

BAB II TINJAUAN PUSTAKA

I. Penelitian Sebelumnya

Penelitian sebelumnya tentang *Image Processing* pada *smartphone* Android sudah banyak dilakukan dengan berbagai macam tujuan. Penerapan *Image Processing* sangat beragam dan pada penulisan penelitian ada beberapa penelitian yang berkaitan *Image Processing* yang dijadikan referensi. Berikut beberapa penelitian terkait mengenai *Image Processing* :

Penelitian dilakukan oleh Yuqing Chen dkk. [10] mengenai Metode Pengolahan Citra Hama Berbasis Lensa Inframerah Termal Android, Pada penelitian ini menggunakan obyek penelitian penyakit tumor jagung tepung jamur, lensa pencitraan termal inframerah berbasis sistem android digunakan untuk mendapatkan citra penyakit jagung. Hama dan penyakit pertanian sangat mempengaruhi hasil dan kualitas tanaman. Untuk mempelajari kejadian hama dan penyakit, serta meningkatkan laju pemanfaatan pestisida, dirancang dan diimplementasikan sistem pemrosesan cepat Android untuk citra inframerah hama dan penyakit serangga berbasis Android. Sedangkan teknologi OpenCV digunakan untuk pengolahan citra untuk mendapatkan efek segmentasi inframerah termal yang lebih baik. Efek segmentasi citra berdasarkan algoritma Otsu tradisional (Otsu) dan metode segmentasi Otsu *single-threshold* berdasarkan algoritma *bare bones fireworks* (BBFWA) dianalisis. Yang penting, hasil penelitian menunjukkan bahwa metode Otsu *single-threshold segmentation* berdasarkan BBFWA memiliki pengaruh terbaik terhadap segmentasi citra *thermal infrared*. Pada penelitian ini diperoleh citra infra merah termal penyakit jagung dengan bantuan lensa pencitraan termal inframerah dan ponsel sistem Android. Sistem ini menerapkan platform Android untuk mengembangkan sistem pemrosesan citra inframerah termal. Sistem ini memiliki tiga fungsi utama: 1) Pengambilan Gambar: memilih gambar inframerah termal yang diambil oleh *FlirOne*

langsung dari album; 2) Pemrosesan gambar: algoritma Otsu yang digabungkan dengan algoritma BBFWA terbaru untuk membentuk algoritma BBFWA-Otsu untuk segmentasi ambang gambar grayscale, dan piksel putih statistik dan piksel hitam untuk menghitung tingkat infeksi penyakit; 3) Menanyakan: menurut tanggal perolehan citra inframerah termal dan waktu perolehan untuk menanyakan informasi historis. Penelitian ini secara efektif dapat mengidentifikasi area lesi yang terinfeksi, memiliki biaya rendah, kinerja waktu nyata yang baik dan portabilitas, dapat memberikan dukungan data untuk jumlah penyemprotan kendaraan udara tak berawak, memiliki signifikansi panduan tertentu untuk penelitian penyakit dan hama serangga, dan juga memberikan dukungan teknis untuk pertanian cerdas.

Penelitian yang dilakukan oleh Sundara dan Sundareshwaran [11], mengenai Aplikasi Pemrosesan Gambar berbasis Android untuk Mendeteksi Penyakit Tanaman, Penyakit pada tanaman memiliki kecenderungan mengurangi umur tanaman. Pengendalian penyakit tanaman selalu menjadi praktek penting untuk kelangsungan hidup tanaman. Deteksi dini dan diagnosis yang tepat diperlukan untuk pengelolaan penyakit tanaman yang memadai. Deteksi awal secara luas didasarkan pada tanda dan gejala penyakit, tandanya jika tanaman terkena penyakit adalah tampak adanya fisik daerah terinfeksi seperti bercak putih, coklat atau hitam, terdapat massa spora pada bagian tanaman. Meskipun bisa dikenali dengan cara profesional dalam mengenali penyakit, namun dengan adanya aplikasi ponsel Pintar Modern dapat digunakan untuk diagnosis pada tahap awal secara mudah. Penelitian ini mengenai Aplikasi Pemrosesan Citra berbasis Android untuk mendeteksi penyakit tanaman menggunakan *platform android embedded vision* pada Android Studio. Aplikasi yang dibuat melakukan pendeteksian titik terinfeksi menggunakan teknik transformasi warna, deteksi tepi dengan mengidentifikasi titik pada bagian tumbuhan. Aplikasi ini dapat diperpanjang untuk berbagai penyakit tanaman dan platform ponsel pintar yang berbeda. Tujuan utama aplikasi

ini untuk mendeteksi terjadinya penyakit pada tanaman serta menampilkan persentase wilayah yang terjangkit penyakit tersebut. Hasil dari aplikasi ini menunjukkan akurasi sekitar 90% untuk mendeteksi penyakit dengan benar. Metode yang digunakan dapat diperluas dengan cara mengembangkan aplikasi berbasis android sehingga dapat mendeteksi dan mengklasifikasikan berbagai penyakit tanaman di semua bagian tanaman seperti batang, buah, akar dan bunga serta dapat memberikan beberapa tindakan pencegahan yang harus dilakukan untuk setiap penyakit.

