

BAB III METODE PENELITIAN

3.1. Kerangka Penelitian

Pada metode penelitian dijelaskan langkah-langkah atau tahapan yang dilakukan pada setiap tahap saat penyelesaian dalam proses penelitian berat buah apel fuji pada citra digital dan tahapan penelitian ini dapat dijelaskan lebih rinci. Adapun rancangan pada penelitian ini sebagai berikut :

3.1.1. Rancangan Data Training

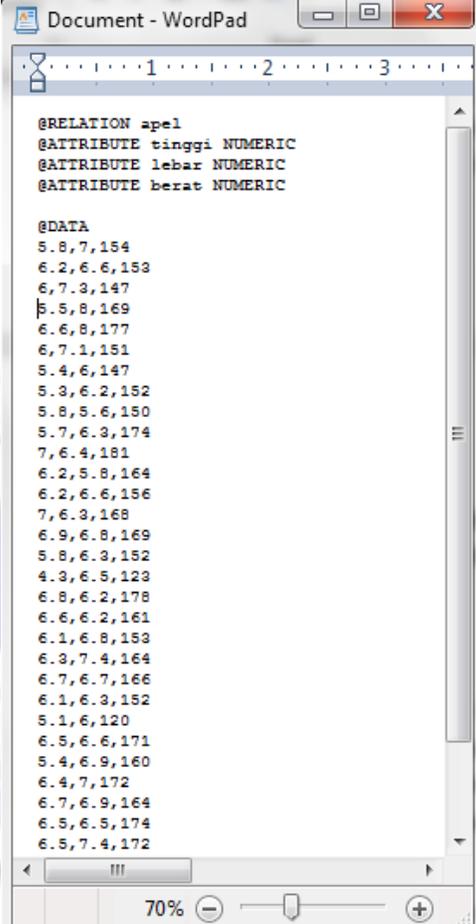
Data *training* adalah latihan, jadi machine learning akan melatih atau belajar dari data yang diberikan untuk dapat memahami informasi pada data tersebut.



Gambar 3.1 Rancangan Data Testing

3.1.1.1. Pengumpulan DataSet

Data yang diambil yaitu 30 buah apel fuji diukur tingginya dengan cara manual menggunakan penggaris, lalu diukur juga lebar buah apel fuji menggunakan penggaris. Setelah itu buah apel fuji ditimbang menggunakan timbangan digital. Kemudian dicatat di *wordpad* data yang sudah didapat dari hasil pengukuran menggunakan penggaris dan timbangan digital. Kemudian di *save* dengan format *.arff. Tujuannya agar dilanjut keproses untuk mendapatkan rumus berat dari regresi linear.



```
Document - WordPad
1 2 3
@RELATION apel
@ATTRIBUTE tinggi NUMERIC
@ATTRIBUTE lebar NUMERIC
@ATTRIBUTE berat NUMERIC

@DATA
5.8,7,154
6.2,6.6,153
6,7.3,147
5.5,8,169
6.6,8,177
6,7.1,151
5.4,6,147
5.3,6.2,152
5.8,5.6,150
5.7,6.3,174
7,6.4,181
6.2,5.8,164
6.2,6.6,156
7,6.3,168
6.9,6.8,169
5.8,6.3,152
4.3,6.5,123
6.8,6.2,178
6.6,6.2,161
6.1,6.8,153
6.3,7.4,164
6.7,6.7,166
6.1,6.3,152
5.1,6,120
6.5,6.6,171
5.4,6.9,160
6.4,7,172
6.7,6.9,164
6.5,6.5,174
6.5,7.4,172
```

Gambar 3.2 Gambar pengumpulan dataset

Dataset diatas berjumlah 30 dengan baris pertama yaitu panjang, untuk baris kedua yaitu lebar, dan untuk baris terakhir adalah berat. Dataset tersebut didapat dari hasil pengukuran panjang dan lebar manual dengan penggaris, dan untuk berat menggunakan timbangan digital.

