

BAB III

METODE PENELITIAN

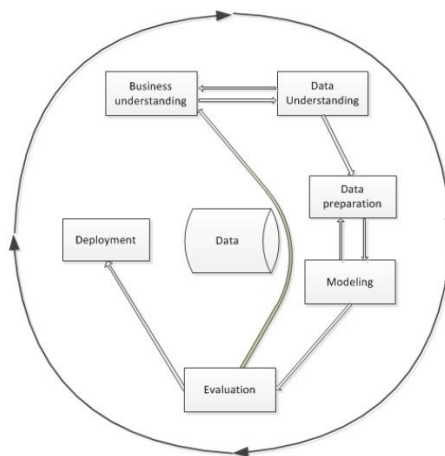
3.1. Jenis Penelitian

Jenis penelitian ini adalah penelitian studi kasus dan eksperimental. Penelitian studi kasus adalah pendekatan ilmiah mendalam pada suatu peristiwa atau aktivitas yang sedang berlangsung dengan tujuan memahami aktivitas tersebut dalam konteks nyata (Rahardjo & Si, 2017). Penelitian eksperimental bertujuan untuk menetapkan hubungan sebab dan akibat antara dua variabel. Dalam penelitian eksperimental, peneliti mengendalikan variabel bebas untuk menemukan pengaruh atau perubahan pada variabel terikat (Ratminingsih, 2010).

Studi kasus difokuskan pada objek spesifik, yaitu sudut pandang CCTV di Sumber Aren, untuk menganalisis performa model YOLOv3 dalam konteks tersebut. Sementara itu, penelitian eksperimental dilakukan untuk menetapkan hubungan sebab-akibat antara performa model dan beberapa hyperparameter yang digunakan selama pelatihan YOLOv3, guna memahami bagaimana perubahan dalam hyperparameter mempengaruhi hasil deteksi objek pada data yang diteliti.

3.2. Desain Penelitian

Desain penelitian yang digunakan adalah CRISP-DM (*Cross Industry Standard Process for Data Mining*), merupakan metodologi penelitian yang umum digunakan dalam industri swasta dan teknologi. Metodologi ini terdiri dari enam fase yang berulang:

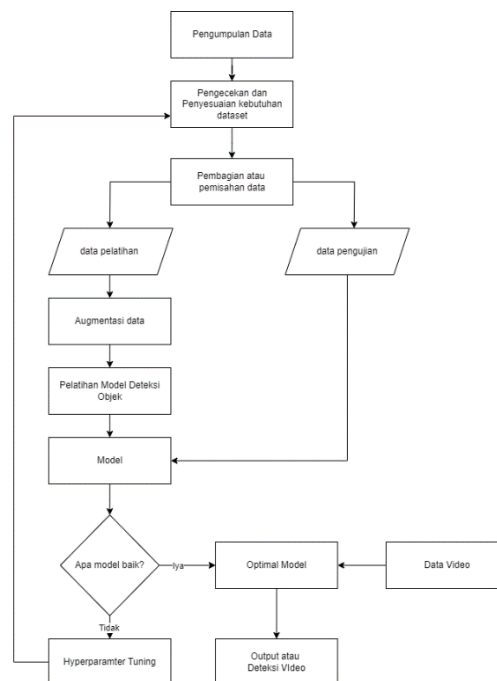


Gambar 3.1 CRISP-DM (Cross Industry Standard Process for Data Mining)

- a. Pemahaman Bisnis (*Bussiness Understanding*): Memahami objektif analisis data dan menerapkannya secara subjektif untuk memenuhi kebutuhan bisnis.
 - b. Pemahaman Data (*Data Understanding*): Mengumpulkan data, menganalisisnya, dan mengevaluasi kualitasnya.
 - c. Penyiapan Data (*Data Preparation*): Mengolah data awal menjadi dataset final.
 - d. Pembuatan Model (*Modelling*): Memilih teknik pemodelan, algoritma, dan parameter optimal untuk mencapai kriteria kualitas model yang diinginkan.
 - e. Evaluasi (*Evaluation*) : Menganalisis dan memperbaiki model sesuai dengan target yang ditetapkan.
 - f. Deployment: Menyajikan hasil dari model kepada pengguna (Niakšu, 2015).
- Metodologi ini membantu dalam proses analisis data dan pengembangan model yang berkualitas.

