

## BAB II KAJIAN LITERATUR

### 2.1 Pengecekan kematangan buah melon melalui kulit

Pada penelitian sebelumnya, pengecekan kematangan buah melon dapat dilakukan dengan melihat ketebalan dan jaring jaring pada kulit buah melon. Hal ini dilakukan melalui pengolahan citra dan menggunakan metode *Support Vector Machine* (SVM). Tahapan yang dilakukan oleh penelitian Agung Prayoga untuk melakukan pengecekan kematangan buah melon melalui kulit adalah [5] :



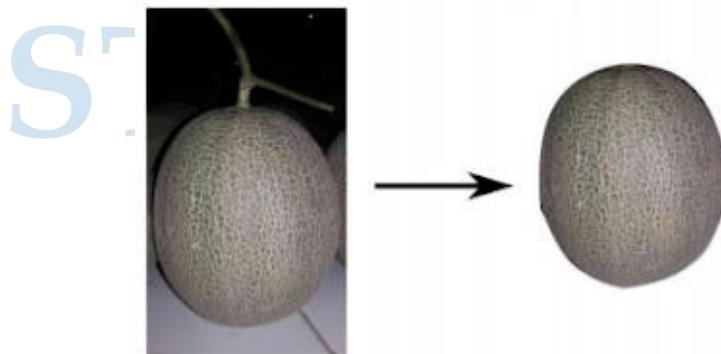
Gambar 2.1.1: Tahapan Pendeteksian Kematangan Buah Melon [5]

1. Menggunakan *dataset* untuk training dan testing

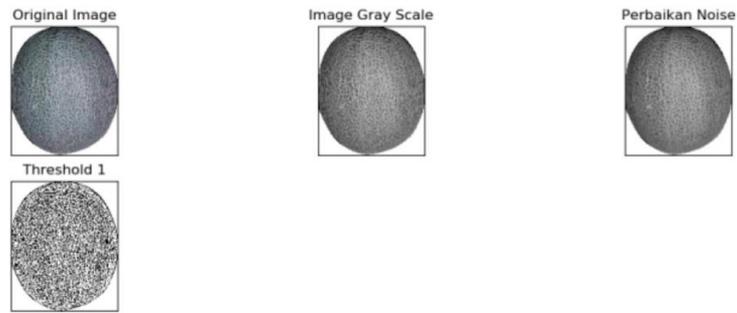
Dataset yang digunakan merupakan data yang sama pada penelitian sebelumnya. Data didapat dari beberapa perkebunan melon yang berasal dari Jawa Timur, Siak Pekanbaru, dan Solo. Data yang digunakan adalah foto buah melon yang diperoleh secara sampling dan berjumlah masing masing 150 buah per kategorinya. Terdapat 3 kategori buah melon yang digunakan sebagai data yaitu melon tidak matang berumur 45-50 hari setelah penanaman, melon setengah matang berumur 51-59 hari setelah penanaman, melon matang berumur 60-70 hari setelah penanaman.

2. *Preprocessing*

Preprocessing digunakan pada citra buah melon untuk memperbaiki kualitas citra tersebut. Proses yang dilakukan terbagi menjadi 2 yaitu *image cropping* dan *image enhancement*. *Image cropping* dilakukan untuk menghilangkan area di luar buah melon sehingga yang tampil hanya area buah melon nya saja sedangkan background di luar area buah melon dihilangkan. Sedangkan, *image enhancement* dilakukan untuk memperbaiki citra buah melon. Pertama, dilakukan konversi dari nilai *rgb* ke bentuk *grayscale*. Lalu, dilakukan *median blur* untuk memperbaiki *noise* pada buah melon. Terakhir, mengambil ciri pada citra buah melon dengan melakukan proses *adaptive threshold*. Dibawah ini merupakan contoh proses *image cropping* dan *image enhancement* :



Gambar 2.1.2: *Image Cropping* pada citra buah melon [5]



Gambar 2.1.3: Image Enhancement pada citra buah melon [5]

### 3. Ekstraksi Ciri Statistik

Setelah dilakukan *preprocessing*, dapat dilakukan ekstraksi dengan cara mengubah citra yang sudah diperbaiki kualitasnya menjadi histogram berbentuk *array*. Histogram digunakan untuk melakukan proses *training* dan *testing* dengan menggunakan *svm*.

