

Pemanfaatan Raspberry Pi untuk Sistem Penghitung Mobil Otomatis pada Kampus UKDW

Kristian Adi Nugraha^{#1}, Laurentius Kuncoro Probo Saputra^{#2}

[#]Program Studi Teknik Informatika, Universitas Kristen Duta Wacana
Jl. Dr. Wahidin Sudirohusodo 5-25, Yogyakarta 55224

¹adinugraha@ti.ukdw.ac.id

²kuncoro@staff.ukdw.ac.id

Abstract — The number of vehicle keep increasing over time, causing some problems such as traffic jam and requirement of extra parking space. Universitas Kristen Duta Wacana (UKDW) is an university that has similar problem, lack of parking space at certain hours. That won't be a problem if campus activities such as course, student activities, etc. can spread evenly at all hours, not focused at one time only. But determining the accurate schedule is a difficult task, because it requires someone who observes all vehicles entering and exiting campus area all over time. This research propose a solution to create automatic system that can count all vehicles (car) that entering and exiting campus area, then the data can be used as a consideration in determining the scheduling at the next time, the lack of parking space area can be avoided. The system was built with internet of things technology using Raspberry Pi and camera as the main component with main server at the backend to store the data. That system is also using background substraction algorithm for counting the number of vehicles that entering and exiting campus area. The accuracy of the system at counting vehicles is 70%.

Keywords— computer vision, background substraction, motion object detection, internet of things, Raspberry Pi

I. PENDAHULUAN

Pertumbuhan jumlah kendaraan bermotor pribadi di Indonesia mengalami peningkatan yang cukup signifikan. Berdasarkan data Badan Pusat Statistik, jumlah kendaraan bermotor pribadi setiap tahun meningkat sekitar 11% baik untuk sepeda motor maupun mobil [1]. Hal tersebut dipicu karena kurangnya jumlah armada maupun cakupan wilayah dari kendaraan umum yang tersedia, sehingga masyarakat lebih banyak memilih untuk menggunakan kendaraan pribadi karena alasan keamanan dan fleksibilitas. Peningkatan jumlah kendaraan bermotor pribadi menimbulkan permasalahan baru, khususnya mengenai kepadatan *traffic* lalu lintas dan kapasitas lahan parkir. Banyak pihak yang terkena dampak dari peningkatan jumlah kendaraan pribadi, khususnya pihak-pihak yang memiliki lahan parkir di dalamnya seperti pusat perbelanjaan (*mall*), kompleks perkantoran, dan kampus.

Universitas Kristen Duta Wacana (UKDW) merupakan salah satu tempat yang mengalami permasalahan mengenai lahan parkir. Pada jam tertentu, jumlah kendaraan yang datang melebihi kapasitas lahan parkir yang ada membuat sebagian besar pengunjung kesulitan untuk mendapatkan tempat parkir. Hal ini dapat menghambat jalannya kegiatan para pengunjung (mahasiswa, karyawan, dan tamu), khususnya dari sisi efisiensi waktu dalam mencari tempat parkir.

Salah satu solusi yang dapat ditempuh untuk mengatasi masalah tersebut adalah dengan menambah kapasitas lahan parkir, namun seringkali hal ini tidak memungkinkan untuk dilakukan karena terbatasnya sisa lahan yang dapat dimanfaatkan untuk membangun lahan parkir. Solusi lain adalah dengan memecah kepadatan aktivitas yang terdapat di kampus secara merata agar tidak bertumpuk di satu titik waktu saja. Namun hal ini terkendala oleh sulitnya menentukan alokasi waktu yang tepat, karena membaca pola aktivitas kendaraan yang masuk dan keluar kampus tidak mudah untuk dilakukan secara konvensional. Dibutuhkan bantuan seseorang yang selalu siaga untuk mengamati kendaraan yang masuk dan keluar kampus selama satu hari penuh.

Berdasarkan pemaparan tersebut, penulis melakukan penelitian untuk membaca pola aktivitas kendaraan yang masuk dan keluar di area Kampus UKDW. Penelitian dilakukan dengan cara membangun sebuah sistem yang dapat membaca aktivitas kendaraan yang masuk dan keluar kampus secara otomatis tanpa memerlukan bantuan tenaga manusia. Penelitian yang dilakukan dengan menggunakan teknologi *Internet of Things (IoT)*, yaitu teknologi yang memanfaatkan perangkat seperti kamera atau sensor yang terhubung dengan jaringan internet dan komputer berukuran kecil sehingga mudah dimodifikasi dan diimplementasikan. Sistem yang akan dibangun oleh penulis memanfaatkan perangkat kamera yang tersambung dengan komputer kecil Raspberry Pi. Kamera ini berfungsi untuk menangkap gambar kendaraan yang masuk dan mengirimkannya ke *server*. *Server* tersebut bertugas untuk mengolah data yang masuk, kemudian menyajikan hasilnya dalam bentuk informasi kepada pengguna. Penulis hanya memilih