3.3. Desain Sistem

Desain Penelitian pada penelitian ini dapat dilihat pada gambar 3.2 yaitu mengenai alur atau proses yang dilakukan untuk melatih model deteksi objek YOLOv3 serta menentukan pengaruh *hyperparameter*-nya.



Gambar 3.2 Desain Penelitian

Berikut rincian mengenai beberapa komponen dari desain penelitian yang ada pada gambar 3.2.

1. Pengumpulan Data

Pada tahap ini, data dikumpulkan dari CCTV di lokasi Sumber Aren. Pengumpulan data dilakukan dengan mengambil klip rekaman video dari CCTV. Klip video ini kemudian dipotong per detik untuk menghasilkan potongan-potongan gambar yang bisa dianalisis lebih mendalam.

2. Pengecekan dan Penyesuaian Kebutuhan Data

Setelah pengumpulan data, langkah ini melibatkan pengecekan format label data yang harus sesuai dengan format YOLO. Selain itu, dataset yang tidak relevan atau tidak berguna dihapus, dan dataset tambahan yang diperlukan ditambahkan untuk memperkaya data.

3. Pemisahan Data

Data yang telah dikumpulkan dan disiapkan kemudian dipisahkan menjadi dua subset: 80% untuk data pelatihan (training) dan 20% untuk data pengujian (testing). Pemisahan ini penting untuk memastikan bahwa model dilatih dengan data yang cukup dan diuji dengan data yang belum pernah dilihat sebelumnya.

4. Data Augmentasi

Augmentasi data dilakukan selama proses pelatihan model tanpa menambah data secara langsung. Ini dapat mencakup teknik seperti rotasi, pemotongan, atau perubahan warna pada gambar untuk meningkatkan keberagaman data dan membantu model belajar dari berbagai variasi data.

5. Modelling

Pada tahap ini, model deteksi objek dibangun dan dilatih menggunakan data pelatihan. Selama pelatihan, berbagai parameter model dievaluasi untuk menghasilkan bobot model. Setelah model mencapai performa yang memuaskan, bobot model disimpan untuk penggunaan selanjutnya.

6. Evaluasi model

Model yang telah dilatih dan bobotnya disimpan kemudian dievaluasi menggunakan data pengujian. Evaluasi ini bertujuan untuk menilai kinerja model dengan metrik seperti precision, recall, dan F1-score. Berdasarkan hasil evaluasi, keputusan diambil apakah model sudah optimal atau perlu penyesuaian.

7. Hyperparameter tuning

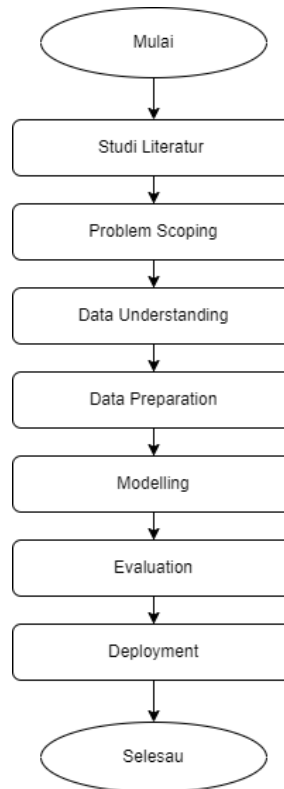
Jika model tidak mencapai performa yang diinginkan selama evaluasi, proses penyesuaian hyperparameter dilakukan. Ini termasuk mengubah parameter seperti *learning rate*, ukuran *batch*, atau jumlah *epoch* untuk meningkatkan performa model.

8. Deployment pada Video

Setelah model mencapai performa yang memuaskan, model yang telah dilatih dideploy pada video untuk aplikasi praktis. Ini berarti model akan diterapkan pada video nyata dari CCTV untuk mendeteksi dalam rekaman.

3.4. Prosedur Penelitian

Studi ini menerapkan kerangka kerja CRISP-DM sesuai dengan desain penelitian sebelumnya yang telah diuraikan. Ilustrasi alur penelitian tersedia pada Gambar 3.3.