Arif dkk. [7] melakukan penelitian yang berjudul “Pengembangan Aplikasi Android Untuk Mengidentifikasi Kualitas Beras dengan Citra Digital”. Beras merupakan makanan utama bagi sebagian besar penduduk Indonesia. Penelitian ini dilakukan untuk mengklasifikasikan beras utuh dan pecah menggunakan citra morfologi. Pengujian visual merupakan salah satu cara yang digunakan Bulog untuk menguji kualitas beras dan digunakan secara langsung melalui citra beras yang ditangkap kamera pintar konsumen. Pengujian ini diukur berdasarkan nilai putih, bersih dan utuh dari citra beras yang dipisahkan dari background. Skor keputihan dan bersih beras diperoleh dengan menganalisis nilai HSV dari citra beras, sedangkan skor penuh diperoleh dengan menganalisis luas objek beras. Kemudian nilai putih, bersih dan utuh dari citra beras tersebut diklasifikasikan menjadi 3 kelas yaitu baik, buruk dan buruk, dengan menggunakan pohon keputusan C4.5. Hasil penelitian menunjukkan bahwa morfologi citra cukup efektif untuk mengklasifikasikan beras utuh dan pecah. Sebelumnya, teks diverifikasi menggunakan metode OCR (optical character recognition) berbasis Android. Hasil penelitian menunjukkan bahwa sistem pengenalan tanda atau teks dapat dibangun untuk Android dengan menggunakan kamera sebagai pemindai atau alat input. Untuk mengklasifikasikan kualitas beras secara visual, HSV mengujinya dengan nilai parameter putih Hue 36° Saturasi $\leq 20^\circ$ sama dengan 97%, sedangkan nilai bersih Hue adalah $72^\circ \leq H \leq 288^\circ$ Saturasi $\leq 40\%$. Nilai $\geq 60\%$. Dari segi integritas atau morfologi, hal ini tercermin dari nilai teritorial dari setiap item. Gambar yang diambil oleh kamera smartphone kemudian diklasifikasikan menurut aturan C4.5. Kualitas beras dari citra digital ditransfer ke ponsel Android menggunakan metode waterfall dan diuji coba dengan black box test.

Penelitian yang dilakukan oleh Cucu dan Ikhwan[12] tentang pengenalan wajah manusia dalam sistem pelacakan kehadiran kuliah menggunakan fungsi analisis komponen utama (PCA), penelitian ini mengembangkan

sistem pelacakan peserta yang menggunakan pengenalan wajah manusia dan terintegrasi dengan database dengan menjalankan Python - Gunakan pemrograman. bahasa dan pustaka opencv. Karena sistem kontrol kehadiran yang ada memiliki kelebihan dan kekurangan dan harus terus diperbaiki untuk mempermudah pengolahan data. Pengumpulan data gambar dilakukan pada ponsel Android, kemudian gambar dikenali dan di-crop untuk menjaga wajah saja. Proses pendeteksian wajah: Dari sepuluh wajah dalam sampel, terdapat satu wajah yang tidak dapat dideteksi, meskipun terdeteksi dengan benar. Pencahayaan merupakan salah satu faktor yang mempengaruhi proses pengenalan wajah, namun perbedaan dalam menangkap wajah dan ekspresi masih dapat ditoleransi dengan baik dalam proses pengenalan. Pengenalan wajah menggunakan metode Haar Cascade Classifier, kemudian metode ekstraksi fitur PCA (Principal Component Analysis) dan database MySQL. Hasil tes PCA diberi label sesuai dengan data yang ada di database kemudian diuji dengan metode Euclidean distance. Hasil dari proses pengujian adalah waktu pengenalan 0,63 detik, sedangkan waktu rata-rata proses pengenalan wajah adalah 0,89 detik, hal ini dikarenakan kecepatan proses perhitungan menggunakan bahasa pemrograman Python dan library OpenCV. Pengenalan wajah dalam proses pelatihan Tingkat keberhasilan aplikasi penelitian ini adalah 100% dan hasil pengenalan wajah dalam proses pengujian adalah 90%.

