

BAB I

PENDAHULUAN

1.1 Pengenalan Masalah

Pengelolaan makanan yang kurang baik di dalam kulkas dapat menyulitkan pengguna dalam memantau kondisi makanan yang mulai membusuk dan mendekati masa kedaluwarsa, yang menyebabkan terjadinya pemborosan makanan. Indonesia merupakan negara dengan jumlah sampah makanan terbanyak di Asia Tenggara. Pada tahun 2021 lalu, jumlah sampah makanan di Indonesia diprediksi mencapai 20,93 juta ton (28,3% dari total sampah yang diproduksi selama tahun 2021) dan merupakan kontributor utama sampah [1].

Lemari pendingin atau yang sering disebut kulkas merupakan sebuah perangkat elektronik penting dalam rumah tangga yang berfungsi untuk menjaga keawetan makanan agar dapat disimpan dalam jangka waktu yang relatif lama. Meskipun demikian, masih terdapat permasalahan dalam hal pengorganisasian dan pemantauan kondisi makanan di dalam kulkas, yang mana ini merupakan salah satu faktor utama yang menyebabkan pembusukan makanan. Tata letak makanan yang tidak baik sering kali menyebabkan pembusukan makanan terutama buah dan sayur lebih cepat. Banyak pengguna kulkas yang lupa kapan memasukan makanan di dalam kulkas, sehingga mereka tidak mengetahui apakah makanan tersebut masih layak dikonsumsi atau sudah melewati masa simpan optimalnya. Selain itu banyak orang menganggap bahwa menyimpan makanan di kulkas, makanan akan bertahan dalam waktu lama sehingga pengguna jarang untuk melakukan pengecekan makanan [2]. Hal ini menyebabkan banyak makanan terbuang sia-sia.

Menurut studi yang dilakukan oleh Antara News dan SurabayaOnline.co menunjukkan bahwa kurangnya pengelolaan makanan di kulkas merupakan salah satu penyebab terbuangnya makanan yang cukup tinggi di rumah tangga. Salah satu penyebabnya adalah tidak adanya sistem pemantauan yang efektif untuk mengetahui apakah makanan masih layak atau sudah busuk. Salah satu tantangan terbesar adalah tidak adanya sistem pencatatan otomatis yang dapat membantu pengguna melacak umur simpan makanan serta memberikan pengingat sebelum makanan mencapai masa kedaluwarsa [3][4].

Dampak dari permasalahan ini dapat mempengaruhi berbagai pihak. Mulai dari rumah tangga dan individu yang merupakan pengguna utama kulkas terdampak dari sisi pemborosan makanan dan kesehatan, produsen kulkas yang dapat memanfaatkan permasalahan ini sebagai peluang untuk mengembangkan produk mereka seperti membuat kulkas pintar, dan terakhir pemerintah dan lembaga kesehatan juga ikut terdampak karena mereka memiliki kepentingan dalam mengurangi limbah makanan, dan menjaga keamanan pangan.

Jika masalah ini dapat diselesaikan, dampaknya akan sangat dirasakan oleh semua *stakeholder* terkait. Rumah tangga dan individu akan dimudahkan dalam pengelolaan makanan di kulkas, yang mana ini nantinya akan mengurangi resiko pemborosan makanan. Produsen kulkas dapat memanfaatkan solusi inovatif ini untuk kemajuan usahanya dengan menambahkan fitur baru untuk meningkatkan daya saing, misalnya dengan menambahkan fitur-fitur pintar pada kulkas yang memungkinkan pengguna untuk lebih mudah mengorganisir dan memonitor makanan di dalam kulkas.

1.1.1 Informasi Pendukung Masalah

Pada penelitian ini, masalah yang diangkat terkait dengan pengorganisasian makanan di dalam kulkas terkhususnya bagian bawah (*refrigerator*). Kulkas ini dirancang untuk mengoptimalkan penyimpanan untuk berbagai jenis makanan. Berikut Informasi teknis mengenai kulkas pada umumnya:

1. Kapasitas penyimpanan : 150-300L
2. Suhu : 1°C hingga 7°C, dengan sistem pendinginan *fun cooling*.
3. Pencahayaan : menggunakan LED hemat energi, guna untuk melihat secara jelas isi kulkas.
4. Ukuran : Lebar (60-70cm), tinggi (65-75cm), dalam (65-75cm).

Bagian *refrigerator* sering kali digunakan sebagai tempat penyimpanan sayur, buah, dan makanan siap saji atau makanan selain buah dan sayur yang memiliki masa simpan lebih pendek dibandingkan makanan beku. Setiap makanan memiliki masa simpan yang berbeda-beda di dalam kulkas, misalnya [5]:

1. Buah dan sayur segar : 3-10 hari
2. Daging segar : 3-5 hari

3. Makanan siap saji : 2-4 hari
4. Produk susu (keju/yogurt) : 7-14 hari
5. Makanan dalam kemasan : mengikuti masa kedaluwarsa makanan

Bagian *refrigerator* kulkas terdiri dari 3 bagian yaitu rak atas, rak tengah, dan rak bawah. Untuk rak atas digunakan sebagai tempat menyimpan makanan yang sudah dimasak, sisa makanan, atau bahan pangan yang tidak perlu dimasak, seperti daging, ikan, telur, dll. Rak tengah digunakan sebagai tempat menyimpan bahan makanan mentah seperti keju, yogurt, sayur, atau lauk pauk yang sudah dimasak. Sedangkan pada rak bawah digunakan untuk menyimpan makanan yang membutuhkan kelembapan lebih seperti sayuran dan buah-buahan. Tapi untuk bagian rak bawah ini juga dibagi atas 2 yaitu makanan dengan kebutuhan kelembapan tinggi seperti kol, brokoli, cabai, dan wortel dan makanan dengan kebutuhan kelembapan rendah seperti anggur, apel, jamur, dan pir. Yang mana untuk kedua kategori ini letaknya harus dipisahkan [6][5].

Dari berbagai makanan yang biasanya umumnya dimasukkan orang ke dalam kulkas, dapat dibagi menjadi 5 bagian, yaitu buah, sayur, daging, makanan siap saji, dan minuman.