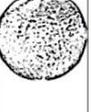
### 4. Training SVM

*Training svm* dilakukan untuk memperoleh model algoritma dengan klasifikasi buah melon matang, setengah matang, dan tidak matang dan dilakukan dengan cara pengambilan keputusan melalui metode SVM. Proses *training SVM* mempunyai tujuan untuk memperoleh informasi seperti *hyperplane*, *support vector*, dan lain nya yang direpresentasikan dalam bentuk *struct*. Pelatihan binary class atau 2 kelas berbeda akan menghasilkan *struct* dimana dalam proses nya akan didapatkan sebuah model yang nantinya akan digunakan untuk *testing*. Training dilakukan dengan melibatkan 100 citra buah melon per kategorinya yaitu melon matang, melon setengah matang, dan melon tidak matang.

### 5. Proses Testing

Model algoritma yang sudah didapat melalui proses *training* dapat digunakan untuk melakukan *testing*. *Testing* dilakukan dengan tujuan untuk meningkatkan akurasi dari pengklasifikasian tingkat kematangan buah melon melalui ekstraksi ciri statistik dan *support vector machine*.

Testing dilakukan dengan melibatkan 50 citra buah melon per kategorinya yaitu melon matang, melon setengah matang, dan melon tidak matang. Hasil dari testing menghasilkan klasifikasi yang mempunyai akurasi sebesar 76.00%. Dibawah ini adalah beberapa hasil klasifikasi dari tingkat kematangan buah melon :

Melon	Citra	Ekstraksi fitur	Klasifikasi	Melon	Citra	Ekstraksi fitur	Klasifikasi
Setengah Matang			Matang	Setengah Matang			Setengah Matang
Matang			Matang	Matang			Matang
Tidak Matang			Tidak Matang	Tidak Matang			Setengah Matang
Matang			Tidak Matang	Setengah Matang			Tidak Matang

Gambar 2.1.3: Hasil pengklasifikasian citra buah melon [5]

## 2.2 Python

*Python* adalah bahasa pemrograman yang mudah untuk dipelajari dan memiliki banyak keunggulan. Penggunaan bahasa pemrograman python memiliki banyak keunggulan seperti pembuatan aplikasi perangkat lunak, memiliki library yang beragam, penulisan kode yang relatif lebih mudah dipahami.

*Python* saat ini memiliki 2 versi besar yang berjalan yaitu *python 2* dan *python 3*. Perbedaan mendasar antara ke 2 versi ini adalah sintak yang banyak berubah pada *python 3* yang berbeda dengan *python 2*. Selain dari

segi *syntax*, *python* 3 juga tidak mendukung penulisan yang ada pada *python* 2 sehingga file yang berjalan pada *python* 2 tidak dapat dijalankan pada *python* 3. Selain itu juga, *python* 2 akan dihentikan perkembangannya di tahun 2020 [7]. Hal ini membuat *python* 2 tidak dapat digunakan lagi dalam jangka panjang karena sudah tidak memiliki dukungan secara resmi.

### 2.3 *Javascript*

*Javascript* adalah bahasa pemrograman yang dapat berjalan secara multiplatform. Saat ini *javascript* bisa berjalan sebagai *front end* maupun *backend*. *Javascript* juga dapat digunakan baik untuk membangun aplikasi berbasis *website*, *mobile*, atau *desktop*. Popularitas *javascript* juga sangat populer beberapa tahun ini. Hal ini bisa dilihat pada survey yang dilakukan *stackoverflow* bahwa *javascript* menempati urutan 5 besar bahasa pemrograman yang terpopuler [8]. Popularitasnya itu memiliki banyak dukungan dan pengembangan dari berbagai *developer*. Hal itu dibuktikan dengan banyaknya *library* atau *framework* pada *javascript* seperti *react*, *vue*, *angular*, dan lainnya.

### 2.4 *Mobile Development*

Aplikasi *mobile* menjadi populer beberapa tahun ini. Total unduhan aplikasi yang dilakukan pada perangkat *android* mencapai lebih dari 1.5 juta perbulan [9]. Hal ini menandakan banyak pengguna dan aplikasi yang hadir pada sektor *mobile*. Dengan banyaknya pengguna aplikasi *mobile*, pengembangan aplikasi *mobile* menjadi sesuatu yang penting akhir-akhir ini.