Penelitian berjudul “Aplikasi Android Koreksi Lembar Jawaban Komputer Menggunakan Opencv” yang dilakukan oleh Yusril dkk. [13] berkaitan dengan inovasi untuk mempercepat peninjauan hasil tes yang ada melalui penggunaan teknologi OMR (Optical Mark Reader), yang melakukan koreksi dengan cepat hanya dengan menggunakan komputer dan scanner. Tujuan dari penelitian ini adalah untuk membuat sebuah aplikasi yang akan menggantikan fungsi scanner dan komputer dalam mengoreksi jawaban menggunakan teknik image processing yang dibangun di Open CV sebagai sistem pengolah utama hasil LJK. Menggunakan Android Studio untuk mengembangkan aplikasi Android. Menggunakan kamera *smartphone*

sebagai alat untuk memindai LJK, kamera akan menangkap gambar LJK dan OpenCV akan mengenali setiap item jawaban yang dipilih dalam pilihan jawaban. Dan kemudian total hasil jawaban yang benar akan muncul di layar aplikasi Android. Pengolahan citra sistem aplikasi ini membaca kemungkinan jawaban dengan menghitung jumlah piksel hitam. Hasil pengujian sistem deteksi respon hitam menggunakan *pen type 2B* menunjukkan akurasi deteksi berhasil 100%, sedangkan akurasi deteksi yang sesuai menggunakan tingkat keberhasilan terendah yaitu H. dari jenis pena 3H, cukup berhasil. Suku bunga 4%. Sistem aplikasi dapat bekerja maksimal dengan intensitas cahaya 50 lux dan jarak optimal dari objek dan kamera 17 cm. Pencahayaan yang baik pada saat pemindaian sangat mempengaruhi sistem aplikasi untuk mengenali setiap pilihan jawaban pada lembar jawaban.

Tabel 2.1 Rangkuman Tinjauan Pustaka

No.	Judul Penelitian	Tujuan Penelitian	Metode Penelitian	Hasil Penelitian	Keterangan
1.	Implementation of Morphology Image Processing and BLOB Analysis in Cow's Milk Somatic Cell Content Calculation [23] (2021)	Dapat mengetahui cara menghitung sel somatik menggunakan komputer tanpa penghitungan manual atau menggunakan alat <i>Fossomatic</i> .	Untuk mengidentifikasi somatik sel dengan Operasi Morfologi dan BLOB Analysis.	Aplikasi komputer untuk penghitungan jumlah sel somatik dengan akurasi mencapai 85%	Masih menggunakan perangkat komputer, penelitian ini masih dapat dikembangkan menjadi aplikasi mobile.
2.	An Android based Image Processing Application to Detect Plant Disease[11] (2019)	Membuat aplikasi Pemrosesan Gambar berbasis Android untuk Mendeteksi Penyakit Tanaman.	<i>Neural Networks</i> (ANN)	Aplikasi untuk pengecekan penyakit pada tanaman dan hasil pengujian yang sudah dilakukan dengan mengolah lebih dari 50 citra. Aplikasi mendeteksi hasil yang benar untuk 45 dari 50 gambar. Oleh karena itu keakuratan aplikasi ini untuk dihitung menjadi 90%.	Masih menggunakan Android versi 3.0 dan belum mendeteksi dan mengklasifikasikan berbagai penyakit tanaman di semua bagian tanaman seperti batang, buah, akar dan bunga serta dapat memberikan beberapa tindakan pencegahan yang harus dilakukan untuk setiap penyakit.

No	Judul Penelitian	Tujuan Penelitian	Metode Penelitian	Hasil Penelitian	Keterangan
3.	Segmentasi Mata Katarak Pada Citra Medis Menggunakan Metode Operasi Morfologi [9] (2018)	Untuk mendeteksi penyakit mata katarak yang dapat dikembangkan ke arah robotic untuk membantu tenaga medis..	Segmentasi citra menggunakan operasi morfologi.	Berdasarkan hasil analisis pakar 1 sebesar 86,4%, pakar 2 88,8%, pakar 3 89,2%, sedangkan pakar 4 88,8%. Dari hasil persepsi pakar tersebut dapat disimpulkan rata-rata persepsi keempat pakar tersebut sebesar 88,3%.	Pada segmentasi citra ini, dapat dikembangkan menjadi berbasis android maupun robotic.
4.	Metode Color Blob Detection Untuk Deteksi Kematangan Tomat Secara Otomatis Berbasis Android[14] (2017)	Untuk mendeteksi kematangan tomat yang terklasifikasi secara otomatis menggunakan blob detection	mendeteksi kematangan tomat menggunakan metode blob analysis	Hasil percobaan dengan sampel tomat matang, dengan 14 data terpilih dari 15 kali pengambilan data, mencapai nilai akurasi 93%	User harus melakukan <i>onTouch</i> pada layar smartphone untuk membentuk kontur pada area yang dikenai proses <i>onTouch</i> .
5.	Aplikasi Android Koreksi Lembar Jawaban Komputer Menggunakan Opencv[13] 2020	Untuk mempercepat proses pemeriksaan hasil ujian	OMR (Optical Mark Reader)	Hasil pengujian sistem deteksi respon hitam menggunakan pen type 2B menunjukkan akurasi deteksi berhasil 100%, sedangkan akurasi deteksi yang sesuai menggunakan tingkat keberhasilan terendah yaitu H. dari jenis pena 3H, cukup berhasil. Rata-rata 4%.	Minimal sistem operasi Android versi 5.0 (Lollipop) baru bisa menjalankan aplikasi dan Jarak efektif untuk proses scan hanya sampai 22 cm serta sudut optimalnya sebesar 120°.