Pada penelitian [7] telah pernah dirancang alat berupa Purwarupa Sistem Monitoring Makanan Kulkas Pintar Berbasis IoT. Namun, masih terdapat beberapa kekurangan yaitu alat tidak maksimal karena kurangnya pencabayaan, masih terjadi delay, dan memerlukan daya yang tidak boleh terputus. Dan juga pernah dilakukan penelitian sebelumnya dengan fokus pemantauan kondisi makanan di dalam kulkas berbasis IoT menggunakan sensor suhu dan kelembaban, namun sistem ini masih memiliki keterbatasan dimana sistem hanya mampu mendeteksi dan kelembaban tanpa adanya indikasi pembusukan yang dapat di deteksi secara langsung [8].

1.1.2 Analisis Masalah

Pengorganisasian makanan yang baik di dalam kulkas memiliki dampak yang besar dalam berbagai aspek, baik itu aspek ekonomi, lingkungan, kesehatan, maupun sosial.

Dari segi ekonomi, penyimpanan makanan yang teratur pada kulkas dapat mengurangi pemborosan makanan yang signifikan, karena mengurangi terbuangnya makanan akibat busuk atau kedaluwarsa. Hal ini memungkinkan rumah tangga maupun individu menghemat anggaran belanja. Dengan implementasi sistem yang dapat memonitor kondisi makanan di kulkas, diharapkan pemborosan makanan dapat diminimalkan, sehingga dapat meningkatkan efisiensi pengeluaran rumah tangga dan membuka peluang bagi produsen kulkas untuk mengembangkan produk inovatif yang meningkatkan daya saing di pasar.

Dari aspek lingkungan, pengurangan limbah sangat penting untuk menjaga kelestarian sumber daya alam, dengan pengelolaan makanan yang baik di kulkas, dapat mengurangi limbah makanan yang mana limbah ini berkontribusi terhadap peningkatan emisi gas rumah kaca dan kerusakan lingkungan. Ketika makanan terbuang ke tempat pembuangan akhir (TPA), sampah makanan ini menghasilkan gas metana, yang merupakan gas rumah kaca yang lebih kuat dari karbon dioksida. Selain itu, sampah makanan yang menumpuk juga menambah beban pada sistem pengelolaan sampah yang sudah ada.

Dari segi Kesehatan, makanan yang tersimpan terlalu lama atau membusuk dalam kulkas dapat menimbulkan risiko kesehatan bagi pengguna. Makanan busuk selain kehilangan nilai nutrisinya juga dapat menjadi media pertumbuhan bagi bakteri seperti E. coli, Salmonella, dan berbagai patogen lainnya yang berbahaya jika dikonsumsi. Selain itu, gas etilen yang dihasilkan oleh buah-buahan dan sayuran yang membusuk dapat mempercepat proses pembusukan pada makanan lain yang berada di dekatnya, sehingga memperbesar risiko kontaminasi.

1.1.3 Kebutuhan yang Harus Dipenuhi

Berdasarkan analisis yang telah dilakukan, kebutuhan yang harus dipenuhi untuk menyelesaikan permasalahan adalah sebagai berikut :

1. Sistem yang mampu mendeteksi dan mengelompokkan makanan.
2. Sistem yang dapat memberikan saran tata letak makanan.
3. Sistem yang dapat memantau dan memberi peringatan kondisi makanan.



1.1.4 Tujuan

Berdasarkan permasalahan yang telah dijabarkan, penulis bertujuan membuat sebuah sistem berbasis teknologi yang dapat membantu meminimalisir pemborosan atau terbuangnya makanan di dalam kulkas dengan cara yang efisien dan terintegrasi. Dengan adanya fitur pengelompokan dan pemantauan makanan dalam kulkas, diharapkan mampu memperpanjang masa simpan makanan, mencegah terbuangnya makanan karena basi, serta mengirimkan notifikasi secara real-time kepada pengguna agar mereka dapat mengambil tindakan tepat.

1.2 Solusi

1.2.1 Karakteristik Produk

a. Fitur Dasar

1. *Accurate data processing*

Sistem yang dirancang harus dapat melakukan pengolahan data secara tepat dan akurat.

2. *Categorizing food capability*

Sistem yang dirancang harus dapat mengelompokkan dan menentukan letak makanan yang tepat.

3. *Monitor food conditions capability*

Sistem yang dirancang harus dapat memantau kondisi makanan secara akurat.

4. *Computational method*

Sistem yang dirancang harus menggunakan metode komputasi untuk pemrosesan datanya.

5. *Reminder*

Sistem yang dirancang dapat memberitahu pengguna terkait kondisi makanan yang busuk.

b. Fitur Tambahan

1. *Low Power Consumption*

Sistem dirancang untuk mengkonsumsi daya rendah agar efisien dan ramah lingkungan.



2. *Low cost*

Biaya yang dikeluarkan untuk membeli komponen harus serendah mungkin.

3. *Real-time*

Alat yang dirancang harus memberikan Informasi secara *real-time*

4. *Compatibility*

Sistem yang dirancang harus mudah diimplementasikan dan dapat diintegrasikan dengan sistem yang sudah ada.

5. *Aesthetic Design*

Sistem yang dirancang enak dipandang dan praktis.

6. Alat Selesai dalam 6 Bulan

Alat harus selesai dalam jangka waktu 6 bulan.

1.2.2 Usulan Solusi

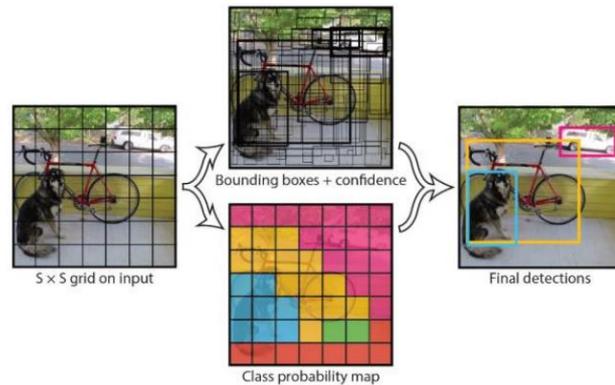
1.2.2.1 Sistem pengelompokan dan pemantauan makanan otomatis menggunakan kamera dengan algoritma YOLO

Pada solusi pertama, dibuat sebuah sistem yang bertujuan untuk menata makanan dalam kulkas secara otomatis menggunakan kamera. Cara kerjanya, sebelum makanan dimasukkan ke dalam kulkas, kamera akan mendeteksi jenis makanan. Dimana makanan ini nantinya akan dikategorikan dalam 5 kelompok, yaitu sayur, buah, makanan siap saji, daging, dan minuman, setelah diidentifikasi nanti sistem akan memberikan saran kepada pengguna untuk meletakkan makanannya di rak 1, 2, 3, atau 4.