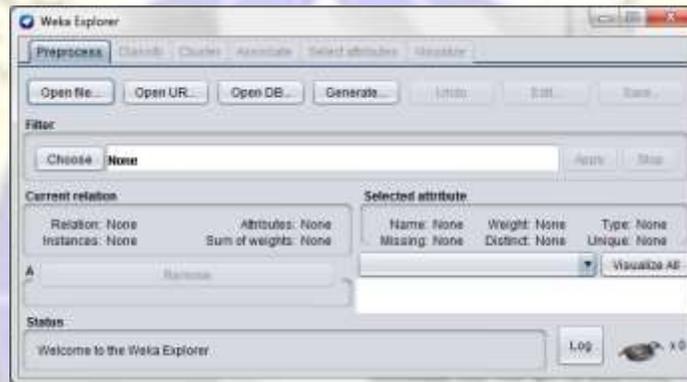
3.1.1.2. Regresi Linear

Regresi linear adalah algoritma jaringan saraf tiruan untuk dapat mengenali sebuah pola. Dimana pada proses ini data perhitungan manual akan dilatih terlebih

dahulu agar dapat dikenali. Kemudian dilakukan perhitungan untuk dapat menemukan berat buah apel fuji pada *library weka* yang sudah di *save* di *wordpad* dengan format *. arff. Berikut langkah menjalankan *weka* sampai mendapatkan rumus yang akan dimasukkan diprogram deteksi berat buah apel fuji.

a. *Preprocess*

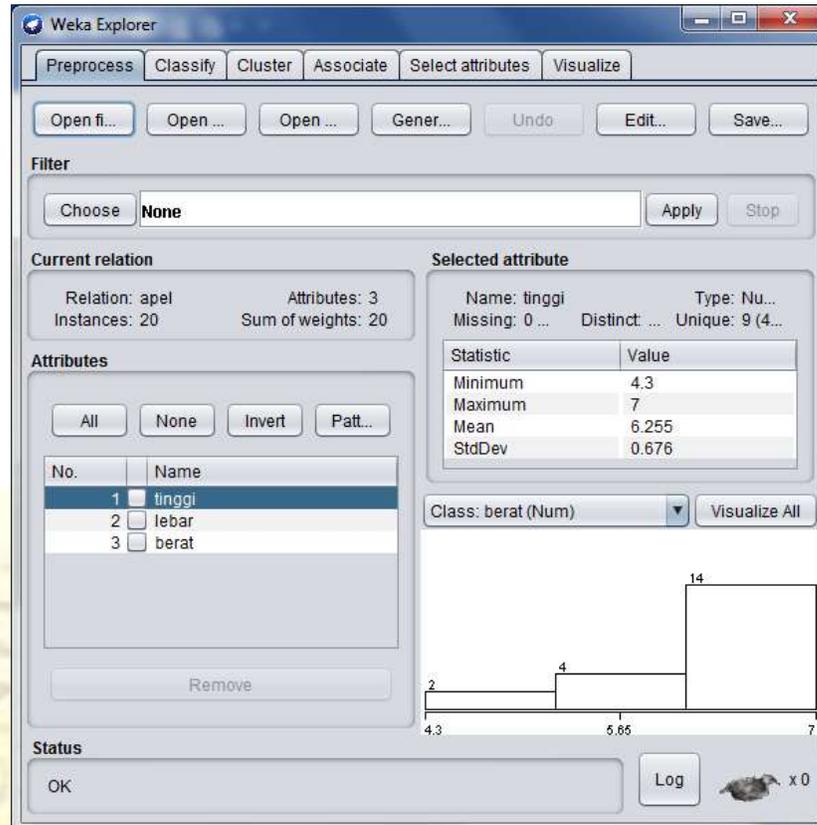
Setelah masuk pada aplikasi *weka*, kita akan langsung masuk pada *Preprocess* seperti gambar , *preprocess* pada *weka* adalah tahap awal agar kita bisa memasukkan *dataset* yang sudah di *save* dengan format *. arff. Pada halaman ini kita pilih *Open file*.



Gambar 3.3 Gambar Halaman Preprocess

b. *Open file*

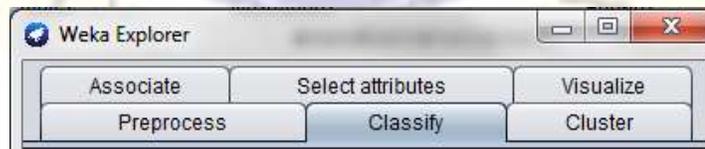
Setelah *Open file* kita akan masuk pada tempat dimana data dengan format *.arff yang akan dipilih, kemudian *open* untuk dilanjutkan keproses selanjutnya.