Gambar 3.3. Prosedur Penelitian

9. Studi Literatur

Pada tahap awal penelitian, peneliti melakukan tinjauan literatur dengan mengumpulkan dan menganalisis sumber-sumber terkait. Hal ini dilakukan untuk memahami konteks, teori, dan penemuan sebelumnya, serta untuk mengidentifikasi kurangnya pengetahuan dan merumuskan masalah penelitian. Tinjauan literatur juga memberikan wawasan tentang metode yang telah digunakan oleh peneliti lain dalam penelitian serupa.

10. *Problem Scoping*

Tahapan ini adalah langkah yang disusun secara terstruktur untuk menguraikan elemen-elemen penting, seperti apa, siapa, di mana, mengapa, dan bagaimana. Tujuannya adalah untuk menetapkan subjek yang jelas,

mengidentifikasi masalah yang perlu diselesaikan, menentukan tempat di mana masalah itu muncul, mengungkapkan alasan mengapa perlu mengatasi masalah tersebut, dan merencanakan strategi untuk menyelesaikannya. Dalam penelitian ini, masalah yang ditangani adalah mengenai keamanan pengawasan CCTV di industri. Adapun beberapa hal yang dilakukan dalam tahapan ini yaitu melakukan wawancara dan penggunaan metode 4W.

11. *Data Understanding*

Tahapan ini meliputi pencarian atau pengumpulan data, analisis data, dan evaluasi untuk memastikan data digunakan sesuai kebutuhan model. Pada penelitian ini, yang digunakan adalah video rekaman CCTV. Penting untuk memahami karakteristik data dan bagaimana data tersebut dapat memenuhi tujuan model.

12. *Data Preparation*

Selanjutnya adalah *pre-processing data* atau *data preparation* yang melibatkan pengkonversian format video jika perlu, segmentasi video menjadi bagian-bagian yang relevan, objek, pembersihan data untuk memastikan kualitasnya, serta pemrosesan tambahan seperti peningkatan kualitas gambar atau pengurangan *noise*. Selain itu, pembagian data menjadi subset yang sesuai dan, jika diperlukan, augmentasi data juga dilakukan untuk memperkaya variasi data.

13. *Modelling*

Data yang dikumpulkan akan digunakan untuk membentuk model atau akan digunakan untuk tahapan ini yaitu tahapan pelatihan. Data tersebut dibagi menjadi 2 yaitu *training data* dan *testing data*. *Data training* akan digunakan pada tahapan training model menggunakan arsitektur YOLOv3 yang merupakan pengembangan dari algoritma CNN dengan python.

14. Evaluasi

Setelah tahap pelatihan, langkah evaluasi dapat dilaksanakan. Evaluasi pada deteksi objek bertujuan untuk menilai seberapa baik model dalam mengidentifikasi dan memprediksi objek pada gambar. Evaluasi umumnya dilakukan setelah model dilatih dengan *training data* (data pelatihan) dan sebelum model diaplikasikan pada tugas deteksi objek praktis.

15. *Deployment*

Tahap ini merupakan langkah dimana model yang telah dilatih akan dideploy atau diimplementasikan pada rekaman video CCTV.

3.5. Populasi dan sampel

Populasi dalam penelitian ini adalah seluruh rekaman visual CCTV CV Sumber Aren Sampelnya terdiri dari sebagian rekaman video yang dihasilkan sejumlah kamera sistem pemantauan CCTV tersebut. Pengambilan data, dilakukan selama 2 bulan pada pertengahan bulan Juni hingga pbulan Agustus.

3.6. Teknik pengumpulan data

Data yang digunakan dalam penelitian ini berasal dari rekaman video CCTV di lokasi produksi CV Sumber Aren. Teknik pengumpulan data yang digunakan termasuk dalam kategori data primer karena diambil langsung dari sumber aslinya. Langkah-langkah pengumpulan data meliputi perekaman video CCTV secara terus-menerus, pemilihan klip video yang relevan, proses screenshot, dan labelling objek dalam gambar menggunakan makesense.ai, merupakan platform yang berguna untuk melakukan anotasi data pada gambar atau video. Setelah itu melakukan penyimpanan gambar yang telah dilabeli sebagai *data training* dan *testing* untuk melatih model deteksi objek menggunakan algoritma Convolutional Neural Network (CNN).