Kamera ini nantinya akan dilengkapi dengan algoritma *You Only Look Once* (YOLO), yang merupakan algoritma *real-time object detection*. YOLO akan memproses gambar makanan untuk mengidentifikasi jenisnya dengan cepat dan akurat. Berdasarkan hasil deteksi, sistem akan menentukan lokasi terbaik untuk menyimpan makanan di dalam kulkas [9].

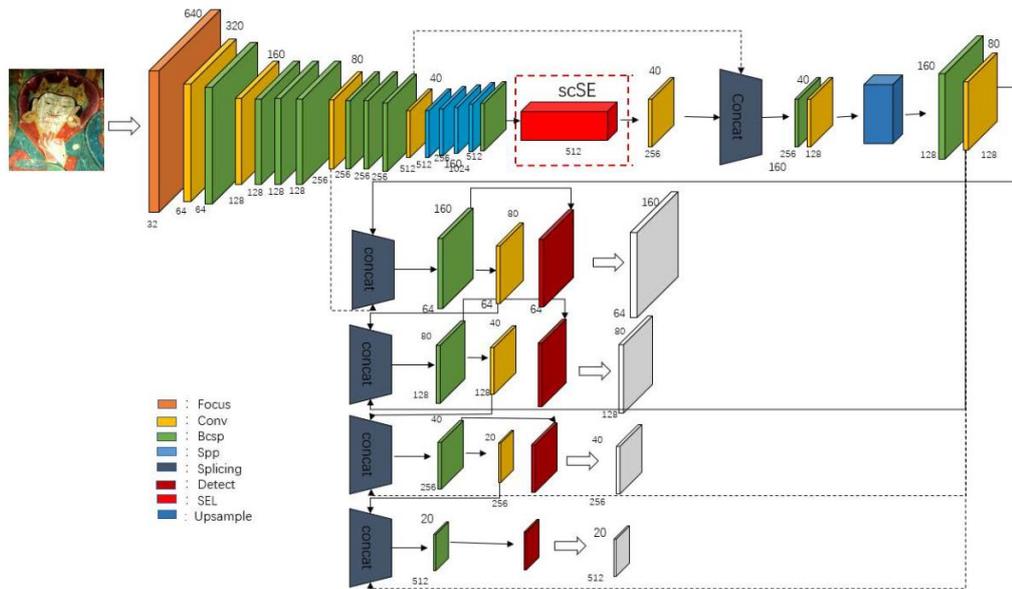
You Only Look Once (YOLO) adalah salah satu metode yang digunakan untuk deteksi objek. Berbeda dari metode deteksi objek lain seperti sliding window dan region proposal-based, YOLO mengubah deteksi objek menjadi masalah regresi tunggal. Secara umum, YOLO membagi gambar input menjadi grid dengan ukuran $S \times S$. Setiap cell pada grid tersebut menghasilkan satu set prediksi yang terdiri dari

dua komponen utama: koordinat bounding box dan skor kelas. Setelah itu, dilakukan proses non-maximum suppression untuk menghilangkan prediksi yang memiliki skor keyakinan (confidence score) rendah di sekitar prediksi dengan skor keyakinan tinggi [9]. Metode ini sering dipakai karena memiliki keunggulan kecepatan deteksi yang lebih cepat dibanding metode lain yang seperti Faster RCNN dan SSD [10]. Algoritma YOLO digambarkan pada Gambar 1.1 [9] :



Gambar 1.1 Algoritma proses deteksi pada YOLO

You Only Look Once atau YOLO adalah sebuah algoritma untuk mengenali objek yang berdasarkan pada Convolutional Neural networks (CNN). YOLO menggunakan pendekatan jaringan saraf tiruan (JST) untuk mendeteksi objek pada sebuah citra. Jaringan ini membagi citra menjadi beberapa wilayah dan memprediksi setiap kotak pembatas dan probabilitas untuk setiap wilayah. Kotak-kotak pembatas ini kemudian dibandingkan dengan probabilitas yang diprediksi. YOLO memiliki beberapa kelebihan dibandingkan dengan sistem yang berorientasi pada classifier, terlihat dari seluruh citra pada saat dilakukan test dengan prediksi yang diinformasikan secara global pada citra[9]. Metode You Only Look Once (YOLO) merupakan salah satu metode yang paling cepat dan akurat pada pendeteksian objek bahkan mampu melebihi hingga 2 kali kemampuan algoritma lain. YOLO mempunyai banyak versi yang sering diterapkan yaitu mulai versi YOLO, YOLOv2 hingga yang terbaru saat ini adalah YOLOv8[11].



Gambar 1.2 *Picture size and channel number of each layer of YOLOv5-scSE*[12]. Berdasarkan gambar 1.2, dapat dilihat YOLOv5 memiliki 32 layer yang terdiri dari beberapa jenis blok yang bekerja untuk mengekstrak fitur, memproses Informasi, dan yang terakhir mendeteksi objek. Berikut penjelasan proses YOLO berdasarkan lapisannya [12] :

1. Focus layer merupakan langkah pertama dalam arsitektur YOLOv5 yang bertujuan mengurangi ukuran gambar input dari resolusi tinggi menjadi resolusi yang lebih kecil. Proses ini dilakukan dengan memotong gambar menjadi beberapa bagian kecil, memungkinkan informasi penting tetap terjaga sambil mengurangi beban komputasi.
2. Convolutional Layers (Conv) bertanggung jawab untuk mengekstrak fitur dari gambar input. Lapisan ini menggunakan operasi konvolusi untuk mengenali pola-pola dasar seperti tepi, tekstur, dan bentuk. Setiap lapisan konvolusi diikuti oleh proses Batch Normalization untuk stabilitas pelatihan dan fungsi aktivasi seperti Leaky ReLU untuk menangkap non-linearitas.
3. Bcsp (Bottleneck CSP) menggunakan pendekatan Cross Stage Partial untuk menangani informasi secara efisien. Blok ini membagi jalur data menjadi dua bagian: satu jalur langsung melewati layer, sementara jalur lainnya digunakan untuk transformasi fitur. Teknik ini membantu mengurangi parameter model tanpa kehilangan informasi penting.