Pengembangan aplikasi *mobile* dapat dilakukan dengan berbagai cara. Ada yang langsung menggunakan *native* dan ada juga yang menggunakan *hybrid* dalam pembuatan aplikasi *mobile* tersebut. Pembuatan aplikasi secara *native* dibuat secara spesifik terhadap platform sistem operasi pada perangkat *mobile* [10]. Contohnya adalah pembuatan aplikasi *mobile* menggunakan *java* atau *kotlin* yang langsung tertuju hanya pada sistem operasi *android*.

Sedangkan untuk pembuatan aplikasi secara *hybrid*, pembuatan aplikasi dapat langsung dikerjakan untuk beberapa platform yang berbeda sistem operasi. Contohnya adalah *react native* yang dapat digunakan untuk membuat aplikasi untuk sistem operasi *android*, *ios*, maupun perangkat *desktop*.

#### **2.4.1 React Native**

*React Native* adalah *framework* berbasis *javascript* yang digunakan untuk membangun aplikasi *android*, *ios*, maupun *windows* [11]. Dengan menggunakan *react native*, pembangunan aplikasi seperti *android* dan *ios* cukup menggunakan bahasa pemrograman *javascript* sehingga pembangunan aplikasi relatif lebih mudah.

*React native* memiliki beberapa fitur yang dapat membantu pembangunan aplikasi. Pertama, dapat membuat berbagai komponen yang dapat dikelola dan disusun menjadi suatu tampilan. Kedua, setiap perubahan akan langsung memperbarui aplikasi secara otomatis sehingga tidak perlu dilakukan penyusunan aplikasi dari awal. Ketiga, sifat *react native* yang dapat digunakan pada beberapa platform membuat penggunaan kode menjadi lebih ringkas karena kode dapat digunakan pada banyak platform sekaligus.

### **2.5 Web Service**

*Web service* adalah sebuah sistem pada perangkat lunak yang digunakan sebagai pemberi layanan atau data yang dapat diakses melalui berbagai bahasa pemrograman berbeda [12]. Dengan menggunakan *web service*, aplikasi dapat saling berbagi data dan service meskipun berbeda platform.

*Web service* dapat dibuat menggunakan berbagai bahasa pemrograman. Salah satunya adalah dengan bahasa pemrograman *python*. Pada bahasa pemrograman *python*, *web service* dapat dibuat menggunakan framework seperti *django* dan *flask*. Setelah *web service* dibuat, data dapat dikirimkan antar platform yang saling terkoneksi melalui *web service*. Tipe data yang

digunakan dalam *web service* ada berbagai macam, seperti *JSON*, *XML*, dan lain nya.

### 2.5.1 *Flask*

*Flask* adalah framework berbasis *python* yang dapat digunakan untuk membangun sebuah web. Bersifat *micro framework*, yaitu dapat digunakan untuk membangun aplikasi sesederhana mungkin. Fungsi umum yang terdapat pada *framework* lain seperti *database*, validasi, dan lain sebagainya secara umum tidak ada pada *flask*. Tetapi, pengguna dapat menambahkan komponen yang disediakan oleh pihak ke 3 secara langsung kedalam *flask* [13]. Sehingga pembangunan web dapat berjalan seefisien mungkin sesuai kebutuhan pembuatan aplikasi.

### 2.5.2 *Javascript Object Notation (JSON)*

*JSON* adalah format teks atau syntax yang digunakan untuk melakukan serialisasi *data*. *JSON* bisa berfungsi sebagai format file yang digunakan untuk menyimpan ataupun mengirim data. Penggunaan *JSON* mirip dengan *xml* dimana keduanya dapat melakukan komunikasi yang berjalan antar platform. Tetapi, json sendiri mempunyai keunggulan yang lebih dibanding xml baik dari segi penulisan yang relatif mudah, kecepatan, dan relatif lebih ringan [14]. *JSON* biasanya lebih sering digunakan sebagai format *file* yang dikirimkan dari ke suatu *server* ke aplikasi lalu data yang ada pada json diolah agar dapat ditampilkan pada aplikasi.