Gambar 3.1 Gambar Halaman Preprocess

d. *Classify*

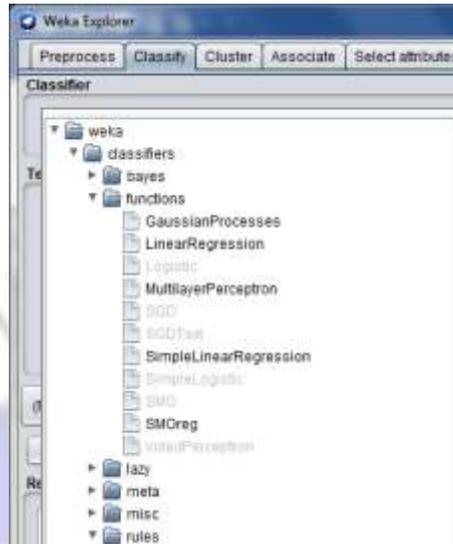
Selanjutnya melakukan klasifikasi, kita klik *classify*.



Gambar 3.2 Gambar Classify

e. Pilih Metode *LinearRegression*

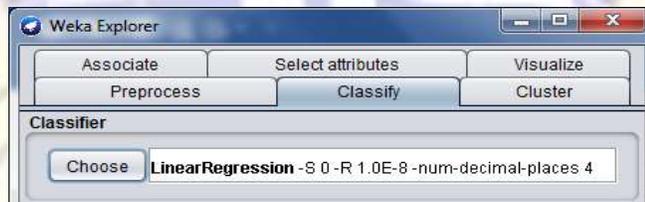
Selanjutnya pilih functions lalu *LinearRegression* agar dapat menghasilkan rumus berat berdasarkan tinggi dan lebar.



Gambar 3.3 Gambar Isi Menu Classify

f. *Choose*

Klik *Choose* kemudian pilih *functions* dan pilih *LinearRegression*



Gambar 3.4 Gambar Halaman Preprocess

g. Classifier Output

Kemudian klik *Start* lalu akan muncul rumus berat dibagian *Classifier output*

```
Classifier output

Linear Regression Model

berat =

17.1416 * tinggi +
4.9106 * lebar +
22.3196
```

Gambar 3.5 Gambar Halaman Preprocess

3.1.2. Rancangan Data Testing

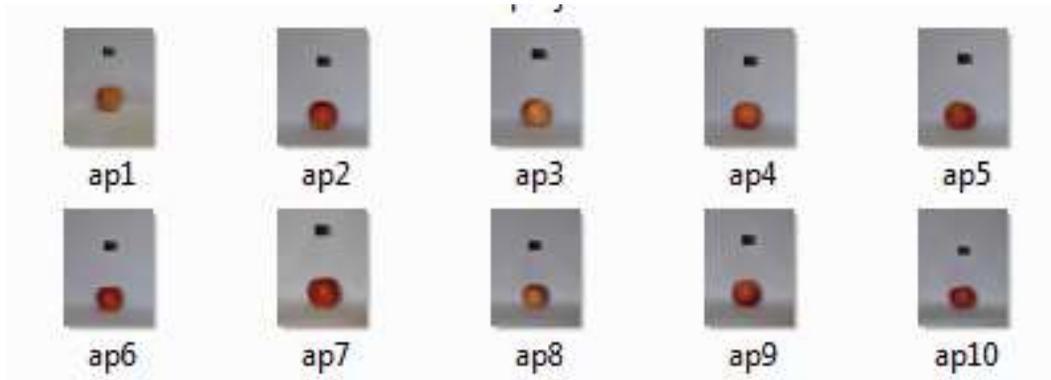
Data testing yaitu proses yang dilakukan setelah data trainig, dimana data testing sendiri adalah proses pengujian.