4. Spatial Pyramid Pooling (SPP) memperluas receptive field agar dapat menangkap konteks yang lebih luas dalam gambar. Proses ini dilakukan dengan pooling menggunakan berbagai ukuran kernel, sehingga model mampu mengintegrasikan informasi dari area yang lebih besar.
5. SCSE (Squeeze-and-Excitation dengan Spatial Attention) adalah mekanisme perhatian yang membantu model fokus pada bagian gambar yang lebih relevan. Lapisan ini menonjolkan fitur penting dalam dimensi saluran dan spasial, sehingga meningkatkan akurasi deteksi objek.
6. Concat Layers digunakan untuk menggabungkan fitur dari berbagai tingkat dalam model. Proses ini penting untuk memastikan bahwa informasi multi-skala, baik dari jalur utama maupun alur tambahan, tersedia dalam tahap deteksi.
7. Upsample Layers bertugas mengembalikan resolusi fitur agar cocok untuk mendeteksi objek pada skala yang lebih kecil. Ini membantu mempertahankan detail penting dari gambar asli setelah downsampling.
8. Detection Layers adalah tahap akhir yang bertugas membuat prediksi lokasi dan kelas objek. Lapisan ini bekerja pada tiga skala berbeda (grid 80x80, 40x40, dan 20x20) untuk mendeteksi objek dengan ukuran besar, sedang, dan kecil secara efektif.

Selain itu, sistem ini akan dihubungkan dengan aplikasi, di mana aplikasi ini nantinya berfungsi untuk memantau makanan yang sudah dimasukkan ke dalam kulkas. Cara kerjanya, setelah sistem mendeteksi dan mengelompokkan makanan menggunakan algoritma YOLO, data hasil deteksi, termasuk jenis makanan dan kategori penyimpanannya, akan otomatis masuk ke dalam database. Setelah pengguna memasukkan makanan ke dalam kulkas, sistem akan mencatat tanggal penyimpanan makanan secara otomatis di aplikasi.

Aplikasi ini akan menyimpan tanggal masuk makanan ke kulkas dan memberikan rekomendasi berapa lama makanan tersebut dapat bertahan berdasarkan kategori makanan yang terdeteksi. Estimasi masa simpan akan didasarkan pada database standar daya tahan makanan dalam kondisi kulkas optimal. Setiap makanan yang tersimpan akan ditampilkan dalam daftar monitoring di aplikasi, sehingga

pengguna dapat dengan mudah melihat makanan apa saja yang ada di kulkas beserta perkiraan waktu kedaluwarsanya.

Sistem ini juga akan dilengkapi dengan notifikasi pengingat, yang dikirimkan ke pengguna berdasarkan estimasi waktu kedaluwarsa makanan. Notifikasi ini memiliki tiga tahap pengingat, yaitu :

1. Pengingat awal : Dikirim 2 hari sebelum kedaluwarsa, dengan pesan: "Makanan ini akan segera mencapai batas penyimpanan optimal, segera konsumsi untuk menghindari pembusukan!".
2. Pengingat utama : Dikirim 1 hari sebelum kedaluwarsa, dengan pesan: "Makanan ini hampir melewati batas masa simpan ideal. Segera gunakan atau pertimbangkan untuk memeriksa kondisinya".
3. Pengingat terakhir : Dikirim pada hari kedaluwarsa, dengan pesan: "Makanan ini telah melewati batas penyimpanan optimal. Periksa apakah masih layak dikonsumsi!".

Selain itu, pengguna dapat memperbarui status makanan di aplikasi. Jika makanan sudah dikonsumsi atau dikeluarkan dari kulkas, pengguna dapat menandai makanan tersebut sebagai 'Sudah Digunakan', sehingga sistem akan menghapus makanan dari daftar monitoring. Hal ini memastikan bahwa notifikasi yang diterima pengguna tetap relevan dan akurat.

Dengan adanya aplikasi ini, pengguna dapat mengelola makanan dengan lebih baik, mengetahui makanan apa saja yang ada di kulkas beserta masa simpannya, serta menghindari pemborosan makanan dengan cara yang lebih sistematis dan praktis.

Komponen yang dibutuhkan :

- a. Kamera : kamera digital dengan kemampuan resolusi gambar cukup tinggi
- b. Prosesor : perangkat yang memiliki kemampuan untuk menjalankan algoritma YOLO secara efisien.
- c. Algoritma YOLO : untuk deteksi objek secara *real-time* dan identifikasi jenis makanan.
- d. Database Penyimpanan : Menyimpan informasi makanan yang terdeteksi, termasuk tanggal masuk makanan dan estimasi masa simpan berdasarkan jenisnya.

- e. Display (Monitor LCD) : Menampilkan hasil deteksi YOLO, saran tempat penyimpanan, serta daftar makanan yang telah terdaftar dalam sistem.
- f. Aplikasi Android : Berfungsi untuk monitoring makanan dalam kulkas, memberikan notifikasi otomatis sebelum makanan kedaluwarsa, serta memungkinkan pengguna untuk memperbarui status makanan (sudah dikonsumsi atau dikeluarkan dari kulkas).
- g. Modul Wi-Fi atau Bluetooth : Untuk menghubungkan sistem kulkas dengan aplikasi Android, sehingga data makanan dapat diperbarui secara real-time dan disinkronkan dengan database.

