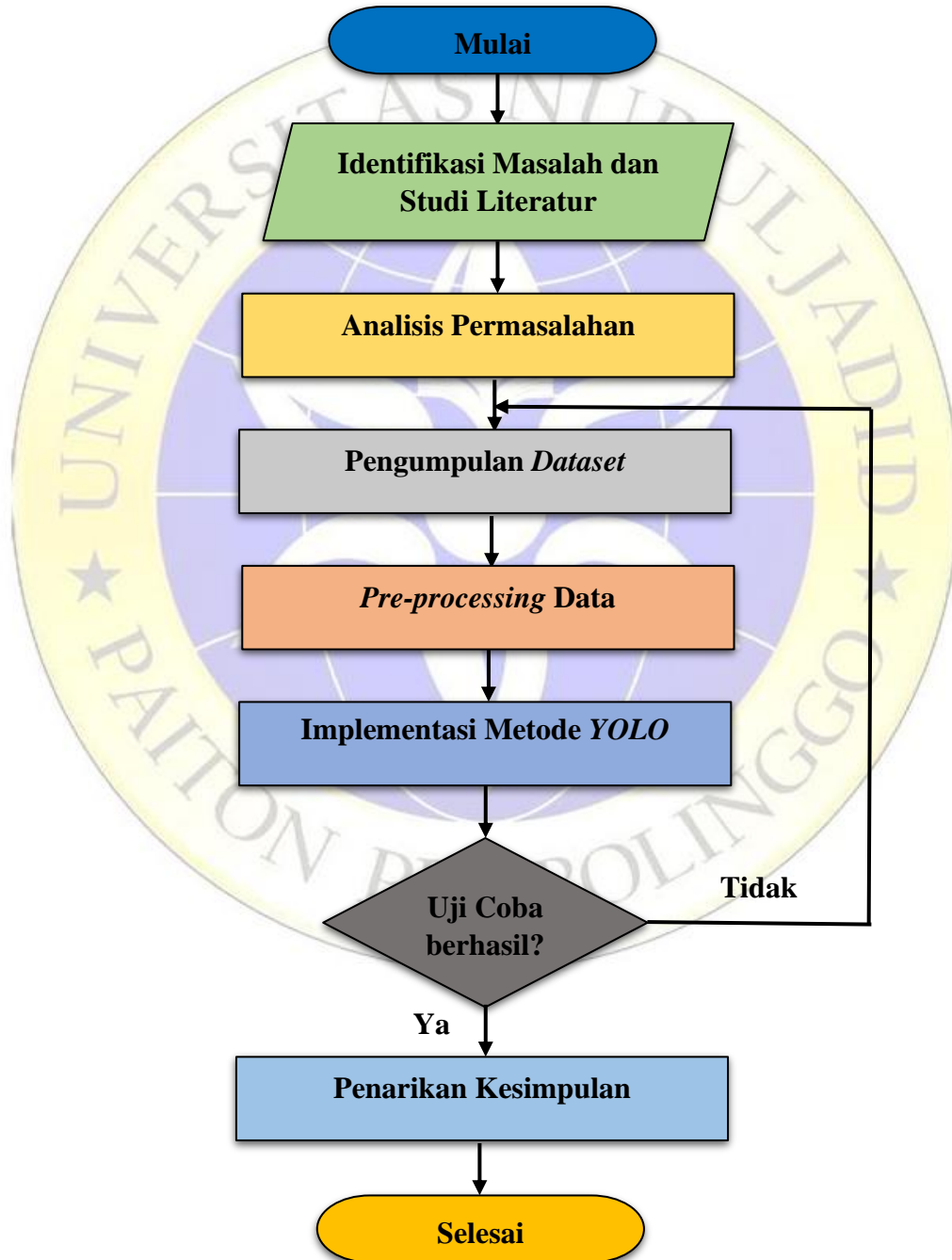


BAB III

METODE PENELITIAN

3.1. Rancangan Penelitian

Rancangan penelitian dapat mempermudah proses penelitian untuk mendapatkan hasil penelitian dengan proses yang sistematis. Tahapan penelitian yang dilakukan tergambar dalam Diagram Alir Gambar 3.1.



Gambar 3.1. Diagram Alir Penelitian

3.2. Tahap Penelitian

Tahapan penelitian menjelaskan lebih rinci terkait tahapan penelitian yang dilakukan dari awal sampai akhir penelitian. Berdasarkan rancangan penelitian yang tergambar dalam Gambar 3.1, maka tahapan penelitian dilakukan dari identifikasi masalah dan studi literatur, analisis permasalahan, pengumpulan *dataset*, *pre-processing* data, implementasi dengan metode *You Only Look Once* (YOLO), uji coba, hingga mendapatkan hasil untuk penarikan kesimpulan.

