kotak *drag* & drop yang memungkinkan pengguna untuk menyeret dan

melepaskan gambar atau memilih *file* secara manual. Sistem mendukung format JPG, PNG, dan BMP dengan ukuran maksimum 10MB.

Secara keseluruhan, tampilan halaman utama ini dirancang untuk memberikan alur penggunaan yang jelas, sederhana, dan interaktif, sehingga pengguna dapat langsung memahami cara menggunakan sistem tanpa memerlukan panduan tambahan.

3.3 Hasil Yang Diharapkan

Penelitian ini diharapkan menghasilkan model segmentasi citra rokok berbasis arsitektur *U-Net* yang mampu melakukan segmentasi objek rokok secara akurat hingga tingkat piksel. Model yang dikembangkan diharapkan mampu mengenali karakteristik rokok yang berukuran kecil dan tipis dengan tetap mempertahankan detail objek melalui mekanisme *skip-connection* pada arsitektur *U-Net*.

Keberhasilan penelitian ini diukur berdasarkan kemampuan model dalam membedakan area objek rokok dari latar belakangnya secara presisi. Kinerja tersebut akan dievaluasi menggunakan beberapa metrik utama, yaitu *Accuracy*, *Dice Coefficient*, dan *Intersection over Union (IoU)*. Model diharapkan menunjukkan nilai yang optimal pada ketiga metrik tersebut sebagai bukti bahwa proses pembelajaran telah berjalan dengan baik.

Luaran akhir dari penelitian ini adalah sebuah model segmentasi yang konsisten dan tangguh dalam mengenali objek rokok pada berbagai variasi kondisi citra. Hasil ini nantinya akan divalidasi melalui tahap pengujian menggunakan data uji untuk memastikan bahwa model tidak hanya bekerja dengan baik pada data pelatihan, tetapi juga memiliki kemampuan generalisasi yang kuat terhadap data baru.