## 2.6 *Sqlite*

*Sqlite* adalah *database* berbasis *sql* yang fleksibel dan mudah digunakan. Dengan menggunakan *sqlite*, *database* dapat dibuat dalam sebuah file tersendiri dan dapat dipindahkan kedalam berbagai sistem dengan mudah.

*Sqlite* membuat *database* yang dijalankan tidak terlalu bergantung pada *server*. *Sqlite* dapat langsung memproses file *database* yang digunakan tanpa

perlu melakukan konfigurasi ataupun instalasi sebelum database digunakan [15]. Karena hal tersebut, *sqlite* cocok digunakan untuk aplikasi yang bersifat sederhana dan tidak membutuhkan database yang kompleks.

## 2.7 Expo

*Expo* adalah framework *javascript* yang digunakan untuk membantu pembangunan aplikasi berbasis *react native*. Pembangunan aplikasi dapat lebih cepat dikarenakan banyaknya *service* dan *tools* yang tersedia pada *expo*. Fitur *native* pada platform seperti kamera, galeri, dan lain sebagainya sudah disediakan oleh *expo* sehingga dapat langsung digunakan dalam pembangunan aplikasi [16].

## 2.8 Rational Unified Process (RUP)

*Rational Unified Process (RUP)* adalah metode yang digunakan untuk pengembangan perangkat lunak yang berbasis *UML* yang menekankan pada *use case*. Pengembangan dilakukan secara iteratif dan juga terfokus pada arsitektur pengembangan perangkat lunak.

Fase yang dilakukan pada RUP terdiri dari 4 buah tahap yaitu [17] :

1. *Inception*, pada tahap ini dilakukan pengumpulan informasi baik melalui wawancara atau observasi. Hasil dari pengumpulan informasi tersebut digunakan untuk menentukan ruang lingkup pengembangan sistem.
2. *Elaboration*, pada tahap ini dilakukan analisis dan perancangan pada sistem. Analisis dilakukan untuk mengidentifikasi masalah yang harus diselesaikan. Hasil dari analisis akan digunakan untuk perancangan sistem yang meliputi diagram uml berbentuk use case dan tampilan aplikasi.
3. *Construction*, pada tahap ini dilakukan implementasi terhadap aplikasi yang telah dibuat. Implementasi yang dilakukan berupa cara untuk membangun aplikasi.

4. *Transition*, pada tahap ini dilakukan pengujian aplikasi dengan cara melakukan *testing* pada aplikasi untuk menentukan aplikasi berjalan dengan baik atau tidak.

## 2.9 UML Diagram

*Unified Modeling Language* (UML) adalah bahasa yang digunakan untuk memvisualisasikan pengembangan aplikasi berbasis objek. Pada tahun 1997, UML menjadi standar bahasa yang digunakan untuk pemodelan oleh Object Management Group (OMG). Penggunaan UML sebagai bahasa untuk melakukan visualisasi bergantung pada cara bagaimana menyusun diagram [18].

Terdapat beberapa diagram yang dapat digunakan ketika mengerjakan UML, beberapa diantaranya adalah [19] :

1. *Use case* melakukan pemodelan dengan cara mendeskripsikan bagaimana user bisa melakukan interaksi dengan sistem agar sistem tersebut bisa dipakai.
2. *Class Diagram* melakukan pemodelan melalui interaksi antar class yang nantinya akan saling berkaitan dan membentuk suatu sistem.
3. *Activity diagram* digunakan untuk menggambarkan bagaimana aktivitas yang bekerja dalam suatu sistem.
4. *Sequence diagram* dapat bekerja dengan cara mengilustrasikan bagaimana objek saling bekerja untuk saling mengirim dan menerima pesan.

## 2.10 Black Box Testing

*Black box testing* adalah suatu cara untuk pengujian perangkat lunak. Pengujian dilakukan tanpa perlu mengetahui bagaimana produk itu dibuat. Seperti misalnya perangkat lunak yang dibuat, tidak perlu diketahui bahasa pemrograman yang digunakan untuk membuat perangkat lunak. Pengujian

hanya terfokus terhadap sesuatu yang bisa dilakukan pada perangkat lunak [20]. Salah satu teknik yang digunakan untuk black box testing adalah *equivalence partitioning*. Metode ini bertujuan untuk mengelompokkan berbagai inputan sesuai fungsinya. Tahapan yang dilakukan pada metode ini dengan melakukan *equivalence partitioning* pada perangkat lunak yang akan diuji untuk menentukan *test case*, lalu masukan dan keluaran melakukan inisialisasi dengan *standar grade partition* [21].