3.2.1. Identifikasi Masalah dan Studi Literatur

Tahapan identifikasi dilakukan untuk mengetahui permasalahan yang mendorong adanya penelitian ini dilakukan. Studi literatur dilakukan untuk mengetahui pemahaman konsep, teori, dan metode yang digunakan dalam melakukan penelitian.

3.2.2. Analisis Permasalahan

Setelah identifikasi permasalahan didapatkan, maka analisis permasalahan dilakukan untuk mengetahui rumusan masalah pada penelitian yang dilakukan dan juga untuk mengetahui beberapa kebutuhan yang digunakan selama penelitian dilakukan.

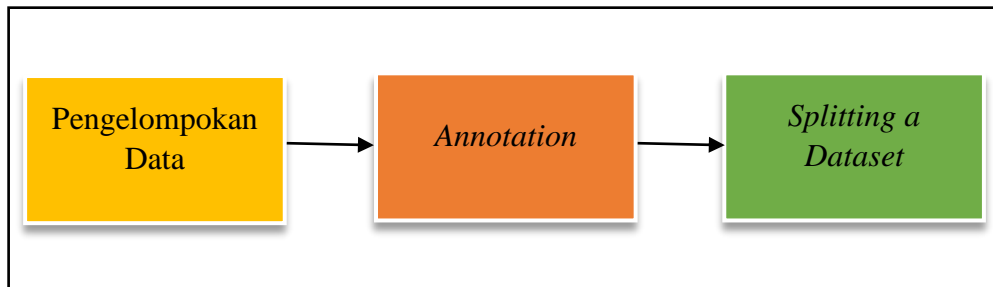
3.2.3. Pengumpulan *Dataset*

Dataset berisikan sekumpulan gambar para pekerja yang terlibat dalam bidang konstruksi. *Dataset* yang digunakan dalam penelitian ini merupakan *dataset* yang didapatkan langsung melalui sumber penyedia *dataset* berbasis *online*, yaitu <https://roboflow.com/>. Akses cepat untuk mendapatkan *dataset*, dapat diakses melalui halaman berikut ini <https://public.roboflow.com/object-detection/hard-hat-workers>. Kumpulan gambar terdiri dari pekerja yang sedang menggunakan APD berupa *helmet* (pelindung kepala), *vest* (rompi), dan juga tidak sedang menggunakan APD. Tersedia 7.041 *dataset* gambar dan anotasi dengan berbagai dimensi gambar, dari 179 x 270 pixels hingga 640 x 959 pixels yang tersimpan dalam format *.jpg.

3.2.4. *Pre-processing* Data

Tahapan *pre-processing* data dilakukan untuk memudahkan proses pengolahan citra, sebelum diimplementasikan kepada metode yang diusulkan.

Beberapa tahapan *pre-processing* data terdiri dari pengelompokan data, *annotation*, dan *splitting a dataset*. Gambar 3.2 merupakan tahapan *pre-processing* data.



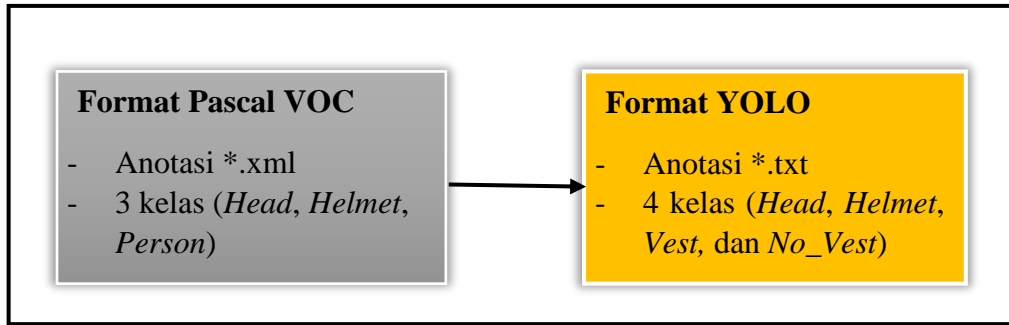
Gambar 3.2. *Pre-processing* data

3.2.4.1. Pengelompokan Data

Dari 7.041 *dataset* yang didapatkan melalui *Roboflow*, maka dari data tersebut didapatkan sebanyak 1.173 data gambar secara acak yang akan digunakan sebagai *object* citra untuk diimplementasikan kepada metode usulan, untuk mendeteksi objek *head*, *helmet*, *vest* dan *no_vest* yang ada pada gambar tersebut.