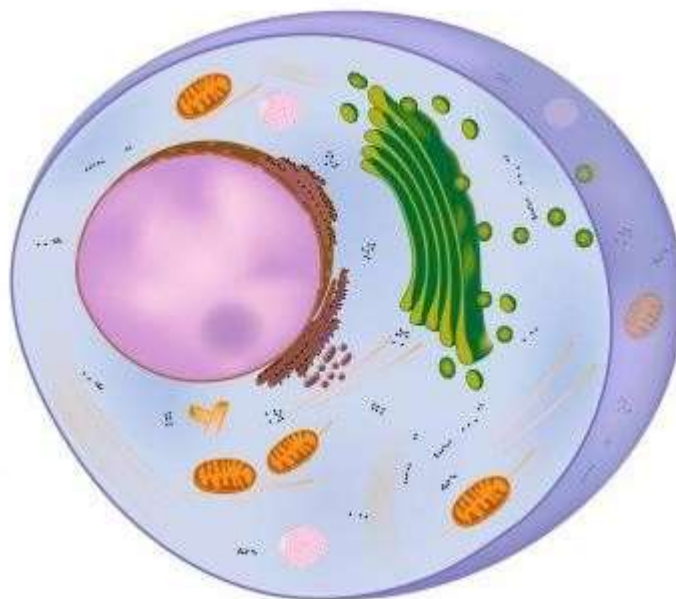
Dari tabel di atas dapat disimpulkan bahwa penelitian ini berkaitan dengan jurnal penelitian dengan judul

Implementation of morphology image processing and BLOB Analysis in cow's milk somatic cell content calculation [23] dengan Citra Digital dan Metode Color Blob Detection Untuk Deteksi Kematangan Tomat Secara Otomatis Berbasis Android [14] sebagai referensi utama.

II. Dasar Teori

A. Somatik Sel

Sel Somatik dalam susu sapi merupakan sel darah putih atau leukosit yang terdapat pada susu yang baru di perah karena leukosit dapat menembus dinding kapiler dalam ambing. Sel kelenjar dan konstituennya juga bisa ditemukan pada susu. Leukosit dalam susu tidak hidup dan jumlahnya tidak meningkat setelah susu diperah. Jumlah sel pada susu normalnya bervariasi antara 50.000 - 100.000/ml. Susu dari ambing yang sakit, misalnya mastitis menunjukkan adanya peningkatan jumlah pada sel somatik. Bila jumlahnya mencapai 150.000, kondisi sapi harus dicek dan bila sel somatik mencapai 500.000 atau lebih, ambing sapi menjadi tidak normal [15].



Gambar 2.1 Ilustrasi Somatik sel [16]

Jumlah sel somatik susu sangat tinggi, 3-14 hari x 10.000 sel somatik susu langsung dilakukan dengan berbagai metode dan alat, seperti: Metode langsung biasanya digunakan dalam studi eksperimental dan observasional, sedangkan metode tidak

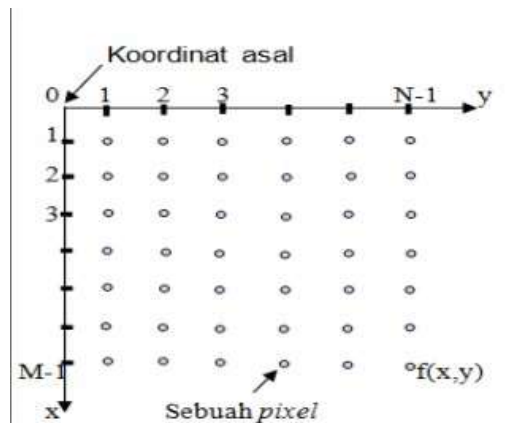
langsung biasanya digunakan dalam survei dan program pengendalian mastitis. Perhitungan jumlah sel somatik menurut metode breed. Variabel yang diamati adalah hasil uji mastitis dan jumlah sel somatik pada susu sapi. Data disajikan dalam tabel dan histogram dan kemudian dijelaskan dengan jelas. Metode ini diyakini memiliki keunggulan antara lain kemudahan penggunaan, kecepatan, sensitivitas dan spesifisitas yang tinggi.[17].