Solusi pertama ini menggunakan metode Computer Vision dengan algoritma YOLO untuk mendeteksi dan mengidentifikasi makanan berdasarkan gambar yang diambil oleh kamera. Setelah makanan dikenali, sistem akan memberikan rekomendasi tempat penyimpanan optimal di kulkas berdasarkan jenis makanan yang terdeteksi.

Selain itu, sistem akan mencatat tanggal masuk makanan ke dalam database dan memperkirakan masa simpan optimal berdasarkan kategori makanan. Pengguna dapat memantau daftar makanan dalam kulkas melalui aplikasi Android, yang akan memberikan notifikasi otomatis sebelum makanan mencapai batas kedaluwarsa.

Sistem ini menggunakan IoT untuk menghubungkan kulkas dengan aplikasi Android melalui Wi-Fi atau Bluetooth, memungkinkan sinkronisasi data secara real-time. Aplikasi juga memungkinkan pengguna memperbarui status makanan, seperti menandai makanan yang sudah dikonsumsi atau dikeluarkan dari kulkas.

Dengan solusi ini, pengguna dapat mengorganisir makanan dengan lebih efisien, memantau masa simpannya, serta mengurangi pemborosan makanan secara sistematis.

1.2.2.2 Sistem pengelompokan dan pemantauan makanan otomatis menggunakan sensor gas dan kamera menggunakan machine learning dengan algoritma SSD dan Fuzzy

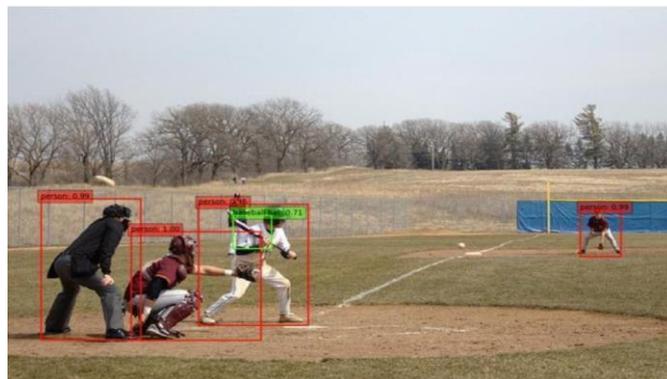
Pada solusi pertama, dibuat sebuah sistem yang bertujuan untuk menata makanan dalam kulkas secara otomatis menggunakan kamera. Cara kerjanya, sebelum

makanan dimasukkan ke dalam kulkas, kamera akan mendeteksi jenis makanan. Dimana makanan ini nantinya akan dikategorikan dalam 5 kelompok, yaitu sayur, buah, makanan siap saji, daging, dan minuman, setelah diidentifikasi nanti sistem akan memberikan saran kepada pengguna untuk meletakkan makanannya di rak 1, 2, 3, atau 4.

Kamera ini nantinya akan dilengkapi dengan algoritma *Single Shot Multibox Detector* (SSD), sebuah metode *object detection* berbasis *deep learning* yang mampu mendeteksi dan mengenali objek secara cepat dalam satu proses. Algoritma SSD memproses gambar makanan dan mengidentifikasinya untuk memberikan rekomendasi penyimpanan yang tepat [13].

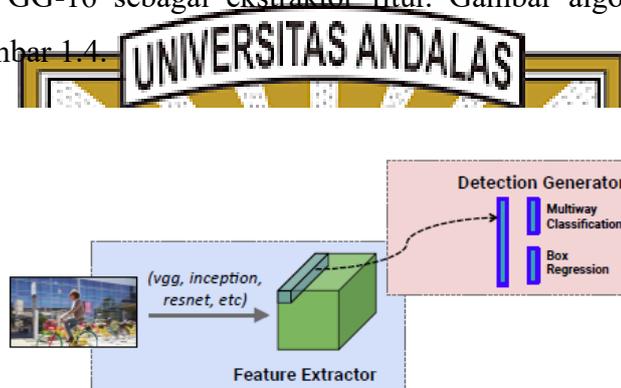
Single Shot Detector (SSD) adalah metode yang digunakan untuk mengenali atau mendeteksi objek dalam gambar menggunakan deep neural network tunggal. Algoritma ini populer karena mudah diimplementasikan, memiliki akurasi yang baik, dan efisien dalam hal komputasi. SSD hanya memerlukan satu pemrosesan tunggal untuk mendeteksi beberapa objek dalam satu gambar.

SSD juga termasuk dalam kategori metode deteksi objek real-time, dan lebih intuitif dibandingkan dengan metode lain seperti R-CNN, Fast R-CNN, Faster R-CNN, serta You Only Look Once (YOLO). SSD dikenal sebagai algoritma yang sangat andal, dengan desain sederhana dan implementasi langsung melalui GPU. Dengan kerangka kerja pembelajaran yang mendalam, SSD mampu menangani tugas deteksi objek secara tepat dan efisien. Salah satu keunggulan utama dari SSD adalah kemampuannya untuk memberikan pemahaman yang lebih baik dibandingkan metode canggih lainnya [13].



Gambar 1.3 Deteksi objek menggunakan metode Single Shot Detector

Cara kerjanya, SSD ini menggunakan *grid anchor* pada gambar, yaitu kotak-kotak yang tersebar dengan bermacam posisi, skala, dan rasio aspek. SSD menghasilkan vektor prediksi untuk setiap kotak yang mencakup confidence score, yaitu tingkat keyakinan terhadap kehadiran kelas objek tertentu dalam kotak tersebut, serta localization offset, yaitu perubahan posisi dan ukuran agar kotak lebih akurat menangkap objek. Untuk mendeteksi objek dengan ukuran yang bervariasi, SSD bekerja di beberapa resolusi peta fitur (feature maps). Misalnya, pada gambar berukuran 512×512, SSD memulai prediksi dari peta fitur 64×64 dan terus berlanjut ke resolusi yang lebih kecil hingga 1×1, menggunakan hingga 7 peta fitur saat menggunakan VGG-16 sebagai ekstraktor fitur. Gambar algoritma SSD dapat dilihat pada Gambar 1.4.