3.2.4.2. *Annotation*

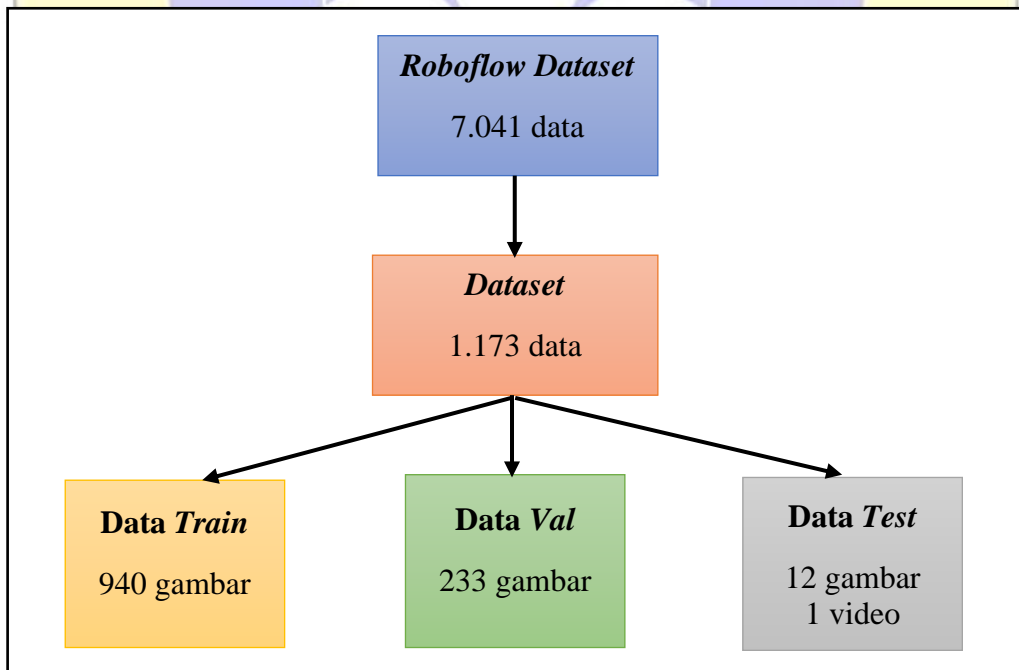
Anotasi merupakan proses pelabelan gambar sesuai kebutuhan kelas yang akan digunakan dalam suatu penelitian. *Dataset* yang diperoleh dari *Roboflow* telah memiliki anotasi dengan kelas *head* (kepala), *helmet* (pelindung kepala), dan *person* (orang) untuk setiap gambar yang tersedia dalam *Hard Hat Workers Dataset* yang dapat diakses melalui halaman yang telah tersedia dalam tahapan pengumpulan *dataset*. Dalam penelitian ini, ada empat kelas yang telah ditentukan yang terdiri dari *head* (kepala), *helmet* (pelindung kepala), *vest* (rompi) dan *no_vest* (tidak berrompi). Kelas *helmet* atau *vest* merepresentasikan pekerja sedang menggunakan Alat Pelindung Diri (APD), sedangkan kelas *head* atau *no_vest* merepresentasikan para pekerja sedang tidak menggunakan Alat Pelindung Diri (APD), sehingga kelas *person* dihapus. Proses anotasi gambar dalam penelitian ini menggunakan alat untuk anotasi gambar yang bernama *LabelImg*. Gambar 3.3. merupakan proses anotasi gambar dari format *.xml menjadi format *.txt.



Gambar 3.3. Anotasi Gambar

3.2.4.3. *Splitting a Dataset*

Splitting (membagi) merupakan tahap pembagian *dataset* menjadi data *training* (data latih) dan data *validation* (data validasi) dengan perbandingan 80% : 20% dari 1.173 *dataset*. Selain menyiapkan data *training* dan *validation*, maka juga diperlukannya data uji coba untuk menguji metode yang telah digunakan setelah adanya proses pelatihan data terhadap metode tersebut, oleh karena itu dalam membagi data juga dibutuhkan data *testing* atau data uji. Data uji merupakan data selain 1.173 *dataset* sebelumnya, dengan jumlah data uji terdiri sebanyak 12 data gambar dan 1 video. Gambar 3.4. merupakan proses dari pembagian data.



Gambar 3.4. *Splitting a Dataset*

3.2.5. Implementasi Metode *You Only Look Once* (YOLO)

Tahapan ini merupakan pembuatan program dalam mendeteksi empat kelas yang telah ditentukan sebelumnya yaitu, *head*, *helmet*, *vest*, dan, *no_vest* dengan menggunakan metode *You Only Look Once* (YOLO) versi 5. Program ini disusun menggunakan *Google Colaboratory*, dengan tahapan *setup YOLOv5*, memanggil *dataset* dalam *drive*, proses *training* dataset, dan *testing* data. Semua tahapan tersebut dimulai dari penyusunan program sampai uji coba program dilakukan menggunakan perangkat ASUS dengan spesifikasi prosesor Intel(R) Core™ i5-7200U CPU @ 2.50GHz 2.71GHz, dengan sistem operasi Windows 10 Pro 64-bit, dan penyimpanan 8,00 GB RAM.