B. Citra

Secara harfiah, citra adalah gambar dalam bidang dua dimensi. Secara matematis, citra adalah fungsi kontinu dari intensitas cahaya dalam bidang dua dimensi. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian berkas cahaya. Pantulan cahaya ini ditangkap oleh alat optik seperti mata manusia, kamera, scanner, dan lain-lain, sehingga bayangan objek dapat ditangkap dalam bentuk gambar[18]. Secara umum pengolahan citra digital mengacu pada pengolahan citra 2 dimensi pada komputer. Dalam arti yang lebih luas, pemrosesan citra digital menggambarkan pemrosesan data 2 dimensi apa pun. Citra digital adalah larik yang berisi nilai nyata atau kompleks yang diwakili oleh garis bit tertentu [8]. Dengan bantuan citra, informasi dapat tersaji dengan lebih jelas, karena terdapat bentuk visualisasi yang lebih mudah dipahami. Namun seringkali kualitas gambar yang kita terima berkurang atau menurun. Pemrosesan citra dilakukan agar citra yang terganggu mudah diinterpretasikan[19]. Citra digital adalah contoh gambar yang ditangkap oleh mesin menggunakan pendekatan sampling dan kuantisasi. Pada sebuah gambar, *Sample* menunjukkan besar kecilnya ukuran piksel gambar tersebut, dan *quantize* menunjukkan jumlah warna pada gambar tersebut [20].

Sebuah citra dapat didefinisikan sebagai fungsi $f(x,y)$ dengan M baris dan N kolom, dimana x dan y adalah koordinat spasial dan

amplitudo f dari titik koordinat (x,y) disebut intensitas atau skala keabuan. Gambar saat ini jika nilai x , y , dan amplitudo dari f semuanya berhingga dan memiliki nilai diskrit, maka citra tersebut dapat dikatakan sebagai citra digital. Gambar di bawah menunjukkan letak koordinat citra digital[8].



Gambar 2.2 Koordinat Citra Digital[8]

C. Jenis Citra

Tiga jenis citra yang umum digunakan dalam pemrosesan citra. Tiga jenis citra tersebut adalah citra berwarna, citra skala abu-abu dan citra biner.

1. Citra Berwarna

Citra berwarna, atau sering disebut dengan citra RGB, merupakan jenis citra dengan tiga komponen warna yaitu merah, hijau, dan biru dalam 8-bit, dengan masing-masing komponen berskala dari 0 hingga 255. warna mencapai $255 \times 255 \times 255$ atau 16.581.375 warna.

2. Citra Berskala Keabuan

Citra skala abu-abu ini diekspresikan oleh intensitas. Intensitasnya berkisar dari 0 hingga 255. Nilai 0 melambangkan hitam dan nilai 255 melambangkan putih.

Istilah lain untuk citra keabuan adalah citra keabuan, yaitu citra yang nilai pikselnya mewakili derajat intensitas

keabuan atau putih. Nilai intensitas terendah mewakili warna hitam dan nilai intensitas tertinggi mewakili warna putih. Pada umumnya citra grayscale memiliki kedalaman piksel 8 bit (256 tingkat keabuan), namun ada juga citra grayscale dengan kedalaman piksel yang bukan 8 bit, seperti 16 bit pada aplikasi yang membutuhkan resolusi tinggi. [8].

3. Citra Biner

Citra biner adalah citra yang memiliki nilai untuk setiap pikselnya, pilihannya hanya ada dua yaitu 0 dan 1. Nilai 0 berarti hitam pada latar belakang dan nilai 1 berarti putih pada latar depan.

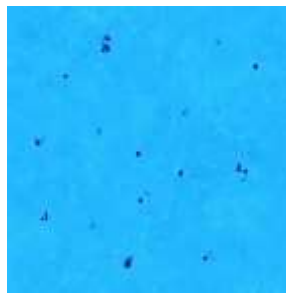
D. Image Processing

Image processing adalah metode atau teknik yang dapat digunakan untuk mengolah citra atau citra dengan cara memodifikasinya menjadi data citra yang dimuat untuk mendapatkan informasi tertentu tentang objek yang diamati. Citra digital dapat diolah oleh komputer karena berbentuk data numerik. Citra digital dengan pengolahan citra digital (*digital image processing*) menghasilkan citra digital baru yang meliputi restorasi citra dan peningkatan kualitas citra (*image enhancement*)[20]. Citra yang dimaksud adalah citra digital yang merupakan representasi dari keadaan nyata yang nyata sebagai kumpulan piksel dua dimensi yang dapat diproses oleh komputer. Banyak informasi dalam gambar, namun sering kali gambar yang kita terima rusak, misalnya terdapat error atau noise, warna terlalu kontras, kurang tajam, tidak jelas, dll. Citra dengan kualitas yang menurun dapat mengalami berbagai proses penyempurnaan untuk memperoleh kualitas citra yang lebih baik. Proses ini disebut pencitraan digital [12]. Teknik dan proses untuk mengurangi atau menghilangkan efek degradasi citra meliputi peningkatan citra atau teknik peningkatan citra,

restorasi citra dan transformasi khusus, topik pemrosesan citra digital lainnya meliputi pengkodean citra, segmentasi citra. Segmentasi citra, representasi citra dan deskripsi (*image representation and description*)[21]. Tahap – tahap *image processing* yang dilakukan pada penelitian ini sebagai berikut