Gambar 1.4 Algoritma SSD

Selain itu sistem ini akan dilengkapi dengan 4 sensor gas yang ditempatkan di setiap rak kulkas, sensor ini akan mendeteksi keberadaan gas pembusukkan yang dihasilkan makanan. Data gas yang dihasilkan oleh sensor akan diproses menggunakan algoritma Fuzzy Logic untuk meningkatkan akurasi dalam pendeteksian dan pengambilan Keputusan, sama seperti solusi pertama.

Notifikasi kemudian akan dikirimkan melalui Telegram untuk memperingatkan pengguna agar segera memeriksa dan membuang makanan yang busuk di rak yang terdeteksi sebelum mengkontaminasi ke makanan yang lainnya.

Komponen yang dibutuhkan :

- a. Kamera : kamera digital dengan kemampuan resolusi gambar cukup tinggi
- b. Prosesor : perangkat yang mendukung pemrosesan algoritma SSD dengan efisien.
- c. Algoritma SSD : untuk deteksi dan klasifikasi objek secara *real-time*.

- d. Sensor gas : untuk mendeteksi gas etilen yang dihasilkan oleh makanan yang membusuk
- e. Display : Monitor untuk memberi tahu pengguna letak yang tepat dari makanan yang di deteksi dan memberikan peringatan kepada pengguna jika ada makanan yang busuk di dalam kulkas.
- f. Telegram API : sistem bisa menggunakan bot Telegram yang diatur untuk memberikan notifikasi tentang kondisi makanan yang mulai membusuk di kulkas
- g. Modul Wi-Fi atau Bluetooth : untuk menghubungkan sistem dengan aplikasi mobile melalui jaringan

Untuk Solusi kedua ini digunakan metode komputasi Computer Vision dan algoritma SSD untuk mendeteksi dan mengidentifikasi makanan berdasarkan gambar yang diambil kamera. Dan IoT untuk pemantauan sensor gas secara real-time untuk mengidentifikasi makanan busuk dan mengirimkan peringatan ke web yang dirancang.

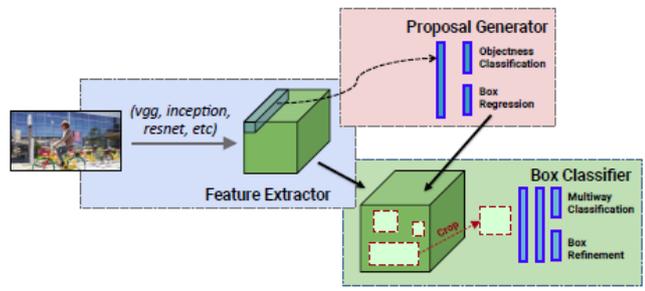
1.2.2.3 Sistem pengelompokan dan pemantauan makanan otomatis menggunakan sensor gas dan kamera menggunakan machine learning dengan algoritma Faster R-CNN dan Fuzzy

Pada solusi ketiga, dibuat sebuah sistem yang bertujuan untuk menata makanan dalam kulkas secara otomatis menggunakan kamera. Cara kerjanya, sebelum makanan dimasukkan ke dalam kulkas, kamera akan mendeteksi jenis makanan secara otomatis. Dimana makanan ini nantinya akan dikategorikan dalam 5 kelompok, yaitu sayur, buah, makanan siap saji, daging, dan minuman, setelah diidentifikasi nanti sistem akan memberikan saran kepada pengguna untuk meletakkan makanannya di rak 1, 2, 3, atau 4.

Kamera ini nantinya akan dilengkapi dengan algoritma *Faster R-CNN (Region-based Convolutional Neural Networks)* untuk mendeteksi dan mengelompokkan makanan di dalam kulkas secara otomatis. *Faster R-CNN* adalah salah satu algoritma *object detection* berbasis deep learning yang sangat kuat dan akurat, meskipun memerlukan sumber daya komputasi yang lebih tinggi dibandingkan beberapa algoritma lainnya.

Faster R-CNN adalah metode deep learning yang sering digunakan untuk pendeteksian objek, dikembangkan dari algoritma R-CNN untuk mengatasi kelemahannya. Faster R-CNN merupakan versi yang disempurnakan dari Fast R-CNN. Perbedaan utama antara keduanya adalah bahwa Fast R-CNN menggunakan Selective Search untuk menghasilkan region proposal, sementara Faster R-CNN menggunakan Region Proposal Network (RPN). RPN adalah jaringan saraf yang menggantikan Selective Search karena proses Selective Search yang lambat, sekitar 2 detik per gambar. RPN berfungsi menghasilkan beberapa bounding box, di mana setiap box memiliki dua skor probabilitas yang menunjukkan apakah terdapat objek di lokasi tersebut atau tidak. Dengan adanya RPN, pemrosesan tidak perlu dilakukan berulang-ulang seperti pada R-CNN memungkinkan model dilatih secara end-to-end [7].

Deteksi Faster R-CNN terjadi dalam dua tahap. Tahap pertama adalah region proposal, di mana gambar diproses oleh feature extractor seperti ResNet50 hingga lapisan jaringan tertentu. Kemudian, jaringan kecil memprediksi kotak proposal yang bersifat class-agnostic berdasarkan grid anchors yang tersebar dengan berbagai posisi, skala, dan rasio aspek. Pada tahap kedua, kotak proposal ini digunakan untuk memotong fitur dari peta fitur yang dihitung di tahap pertama. Kotak proposal ini kemudian diproses lebih lanjut oleh feature extractor untuk melakukan klasifikasi dan perbaikan posisi kotak spesifik objek. Meskipun Faster R-CNN menghindari perhitungan berulang dengan berbagi peta fitur antara tahap proposal dan klasifikasi, waktu komputasinya masih bergantung pada jumlah proposal daerah yang dihasilkan [14]. Berikut gambar algoritma Faster R-CNN :



Gambar 1.5 Algoritma Faster R-CNN

Selain itu sistem ini akan dilengkapi dengan 4 sensor gas yang ditempatkan di setiap rak kulkas, sensor ini akan mendeteksi keberadaan gas pembusukkan yang dihasilkan makanan. Data gas yang dihasilkan oleh sensor akan diproses menggunakan algoritma Fuzzy Logic untuk meningkatkan akurasi dalam pendeteksian dan pengambilan Keputusan, sama seperti solusi pertama.