Pada penelitian menggunakan YOLOv3 format label yang digunakan adalah YOLO format, yang terdiri dari informasi tentang kelas objek, serta koordinat dan ukuran bounding box yang dinormalisasi. Setiap baris dalam file label yang menggunakan format YOLO (*You Only Look Once*) mendefinisikan objek dengan beberapa komponen, yaitu

```
<class_id> <x_center> <y_center> <width> <height>
```

Gambar 3.4 format label YOLO

Berikut informasi komponen label dengan format YOLO.

- a. *class_id*: Nomor yang menunjukkan kelas objek
- b. *x_center* dan *y_center*: Koordinat pusat dari bounding box atau objek pada sumbu x dan y. Nilai ini dinormalisasi menjadi angka antara 0 hingga 1 relatif terhadap ukuran gambar.

- c. *Width dan height*: Lebar dan tinggi bounding box yang dinormalisasi relatif ukuran gambar, dengan nilai berkisar antara 0 hingga 1.

Satu gambar dapat berisi beberapa objek, sehingga file .txt anotasi menyertakan beberapa baris untuk mendeskripsikan setiap objek. Format ini memfasilitasi model YOLO dalam mengenali dan mendeteksi objek berdasarkan informasi dalam file label, yang terhubung dengan gambar melalui nama yang sama.

3.7. Teknik Analisis Data

3.7.1. Metrik Evaluasi

Metrik evaluasi yang sering digunakan untuk melakukan pengukuran seberapa akurat algoritma deteksi objek adalah dengan menggunakan perbandingan prediksi *bounding box* dan *bounding box ground-truth*. Kemampuan deteksi dinilai menggunakan *intersection over union* (IOU) yang melakukan pengukuran dari seberapa besar ukuran tumpang tindih antara prediksi *bounding box* dengan *bounding box ground-truth*. Nilai IOU yang besar menunjukkan deteksi yang lebih akurat.

$$J(B_p, B_{gt}) = IOU = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})}$$

B_p merupakan nilai prediksi dan B_{gt} nilai *ground truth*. Berikut ilustrasi dari *intersection of union*.

$$IOU = \frac{\text{area of overlap}}{\text{area of union}} = \frac{\text{[Diagram: Two overlapping rectangles, one green and one red, with their intersection shaded blue. Below them is a single blue shape representing the union of the two rectangles.]}}{\text{[Diagram: Two overlapping rectangles, one green and one red, with their intersection shaded blue. Below them is a single blue shape representing the union of the two rectangles.]}}$$

Gambar 3.5 Intersection Over Union (Padilla dkk., 2021)

Pengaturan *threshold* untuk ambang batas tertentu dapat menjadi sebuah metrik dalam menilai benar atau salah. Deteksi yang memiliki IOU dengan nilai kecil di bawah ambang batas akan dianggap deteksi yang salah dan nilai IOU yang lebih besar dari *threshold* dianggap sebagai benar. IOU biasa dinyatakan dengan menggunakan persentase, seperti nilai 50% dan 75% (Padilla dkk., 2021).

3.7.1.1. Precision dan Recall

Untuk menghitung nilai *precision* dan *recall*, tiap *bounding box* prediksi perlu di klasifikasi menjadi.

- a. *True positive (TP)*, deteksi yang benar atau sesuai dengan *ground-truth bounding box* berdasarkan *threshold* atau ambang batas IOU
- b. *False positive (FP)*, deteksi yang salah dimana *bounding box* prediksi tidak sesuai dengan *bounding box ground-truth* atau prediksi untuk objek yang tidak ada.
- c. *False Negative (FN)*, tidak ada prediksi *ground-truth bounding box* oleh model atau tidak ada prediksi yang cocok (Padilla dkk., 2021)

Precision mengukur seberapa baik model mengidentifikasi sebuah objek yang relevan atau persentase prediksi positif yang benar (Padilla, 2020).

$$precision = \frac{TP}{TP + FP} = \frac{TP}{all\ detections}$$

Recall melakukan deteksi untuk menemukan seberapa baik model menemukan seluruh objek yang relevan, atau persentase prediksi positif yang benar dari semua groundtruth yang ada (Padilla dkk., 2020).

$$recall = \frac{TP}{TP + FN} = \frac{TP}{all\ ground - truth}$$

3.7.1.2. F1-Score

$$F_1 = 2 \frac{precision \times recall}{precision + recall}$$

F1-score merupakan rata-rata harmonik dari *precision* dan *recall* model deteksi. Nilai yang di dapat dari *F1-score* berada pada rentang 0 dan 1. *F1-score* tidak digunakan untuk mempertimbangkan model deteksi dengan berbagai nilai *confidence threshold* sehingga hanya digunakan untuk *confidence threshold* tetap (Padilla dkk., 2021).