1. Akuisisi Citra

Pada tahap ini merupakan tahap awal untuk mendapatkan citra digital. Berikut merupakan gambar akuisisi citra asli , sampel susu didapatkan dari Laboratorium Fakultas Peternakan Universitas Jenderal Soedirman. Akuisisi citra ini digunakan optilab untuk mengkonversi gambar dari mikroskop menjadi gambar digital berformat JPG dengan RGB Channel resolusi 850 x 850 *pixel*.

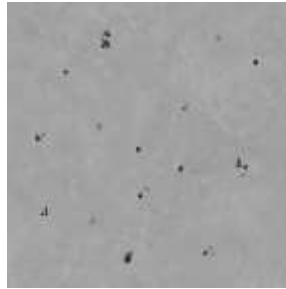


Gambar 2.3 Contoh Gambar Akuisisi Citra

2. *Grayscale*

Pada tahap ini dilakukan konversi dari citra *channel* RGB menjadi *grayscale*. Konversi ke grayscale ini digunakan untuk memisahkan antara background dengan objek menjadi lebih jelas. Pada skala RGB, jumlah piksel adalah R255 x G255 x B255 rentang warna. Sementara itu, pada skala abu-abu rentang warna yang digunakan adalah 0 sampai 255 sehingga memudahkan proses perhitungan. Konversi warna dari RGB ke *grayscale* bertujuan untuk melihat warna yang jelas perbedaan antara sel dan objek lain yang akan kemudian

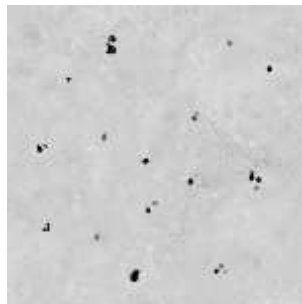
diubah menjadi skala biner. Dibawah ini merupakan hasil dari perubahan channel RGB ke *grayscale*.



Gambar 2.4 Konversi channel RGB ke grayscale

3. *Contrast Adjustment* / Peningkatan Kontras

Tahap selanjutnya grayscale disesuaikan agar kontrasnya meningkat. Peningkatan kontras akan menghasilkan gambar dimana nilai intensitas rendah akan semakin kecil mendekati 0, dan nilai intensitas tinggi akan meningkat mendekati 255. Dengan langkah ini, gambar akan terlihat lebih jelas untuk membedakan daerah gelap dan terang. Dibawah ini contoh gambar *grayscale* yang kontrasnya sudah ditingkatkan.

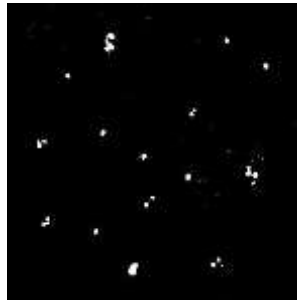


Gambar 2.5 Peningkatan Kontras

4. *Thresholding*

Pada penelitian ini digunakan segmentasi menggunakan metode *thresholding*. Proses memisahkan objek *foreground* dengan *background*. Hasil dari proses segmentasi adalah citra biner dengan logika 1 di latar depan dan logika 0 di latar

belakang, yang artinya nilai 1 menjadi putih dan nilai 0 menjadi hitam. Berikut adalah contoh gambar yang telah disegmentasi menggunakan metode *threshold* :



Gambar 2.6 *Thresholding*

E. OpenCV

OpenCV adalah singkatan dari *Open Computer Vision*, sebuah perpustakaan gratis yang dikembangkan oleh *Intel Corporation* khusus untuk pemrosesan gambar. *OpenCV* dirancang untuk efisiensi komputasi dengan fokus kuat pada aplikasi real-time. *OpenCV* ditulis dalam bahasa pemrograman C yang dioptimalkan dan dapat menggunakan prosesor *multi-core*. Tujuannya agar komputer memiliki kemampuan yang mirip dengan pemrosesan visual manusia. *OpenCV* menyediakan infrastruktur visi komputer yang mudah digunakan yang memungkinkan pengguna membuat aplikasi video yang cukup canggih dengan cepat. Perpustakaan *OpenCV* berisi lebih dari 500 fungsi yang mencakup banyak bidang termasuk inspeksi produk manufaktur, pencitraan medis, keamanan, antarmuka pengguna, kalibrasi kamera, penglihatan stereo, dan robotika. Berkat visi komputer dan pembelajaran mesin, OpenCV juga menyertakan pustaka pembelajaran mesin (MLL) yang lengkap dan serbaguna. Sublibrary ini berfokus pada pengenalan pola statistik dan pengelompokan. *OpenCV* memiliki API (*Application Programming Interface*) baik untuk level tinggi maupun rendah, ada fungsi yang siap untuk mendownload, menyimpan, menangkap gambar dan video[22].