Notifikasi kemudian akan dikirimkan melalui Telegram untuk memperingatkan pengguna agar segera memeriksa dan membuang makanan yang busuk di rak yang terdeteksi sebelum mengkontaminasi ke makanan yang lainnya.

Komponen yang dibutuhkan :

- a. Kamera : kamera digital dengan kemampuan resolusi gambar cukup tinggi
- b. Prosesor : perangkat yang mendukung pemrosesan algoritma SSD dengan efisien.
- c. Algoritma Faster R-CNN : Digunakan untuk melakukan deteksi objek dan klasifikasi makanan secara akurat.
- d. Sensor gas : untuk mendeteksi gas etilen yang dihasilkan oleh makanan yang membusuk
- e. Display : Monitor untuk memberi tahu pengguna letak yang tepat dari makanan yang di deteksi dan memberikan peringatan kepada pengguna jika ada makanan yang busuk di dalam kulkas.
- f. Telegram API : sistem bisa menggunakan bot Telegram yang diatur untuk memberikan notifikasi tentang kondisi makanan yang mulai membusuk di kulkas
- g. Modul Wi-Fi atau Bluetooth : untuk menghubungkan sistem dengan aplikasi mobile melalui jaringan.

Untuk Solusi ketiga ini digunakan metode komputasi Computer Vision dan algoritma Faster R-CNN untuk mendeteksi dan mengidentifikasi makanan berdasarkan gambar yang diambil kamera. Dan IoT untuk pemantauan sensor gas secara real-time untuk mengidentifikasi makanan busuk dan mengirimkan peringatan ke web yang dirancang.

1.2.3 Analisis Usulan Solusi

Analisis *House of Quality* untuk kebutuhan dan *Konstrain* yang dibutuhkan pada Solusi yang diajukan dapat dilihat pada Tabel 1.1.

Tabel 1.1 *House of Quality*

		↑	↑	↑	↑	↑			
		<i>Accurate data processing</i>	<i>Categorizing food</i>	<i>Monitor food conditions</i>	<i>Computational method</i>	<i>Reminder</i>	Solusi 1	Solusi 2	Solusi 3
<i>Low Power Consumption</i>	2	△	△	△	△	△	⊙	○	△
<i>Low cost</i>	5	-	△	△	△	-	○	△	△
<i>Real-time</i>	5	⊙	⊙	⊙	⊙	⊙	⊙	○	△
<i>Compatibility</i>	2	-	△	△	○	△	⊙	○	△
<i>Aesthetic Design</i>	3	-	○	○	○	△	⊙	○	○
Alat Selesai dalam 6 Bulan	5	△	△	△	⊙	⊙	○	○	○
<i>Imprtant Rating</i>	32	48	48	48	57	57	90	56	38
<i>Percentage Rating</i>		12.90%	19.35%	19.35%	25.40%	22.98%	48.91%	30.43%	20.65%
							Total		
Solusi 1		○	⊙	⊙	⊙	○	4.572		
Solusi 2		○	⊙	⊙	○	○	4.168		
Solusi 3		○	⊙	⊙	△	○	3.764		

Keterangan :

⊙ : Hubungan erat (5 poin)

△ : Hubungan kurang (1 poin)

○ : Hubungan normal (3 poin)

- : tidak ada hubungan

Berikut perhitungan menggunakan *House of Quality* ketiga Solusi terhadap fitur utama :

$$\text{Solusi 1 : } [(3 \times 12.90\%) + (5 \times 19.35\%) + (5 \times 19.35\%) + (5 \times 25.40\%) + (3 \times 22.98\%)] \\ = 4.28$$

$$\text{Solusi 2 : } [(3 \times 12.90\%) + (5 \times 19.35\%) + (5 \times 19.35\%) + (3 \times 25.40\%) + (3 \times 22.98\%)] \\ = 3.77$$

$$\text{Solusi 3 : } [(3 \times 12.90\%) + (5 \times 19.35\%) + (5 \times 19.35\%) + (1 \times 25.40\%) + (3 \times 22.98\%)] \\ = 3.26$$

Analisa yang dilakukan dengan metode *House of Quality*, dengan cara membandingkan fitur dasar dan fitur tambahan dari sistem yang akan dirancang dengan ke tiga Solusi yang ditawarkan. Terdapat 4 solusi dasar dan 5 solusi tambahan.

Hubungan fitur tambahan dengan fitur dasar yang saya buat, yang pertama yaitu *Low Power Consumption* memiliki hubungan yang bertimbal balik dengan semua fitur dasar. Yang mana semua fitur dasar yang saya buat ingin semuanya tinggi, jika *Accurate data processing*, *Categorizing food capability*, *Monitor food conditions capability*, *Computational method capability*, dan *Reminder* akurat maka dibutuhkan konsumsi daya yang tinggi juga.

Untuk fitur tambahan *low cost* tidak berhubungan dengan *Accurate data processing* dan *Reminder*, tapi sangat berhubungan erat dengan *Categorizing food*, *Monitor food conditions capability*, dan *Computational method capability*.

Untuk fitur *Real-time* sangat berhubungan erat dengan semua fitur dasar. Karna semakin baik kerja pengolahan data, sensor, komputasi dan remindernya maka sangat menentukan sistem yang dirancang akan real time.

Untuk fitur *Computability* tidak ada hubungannya dengan *Accurate data processing*, berhubungan berkebalikan dengan *Categorizing food capability*, *Monitor food conditions capability*, dan *reminder* karena semakin bagus kemampuan alat untuk mengorganisir makanan dan memonitoring kondisi

makanan dan *reminder* yang cepat maka akan semakin sulit untuk mengimplementasikan dan mengintegrasikannya ke sistem yang sudah ada.

Untuk fitur *Aesthetic Design* tidak ada hubungannya dengan *Accurate data processing* dan *Computational method capability*. Tapi cukup berhubungan dengan *Categorizing food capability*, *Monitor food conditions capability* karena jumlah dan bentuk komponen seperti sensor dan kamera mempengaruhi bentuk alatnya.