Gambar 3.6 Rancangan Data Testing

3.1.1.3. Pengumpulan Dataset

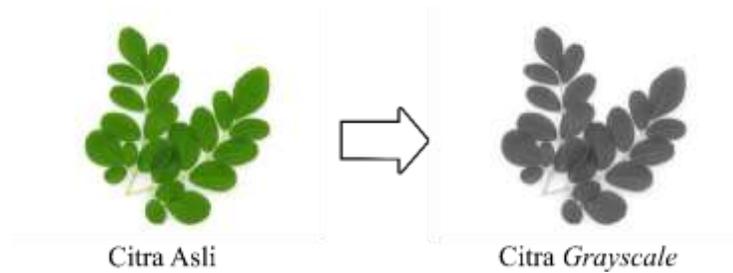
Data yang diambil adalah foto buah apel fuji yang diletakkan pada *background* yang sudah ditentukan sebanyak 10 gambar data testing. Alat yang digunakan untuk mengambil gambar adalah *handphone merk* samsung A50 dengan spesifikasi kamera belakang 25 MP. Proses pengambilan gambar dilakukan pada saat kondisi terang, supaya mendapatkan kecerahan gambar yang berbeda dan format gambar yang dihasilkan dari pengambilan foto adalah berformat *.jpg dengan resolusi gambar yang sama yaitu 3024×4032 pixel dengan tinggi stiker 15 cm.



Gambar 3.7 DataSet Buah Apel Fuji

3.1.1.4. Pre-Processing

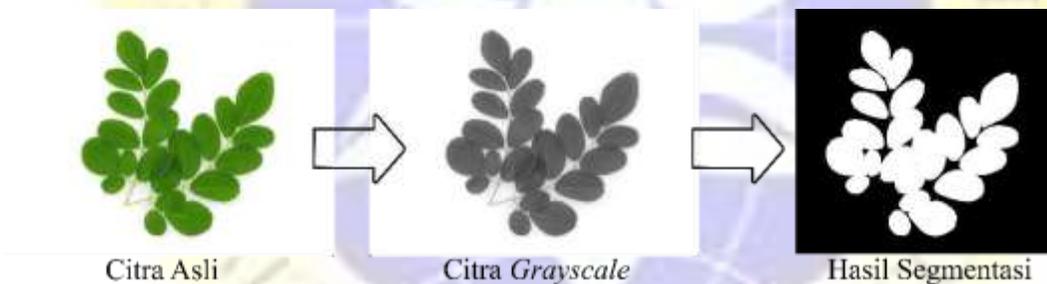
Pre-Processing dimulai dengan tahap awal penentuan *background* dan objek bantuan yang akan di tempel pada *background* terletak pada bagian atas objek utama atau buah apel fuji, kemudian dilanjutkan dengan pengambilan gambar menggunakan *handphone*. Pengambilan gambar ini bertujuan untuk memfokuskan hanya pada objek buah apel fuji dan *background* saja. Proses ini dilanjutkan dengan mentransformasi gambar hasil pengambilan (citra asli) menjadi gambar keabuan (citra *grayscale*). Konversi Citra *Grayscale* merupakan tahap awal *Image Processing* (pengolahan citra) yang merubah citra warna dari objek buah apel menjadi citra berskala keabuan (*grayscale*) bertujuan memudahkan saat melanjutkan ke proses *thresholding*. Pada proses ini, untuk mendapatkan *image grayscale* atau citra keabuan, cukup hanya memanggil fungsi yang telah disediakan pada *library OpenCV*. Berikut ini contoh tranformasi dari citra asli menjadi citra *grayscale* :



Gambar 3.8 Contoh Transformasi Citra Asli ke Citra Grayscale

3.1.1.5. Segmentasi Citra Menggunakan *Thresholding*

Segmentasi Citra Menggunakan *Thresholding* dilakukan pada tahap ini supaya memudahkan untuk memisahkan antara objek buah apel fuji dengan *background*, sehingga dapat membuat region citra pada objek buah apel fuji menjadi sangat gelap atau hitam sempurna dengan intensitas sebesar 0, dan region citra pada *background* menjadi putih sempurna dengan intensitas sebesar 1.



Gambar 3.9 Contoh Proses Segmentasi

3.1.1.6. Implementasi Metode Deteksi Kontur

Pada tahap ini dilakukan proses deteksi tepi pada objek menggunakan metode deteksi kontur. Dan juga digunakan untuk menghitung berapa banyak jumlah objek dalam sebuah frame menggunakan fungsi *findContours*. Pada penelitian ini metode deteksi kontur digunakan untuk membedakan objek yang dipilih dari objek bantuan berdasarkan minimum kontur-area yang ditentukan, sehingga penentuan

objek akan lebih akurat. Dengan menggunakan fungsi *drawContours*, maka hasil dari proses ini adalah berupa garis pada setiap sisi atau tepi objek yang terdeteksi. Pada Gambar 3.4 ini merupakan contoh proses deteksi kontur.