Program dimulai dengan *Setup YOLOv5* dengan *clone YOLOv5* di *Google Colaboratory*, lalu memasang *drive* agar *dataset* yang ada pada *drive* dapat terbaca untuk diproses menggunakan metode *YOLOv5*, lalu *unzip dataset* karena *dataset* yang tersedia masih dalam bentuk *.zip. Setelah itu melatih *dataset* dengan memanggil *train.py* yang telah tersedia di *YOLOv5*. Sebelum proses *training* dilakukan, perlu memanggil data *train* dan *val* yang telah disiapkan terlebih dahulu dan merubah jumlah kelas menjadi 4 dan nama kelas menjadi *head*, *helmet*, *vest*, dan *no_vest* disesuaikan dengan urutan dan penulisan kelas ketika dilakukan anotasi atau dapat diakses di *classes.txt*. Kemudian sesuaikan nilai parameter yang ada (*batch* dan *epochs*) untuk menemukan parameter metode yang baik, lalu latih data dengan model *yolo5s.pt* dengan menggunakan fungsi dari *train.py*. Hasil *train* tersebut akan tersimpan di dalam *yolov5/runs/train/*. Dalam melakukan *testing* atau uji coba cukup memanggil *best.pt* atau model yang telah didapatkan setelah melatih *dataset*, dengan menggunakan *detect.py* dengan menyertakan data yang akan diuji coba, sehingga hasil uji coba dapat dilihat melalui *yolov5/runs/detect/*.

3.2.6. Uji Coba

Uji coba dilakukan untuk mengetahui tingkat keberhasilan dari metode yang diusulkan. Uji coba dilakukan pada 12 data gambar dan 1 data berupa video yang didapatkan melalui *youtube* dengan kata kunci “konstruksi” dengan durasi video 1 menit 29 detik.

Pada proses uji coba, keberhasilan dalam pengujian metode *You Only Look Once* (YOLO) dalam *monitoring* penggunaan Alat Pelindung Diri (APD), dinilai berdasarkan tingkat akurasi, presisi, sensitifitas, dan nilai kesalahan deteksi (*loss*) dalam mendeteksi objek yang telah ditentukan kedalam empat kelas. Perhitungan akurasi, presisi, dan sensitifitas dilakukan menggunakan perhitungan *Confusion Matrix*. Perlu adanya pemahaman terkait 4 variabel kemungkinan dari hasil deteksi yang terangkum pada Tabel 3.1.

Tabel 3.1. *Confusion Matrix*

<i>Predicted Values</i>	<i>Actual Values</i>	
	<i>Positive</i>	<i>Negative</i>
<i>Positive</i>	TP	FP
<i>Negative</i>	FN	TN

- a. *True Positive* (TP) atau Benar Positif, jumlah prediksi yang benar dalam kelas positif.
- b. *False Positive* (FP) atau Salah Positif, jumlah prediksi yang salah dalam kelas positif.
- c. *False Negative* (FN) atau Salah Negatif, jumlah prediksi yang salah dalam kelas negatif.
- d. *True Negative* (TN) atau Benar Negatif, jumlah prediksi yang benar dalam kelas negatif.

Persamaan dalam menentukan nilai akurasi, presisi, dan sensitifitas dijelaskan berdasarkan pada Tabel 3.1.

- a. Akurasi (*Accuracy*)

Rasio dari prediksi yang bernilai benar (positif dan negatif) dengan keseluruhan data yang diuji. Persamaan tersebut seperti pada Persamaan 3.1.

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad \text{-----} \quad (3.1)$$

b. Presisi (*Precision*)

Rasio dari prediksi benar positif dibandingkan dengan keseluruhan hasil yang terdeteksi dengan nilai positif. Persamaannya seperti pada Persamaan 3.2.

$$Precision = \frac{TP}{TP+FP} \text{ -----} \quad (3.2)$$

c. Sensitifitas (*Recall*)

Rasio dari prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif pada data sebenarnya. Persamaannya seperti pada Persamaan 3.3.

$$Recall = \frac{TP}{TP+FN} \text{ -----} \quad (3.3)$$

3.2.7. Penarikan Kesimpulan

Tahapan penarikan kesimpulan merupakan tahapan akhir setelah seluruh rancangan metode penelitian dilakukan. Dengan adanya tahapan ini maka didapatkan kesimpulan untuk metode yang diuji dalam menentukan metode tersebut dapat dikembangkan dalam mendeteksi objek kedalam empat kelas yang digunakan atau sebaliknya, sehingga dengan adanya tahapan ini para peneliti selanjutnya dapat mengetahui langkah selanjutnya yang akan dilakukan dalam mengembangkan penelitian yang telah dilakukan saat ini.