Untuk fitur alat selesai dalam waktu 6 bulan sangat berhubungan erat dengan metode komputasi dan reminder, semakin cepat metode komputasi dan remindernya maka alat akan semakin cepat selesai. Dan berhubungan timbal balik dengan *Accurate data processing*, *Categorizing food capability*, *Monitor food conditions capability*, karena semakin rumit data dan pengolahannya maka akan semakin lama proses pembuatan sistem.

Ketiga Solusi yang ditawarkan memenuhi setiap fitur dasar yang diinginkan, karna solusi yang dirancang harus dapat mengorganisir makanan dengan baik dan memantau kondisi makanan serta memberikan peringatan kepada pengguna terkait kondisi makanan yang terdeteksi tidak baik atau busuk.

Untuk ketiga solusi yang ditawarkan pada dasarnya memiliki fungsi yang sama, yaitu sama-sama menggunakan Machine Learning, dan IoT. Hanya saja yang membedakannya yaitu metode komputasinya dan pendekatan yang digunakan untuk setiap solusi. Untuk solusi 1 menggunakan algoritma YOLO dan aplikasi Android untuk mendeteksi dan memantau makanan di dalam kulkas. Solusi 1 menggunakan Machine Learning melalui YOLO untuk identifikasi objek secara real-time, sedangkan Solusi 2 dan Solusi 3 menggunakan sensor gas untuk memantau kondisi makanan dan Fuzzy Logic untuk mendeteksi pembusukan makanan. Solusi 2 dan Solusi 3 menggunakan algoritma SSD dan Faster R-CNN, yang keduanya memanfaatkan sensor gas dan Fuzzy Logic untuk mendeteksi gas pembusukan yang dihasilkan oleh makanan. Kedua solusi ini memerlukan komponen sensor untuk mendeteksi gas seperti metana, etilen, amonia, dan lainnya yang digunakan untuk menganalisis kondisi makanan.



Perbedaan utama antara Solusi 1 dan Solusi 2/3 adalah penggunaan sensor gas dalam Solusi 2 dan Solusi 3, sedangkan Solusi 1 lebih mengutamakan deteksi visual menggunakan YOLO dan estimasi masa simpan makanan dengan aplikasi Android.

Dari analisis House of Quality (HoQ) yang dilakukan, Solusi 1 yang menggunakan algoritma YOLO memiliki poin metode komputasi yang lebih tinggi dibandingkan dengan SSD dan Faster R-CNN. YOLO lebih efisien dalam hal kecepatan pemrosesan dan real-time object detection, yang sangat cocok untuk sistem yang mengutamakan konsumsi daya rendah dan pemrosesan gambar secara cepat. Sedangkan Faster R-CNN dan SSD memerlukan lebih banyak daya dan tahap pemrosesan yang lebih rumit.

Dengan perhitungan HoQ terhadap fitur tambahan Solusi 1 menunjukkan keunggulan karena lebih efisien, lebih mudah diimplementasikan, dan dapat dijalankan pada perangkat keras standar dengan konsumsi daya rendah. Solusi 2 dan Solusi 3 membutuhkan lebih banyak sumber daya dan lebih rumit dalam implementasinya, meskipun menggunakan sensor gas yang lebih akurat untuk mendeteksi pembusukan makanan.

Karena perbedaan hubungan ketiga solusi dengan fitur dasar tidak mencapai 1, dibutuhkan perhitungan hubungan ketiga solusi dengan fitur tambahan. Berikut perhitungan menggunakan HoQ ketiga Solusi terhadap fitur tambahan :

1. Solusi 1

$$[(5 \times 9.09\%) + (3 \times 22.73\%) + (5 \times 22.73\%) + (5 \times 9.09\%) + (5 \times 13.64\%) + (3 \times 22.73\%)] = 3.26$$

2. Solusi 2

$$[(3 \times 9.09\%) + (1 \times 22.73\%) + (3 \times 22.73\%) + (3 \times 9.09\%) + (3 \times 13.64\%) + (3 \times 22.73\%)] = 2.11$$

3. Solusi 3

$$[(1 \times 9.09\%) + (1 \times 22.73\%) + (1 \times 22.73\%) + (1 \times 9.09\%) + (3 \times 13.64\%) + (3 \times 22.73\%)] = 1.43$$

Dari hasil perhitungan menggunakan HoQ di atas dapat dilihat bahwa hubungan Solusi dengan fitur tambahan juga diunggulkan oleh Solusi 1. Yang mana untuk algoritma YOLO pada solusi 1 lebih efisien dan mudah diimplementasikan.

Algoritma YOLO dapat dijalankan pada perangkat keras dengan prosesor standar atau daya rendah sehingga cocok diimplementasikan pada alat hemat energi. Sedangkan Faster R-CNN dan SSD membutuhkan konsumsi daya yang lebih tinggi. Selain itu karena kecepatan pemrosesannya yang sangat tinggi Algoritma YOLO dapat memproses gambar secara *real-time*. Dan karena strukturnya yang lebih sederhana dibandingkan algoritma Faster R-CNN dan SSD mempermudah proses implementasi dan pemeliharaan sistem. Dan untuk solusi 1 lebih hemat dibandingkan solusi 2 dan 3, karena tidak menggunakan sensor.

1.2.4 Solusi yang Dipilih

Setelah melakukan analisis menggunakan *House of Quality*, didapatkan hasil bahwa Solusi pertama memiliki nilai tertinggi dibandingkan dengan solusi lainnya. Solusi pertama ini memberikan keseimbangan optimal antara akurasi, kecepatan pemrosesan, dan kemudahan implementasi. Algoritma YOLO memungkinkan deteksi objek secara *real-time* dengan konsumsi daya yang rendah, dan ideal untuk penggunaan pada kulkas yang sangat penting dalam konteks alat di kulkas. Selain itu, solusi ini memenuhi kebutuhan utama baik dalam fitur dasar maupun fitur tambahan yang diinginkan, tanpa menambah kompleksitas berlebih. Dengan ini diharapkan solusi 1 dapat menjawab semua permasalahan yang ada.