### **2.11 Usability Testing**

Usability testing adalah teknik pengujian antarmuka pada perangkat lunak untuk mengetahui fitur pada perangkat lunak yang dibuat untuk menguji seberapa baik aplikasi dapat berjalan [22]. Pengujian yang dilakukan akan menghasilkan data kepuasan pengguna dalam menggunakan antarmuka pada perangkat lunak dan data tersebut akan digunakan untuk melakukan perbaikan atau penambahan yang perlu dilakukan untuk meningkatkan kualitas perangkat lunak.

### **2.12 Skala Likert**

Skala likert adalah skala yang digunakan dalam pengukuran suatu hal yang bersifat positif atau negatif. Penggunaan skala *likert* banyak digunakan ketika ingin mengukur kepuasan dalam suatu sistem. Skala *likert* memiliki 4 skor untuk mengukur suatu hal yang positif dan 4 skor untuk mengukur suatu hal yang negatif. Pemberian skor untuk hal positif adalah 4,3,2,1 dan untuk hal negatif adalah 1,2,3,4 [23].

### **2.13 Postman**

*Postman* adalah alat yang digunakan untuk melakukan pengujian *api*. Pengujian dilakukan dengan memasukan alamat *url* pada *postman* dan data yang diperlukan, lalu *postman* akan menuju alamat tersebut beserta data yang dibutuhkan. Penggunaan *postman* dapat membantu pengguna untuk

melakukan pengujian *api* khusus nya pengujian *api* dalam arsitektur *restful web service* [24]

## 2.14 Penelitian Terkait

Tabel 2.14 1 Penelitian Terkait

Nama Peneliti	Judul	Metodologi	Persamaan/Perbedaan
Lazuardi Arsy, Okky Dwi Nurhayati, Kurniawan Teguh Martono  (2016, Universitas Diponegoro)	Aplikasi Pengolahan Citra Digital Meat Detection Dengan Metode Segmentasi K-Mean Clustering Berbasis OpenCV Dan Eclipse	Metode yang digunakan pada penelitian ini adalah segmentasi <i>k-mean clustering</i>	Persamaan : Membuat aplikasi android untuk pendeteksiian berbasis citra  Perbedaan : Tidak menggunakan web service untuk pendeteksiian gambar, melainkan menggunakan open cv langsung di aplikasi android.  Pembuatan aplikasi android berbasis java
Joshua Caine Buwono, Liliana , Henry Novianus Palit  (2017, Universitas Kristen Petra)	Aplikasi Pendeteksi Citra yang diambil dari Kamera Mobile Device untuk Menghasilkan Ensiklopedia Fauna Berbasis Android	Metode yang digunakan pada penelitian ini adalah <i>image recognition</i> dengan pengambilan citra melalui <i>mobile</i> dan <i>web service</i>	Persamaan : Menggunakan web service sebagai bagian untuk melakukan <i>image processing</i>  Perbedaan : Menggunakan java sebagai basis pembuatan aplikasi android
Agung Prayoga  (2018, STT Terpadu Nurul Fikri)	Pengembangan Metode Deteksi Tingkat Kematangan Buah Melon Berdasarkan Tekstur Kulit	Metode yang dilakukan pada penelitian adalah metode eksperimen	Persamaan : Menggunakan kulit melon sebagai bahan penelitian untuk pengecekan kematangan buah melon.

	Buah		Perbedaan : Program yang dijalankan untuk melakukan pendeteksian hanya dijalankan melalui command prompt.
Ihsanul Fikri Abiyyu (2020, STT Terpadu Nurul Fikri)	Perancangan Aplikasi Android Pendeteksi Kematangan Buah Melon	Penelitian ini menggunakan metode kualitatif	

Dari hasil penelitian terkait diatas, persamaan terbesar yang dapat diambil adalah perancangan aplikasi berbasis android untuk pendeteksian gambar menggunakan *image processing*. Untuk perbedaan terbesar nya, penelitian ini menggunakan *react native* untuk pembuatan aplikasi *android* sedangkan penelitian lain menggunakan *java* sebagai basis pembuatan *aplikasi android*.

STT - NF