F. Python

Python adalah bahasa pemrograman berorientasi dinamis yang dapat digunakan untuk banyak jenis perangkat lunak. Python menawarkan dukungan kuat untuk integrasi dengan bahasa dan alat pemrograman lain.

G. Anaconda Navigator

Anaconda Navigator adalah antarmuka pengguna grafis (GUI) desktop yang disertakan dengan distribusi *Anaconda*® yang memungkinkan pengguna meluncurkan aplikasi dan mengelola paket conda, lingkungan, dan saluran dengan mudah tanpa perintah baris perintah. *Navigator* dapat mencari paket di *Anaconda.org* atau di repositori *Anaconda* lokal yang tersedia untuk Windows, macOS, dan Linux.

Anaconda Navigator adalah cara mudah untuk menggunakan banyak alat lain di Navigasi Anaconda. Pengguna cukup mengarahkan dan mengklik untuk memanipulasi paket dan lingkungan *Navigator* tanpa harus mengetikkan perintah conda di jendela terminal.

H. Flask

Flask adalah *framework* yang dikembangkan dalam bahasa pemrograman Python. Flask ini dapat digunakan untuk menarik data yang dihasilkan oleh *Python* ke dalam aplikasi *Android*.

I. BLOB Detection

Blob Detection mendeteksi kumpulan titik piksel yang memiliki warna berbeda (lebih terang atau lebih gelap) dari latar belakang dan menggabungkannya menjadi satu wilayah[24]. Setiap objek/piksel putih yang menyatu akan dideteksi sebagai objek tunggal. Benda-benda tersebut kemudian akan diberikan pengenal atau penanda sebagai satu area yang terpisah satu sama lain. BLOB analisis memindai area gambar dari kiri atas ke bawah kanan dan menentukan setiap objek putih sebagai grup

independen. Dari sini, analisis BLOB melakukan fisik karakteristik objek seperti luas, keliling, pusat massa, panjang sumbu mayor-minor, dan soliditas [25].

J. Java

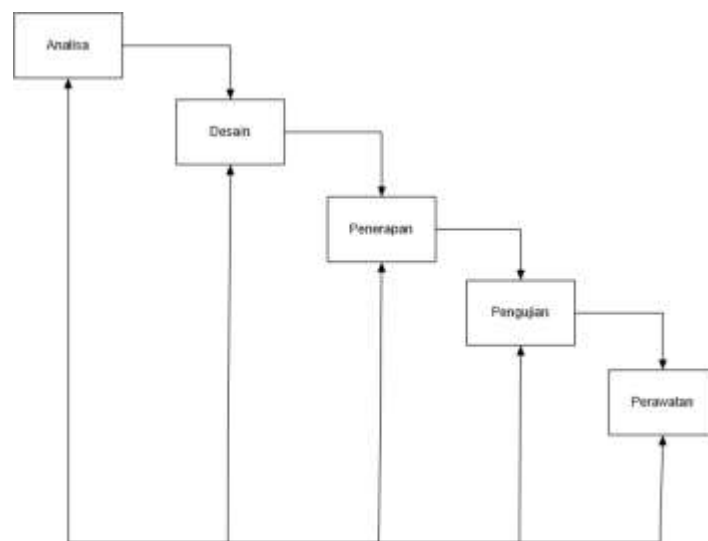
Java merupakan bahasa pemrograman yang sangat populer, bahasa pemrograman ini pertama kali dikembangkan oleh *Sun Microsystems* yang dibuat oleh James Gosling pada tahun 1995. Java adalah *Write Once and Run Anywhere* atau program yang ditulis sekali dan dapat digunakan di berbagai *platform*. Java adalah bahasa pemrograman yang dapat bekerja dengan *database*. Bahasa pemrograman *Java* merupakan bahasa berorientasi objek atau dikenal dengan *Object Oriented Programming* (OOP)[26]:

1. *Object Oriented*
2. *Platform Independent*
3. Sederhana
4. *Safe*
5. *Architectural-neutral*
6. *Portable*
7. *Strong*
8. *Multithreaded*
9. Terinterpretasi
10. Performa tinggi
11. Terdistribusi
12. Dinamis

K. Metode Waterfall

Model *Waterfall* proses desain dan pengembangan perangkat lunak yang memandang pengembangan sebagai gerakan ke bawah secara bertahap (seperti air terjun) melalui daftar langkah-langkah yang harus diselesaikan untuk berhasil membuat perangkat lunak komputer[27]. Ini membutuhkan

pendekatan yang sistematis dan berurutan untuk perangkat lunak [28]. Kelebihan dari model *waterfall* ini memungkinkan pembagian dan kontrol departemen, proses pengembangan model adalah model satu langkah, yang mengurangi kemungkinan kesalahan yang mungkin terjadi[29]. Model *waterfall* terdiri dari 5 fase yaitu analisis, desain, implementasi, pengujian dan pemeliharaan. Setiap tahapan model *waterfall* ditunjukkan pada Gambar 2.7.