3.7.1.3. Precision-Recall Curve (PRC)

Kurva *Precision-Recall* menggambarkan *trade-off* antara *precision* dan *recall* untuk berbagai nilai *confidence* dari detector objek *Precision* yang tinggi, akan mengahailkan nilai FP yang sedikit tetapi, akan ada beberapa objek yang mungkin terlewat (FN) mengakibatkan rendahnya nilai *recall*. Sedangkan *recall*

yang tinggi akan menemukan banyak objek *ground truth* tetapi nilai dari FP akan mengingkat yang juga meningkatkan nilai *precision*.

Model yang sangat ideal akan melakukan deteksi dan menemukan seluruh objek *ground truth* (FN = 0), dan hanya menemukan objek yang relevan (FP = 0). Model deteksi yang baik akan memiliki *precision* yang meningkat atau tetap tinggi ketika *recall* meningkat yang menunjukkan baik *precision* atau *recall* tetap baik meskipun *confidence threshold* diubah.

Area Under the Curve (AUC) tinggi, pada kurva PRC akan menunjukkan bahwa model deteksi memiliki *precision* dan *recall* yang seimbang. Kasusnya seringkali, kurva yang didapatkan berbentuk zig-zag, yang menyulitkan dalam pengukuran AUC sehingga PRC memproses kurvanya untuk menghilangkan zig-zag sebelum perhitungan AUC (Padilla dkk., 2020).

3.7.1.4. *Average Precision (AP) dan Mean Average Precision (mAP)*

Hasil deteksi objek dari model biasanya memberikan nilai *bounding box*, kelas, dan tingkat *confidence*. Untuk menghitung *precision* dan *recall*, diperlukan *threshold confidence* yang menentukan apakah deteksi dianggap positif (jika di atas *threshold*) atau negatif (jika di bawah *threshold*) (Padilla dkk., 2021).

Nilai *precision* dan *recall* yang dihasilkan dari berbagai *threshold* ini kemudian digunakan untuk menghitung *Average Precision (AP)*. Ada dua metode utama untuk menghitung AP, *11-point interpolation* dan *all-point interpolation*. Kedua metode ini menginterpolasi nilai kurva *precision-recall* pada titik-titik tertentu untuk menghitung rata-rata *precision* pada berbagai tingkat *recall*, mendapatkan estimasi nilai AP yang lebih stabil dan akurat (Padilla, 2020).

MAP merupakan matriks yang digunakan untuk mengukur kinerja model. Nilai tersebut didapat dari rata-rata *average precision* di semua kelas. AP merupakan nilai rata-rata presisi yang dicapai pada berbagai tingkat *recall*. Nilai AP menggambarkan seberapa baik model mendeteksi dan mengklasifikasikan objek. (Rafael dkk., 2020).

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

3.8. Lingkungan Komputasi

Penelitian ini didukung dengan beberapa alat pendukung dalam bentuk perangkat (hardware) keras maupun perangkat lunak (*software*):

1. *Hardware*

Kebutuhan *Hardware* yang dipakai antara lain adalah perangkat komputer atau laptop. Spesifikasi perangkat komputer atau laptop yang digunakan dilampirkan pada tabel dibawah:

- a. *Windows* 11 Pro 64-bit
- b. 11th Gen Intel(R) Core(TM) i5-1135G7
- c. HDD 350 GB
- d. Xe Graphics
- e. Hi-Look

2. *Software*

Selama penelitian, software pendukung yang digunakan sebagai berikut:

- a. Bahasa pemrograman Python Versi 3.10.9
- b. Google Colaboratory Pro
- c. *Visual Studio Code*
- d. Streamlit and ngrok
- e. Makesense.ai
- f. Google chrome