Gambar 3.10 Contoh Proses Deteksi Kontur

3.1.1.7. Kalkulasi Rasio Objek

Kalkulasi Rasio Objek merupakan tahapan perhitungan untuk menampilkan Rasio tinggi dan lebar pada objek buah apel fuji dan objek bantuan berupa stiker dengan tinggi 15 cm untuk kemudian melanjutkan kepada proses perhitungan jumlah piksel pada citra buah apel fuji dan citra stiker dengan cara menggunakan fungsi-fungsi *height* dan *width* untuk mendapatkan nilai piksel pada objek dan dengan perhitungan-perhitungan manual lainnya.

Selanjutnya tahap konversi ini merupakan tahapan penting untuk menghitung tinggi buah apel pada citra. Setelah jumlah masing-masing piksel objek telah di temukan, maka tinggi citra dan lebar citra dalam satuan piksel dapat di konversikan ke satuan centimeter (cm) untuk menghasilkan tinggi dan lebar objek yang sebenarnya. Adapun persamaan perhitungannya sebagai berikut:

Diketahui : t_s (px), l_{pa} (px), t_{pa} (px), t (cm)

$$\bullet 1 \text{ px} = a \text{ (cm) ?}$$

$$1 \text{ px} = \frac{t}{ts}$$

ts

$$1 \text{ px} = s$$

- ta (cm) ?

$$ta = tpa \times s = ta \text{ (cm)}$$

- la (cm) ?

$$la = lpa \times s = la \text{ (cm)}$$

Keterangan :

ts = Jumlah piksel citra stiker (px)

lpa = Lebar piksel buah apel fuji (px)

tpa = Tinggi piksel buah apel fuji (px)

tt = Tinggi objek stiker sebenarnya (cm)

la = Lebar buah apel fuji (cm)

ta = Tinggi buah apel fuji (cm)

s = Jumlah 1 piksel per cm (cm)

3.1.1.8. Implementasi Metode Regresi Linier

Pada tahap ini dilakukan deteksi pengenalan pola setelah dilatih terlebih dahulu baru setelah itu regresi linear dapat mengenali pola. Dimana data yang disimpan di *wordpad* dengan format *.arff kemudian dijalankan di *library weka* agar dapat menghasilkan rumus untuk berat. Setelah itu rumus tersebut diletakkan diprogram yang dibuat.

3.1.1.9. Membuat Aplikasi Deteksi Berat Apel Fuji

Pada tahap ini, akan dilakukan pengimplementasian dari metode yang telah di gunakan, di tahap ini proses penulisan kode dan membuat desain aplikasi sederhana

supaya mudah digunakan untuk menginput data gambar yang kemudian diproses dengan hasil output dari aplikasi adalah data berat apel fuji. Aplikasi ini dibuat dengan berbasis desktop menggunakan *tool Java Netbeans*.

3.1.1.10. Uji Coba

Uji coba merupakan tahap yang terpenting dalam penelitian, karena untuk menentukan apakah metode yang digunakan dalam sebuah aplikasi deteksi berat buah berdasarkan tinggi dan lebar berhasil atau gagal. Pengujian ini memfokuskan pada penerapan metode regresi linear untuk menemukan berat buah apel fuji berdasarkan tinggi dan lebar. Terdapat pengujian pada tahap uji coba kemudian di klasifikasikan hasil akurasi yang paling optimal. Untuk tahap ini rata-rata tingkat akurasi dari pengukuran berat buah apel fuji pada citra digital dapat dirumuskan sebagai berikut :

$$\text{Rata-rata akurasi} = \frac{\text{Jumlah Nilai Yang Sesuai}}{\text{Jumlah Total Gambar}}$$

Pada tahap ini, didapat tingkat akurasi dari penerapan metode regresi linier.

3.1.1.11. Penarikan Kesimpulan

Pada tahap ini, dapat ditarik kesimpulan tentang apa yang dibahas dalam penelitian menghitung berat menggunakan metode regresi linear, dan didapat tingkat akurasi berdasarkan hasil uji coba.