Gambar 2.7 Fase Model Waterfall[30]

1. Analisis

Pada tahap analisis, dalam proses pengumpulan data, informasi diperoleh melalui perangkat lunak yang dikembangkan dengan cara yang mudah dipahami sesuai dengan kebutuhan pengguna. Teknik pengumpulan data untuk tahap analisis ini dapat berupa observasi, wawancara atau kuesioner. Data yang dikumpulkan harus didokumentasikan.

2. Desain

Desain pada perangkat lunak yaitu proses yang berfokus pada desain arsitektur perangkat lunak dan penyajian antarmuka pengguna. Fase desain berfungsi untuk menerjemahkan kebutuhan perangkat lunak dari

analisis ke fase desain. Desain yang dibuat berupa sketsa desain aplikasi menggunakan UML dan desain *user interface*.

3. Penerapan

Setelah desain dibuat, tahap selanjutnya yaitu membuat dan menerapkan aplikasi perangkat lunak.

4. Pengujian

Periksa dan validasi perangkat lunak selama fase pengujian. Karena berguna untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan langkah pengujian untuk menemukan kesalahan atau memperbaiki kesalahan selama pengembangan aplikasi.

5. Perawatan

Fase terakhir dari model *waterfall* ini adalah fase *maintenance*. Proses pemasangan dan perawatan produk dilakukan pada tahap perawatan. Karena tidak mengecualikan bahwa perangkat lunak berubah saat dikirimkan ke pengguna, perubahan tersebut mungkin disebabkan oleh bug yang muncul dan tidak terdeteksi selama pengujian, atau perangkat lunak harus beradaptasi dengan lingkungan baru[31].

L. Android Studio

Android studio merupakan sebuah *Integrated Development Environment* (IDE) untuk platform *Android*. Manajer Produk Google *Ellie Powers* mengumumkan *Android Studio* pada 16 Mei 2013 di *Google I/O Conference*. *Android Studio* gratis di bawah Lisensi *Apache 2.0*. *Android Studio*, berdasarkan *JetBrainns IntelliJ IDEA*, dirancang khusus untuk pengembangan *Android* (*SMART BABY MONITORING "SYSTEM"*). Selain itu, *Android Studio* memiliki fitur peningkat produktivitas saat membuat aplikasi, seperti [32],

1. Sistem versi berbasis skala yang fleksibel
2. Emulator yang cepat dan serbaguna
3. Lingkungan pengembangan terpadu untuk semua perangkat *Android*
4. *Instant Run* mengubah aplikasi yang sedang berjalan

tanpa membuat APK baru

5. Kode model dan integrasi *Github* untuk membangun fungsionalitas aplikasi yang sama dan mengimpor kode model
6. Memiliki alat pengujian yang ekstensif

M. BlackBox Testing

BlackBox Testing merupakan pengujian yang berfokus pada spesifikasi fungsional perangkat lunak. Penguji dapat menentukan serangkaian kondisi masukan dan menjalankan pengujian sesuai dengan spesifikasi fungsional program[22]. Pengujian *Blackbox* yang umum digunakan adalah fungsional. Pengujian fungsional menggunakan teknik pengujian *blackbox*, di mana penguji menjalankan spesifikasi dan merancang kasus uji untuk menentukan apa arti kode tersebut. Tes fungsional memastikan bahwa fungsi yang didefinisikan dalam spesifikasi persyaratan berfungsi[33]. *BlackBox Testing* biasanya untuk menemukan hal berikut[22]:

1. Fungsionalitas salah atau hilang.
2. *UI bug* (interface errors).
3. Kesalahan dalam struktur data dan penggunaan basis data.
4. *Performance errors*.
5. Kesalahan inisialisasi dan terminasi.

Saat ini banyak metode atau teknik untuk melaksanakan *BlackBox Testing*, antara lain:

1. Kesetaraan Partisi
2. Analisis Kendala/Uji Batas
3. Uji Perbandingan
4. Uji Sampel
5. Uji Daya Tahan
6. *Behavior Testing*
7. *Requirement Testing*
8. *Performance Testing*

9. Uji Ketahanan (*Endurance Testing*)
10. Uji Sebab-Akibat (*Cause-Effect Relationship Testing*)