kendaraan berjenis mobil sebagai obyek penelitian, hal ini dikarenakan kendaraan jenis lain seperti sepeda atau sepeda motor tidak memiliki masalah dalam mencari tempat parkir karena area parkir selalu tersedia untuk kendaraan jenis ini meskipun pada jam padat sekalipun. Sedangkan kendaraan berjenis mobil selalu mengalami kesulitan untuk mendapatkan tempat parkir pada waktu-waktu tertentu, terutama pada saat jam padat. Berdasarkan data yang didapatkan oleh perangkat Raspberry Pi dan dikirimkan pada server, maka pada server akan didapatkan pola aktivitas jumlah mobil yang masuk dan keluar setiap hari. Harapan penulis dengan adanya sistem tersebut, maka informasi yang diberikan dapat menjadi bahan analisa dan pertimbangan dalam menentukan jadwal aktivitas dan kegiatan yang dilaksanakan di kampus UKDW supaya dapat tersebar secara merata sehingga tidak ada pemilik kendaraan, khususnya mobil, yang mengalami kesulitan untuk mencari tempat parkir karena area parkir yang penuh.

II. TINJAUAN PUSTAKA

Informasi merupakan sesuatu yang sangat penting, karena dengan informasi seseorang dapat menentukan apa yang harus dikerjakan atau untuk mendukung keputusan (*Decision Support System*). Infrastruktur dan layanan internet di era saat ini sangat mendukung penyediaan informasi yang cepat dan tepat. Berbagai penelitian telah banyak dilakukan khususnya untuk penyediaan informasi terkait kepadatan atau *traffic* sebuah obyek yang mengimplementasikan sistem pemantauan atau *surveillance* system. Obyek bisa berupa orang, kendaraan, barang, atau lainnya.

Masalah parkir saat ini dirasa sangat penting untuk diatasi di kota metropolitan ini. Kepadatan kendaraan yang terjadi di lahan parkir sangat menguras tenaga dan waktu dalam mencari tempat parkir. Banyak penelitian telah dilakukan dengan memanfaatkan teknologi *Internet of Things*. Kemudahan yang diberikan akan sangat membantu seseorang dalam mengetahui informasi parkir. Muftah F. dan Mikael F. [2] mengembangkan smart parking yang bisa memberikan informasi detail lahan parkir yang bisa didapatkan melalui perangkat *mobile* berbasis CCTV. Seseorang yang akan menggunakan lahan parkir dapat mengakses melalui aplikasi yang terhubung ke parking server untuk mendapatkan informasi detail denah lokasi lahan parkir yang tersedia.

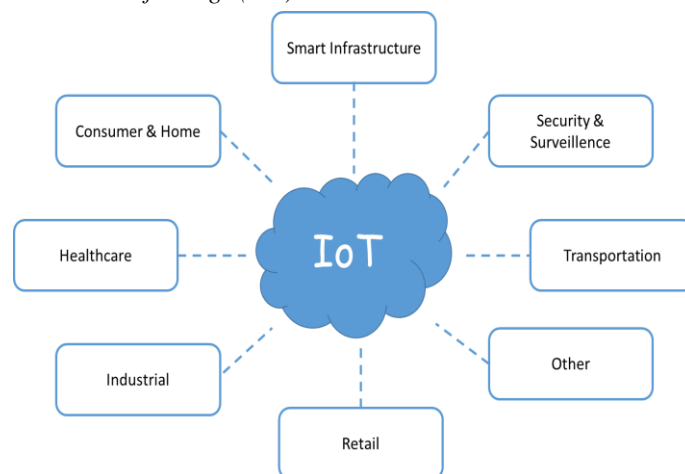
Berkembangnya penerapan *Internet of Things* membuat pengembangan sistem informasi semakin baik. Muhammad Alam dkk. [3] mengembangkan sistem *smart parking* dengan menempatkan sensor *magnetic* di setiap slot parkir, sehingga apabila ada kendaraan yang terparkir pada slot tersebut dapat diketahui dan dicatat di server. Semua data kendaraan yang terparkir dan lokasi slot yang digunakan dicatat oleh server sehingga pihak pengelola parkir dapat mengetahui data statistik penggunaan lahan parkir. Pengguna pun dapat mengakses aplikasi untuk mengetahui

lahan parkir seperti halnya pada penelitian Muftah F. dan Mikael F. [2].

Permasalahan ketersediaan tempat parkir menjadi permasalahan banyak orang di kota metropolitan di mana semua orang memiliki waktu yang terbatas hanya untuk sekedar mencari tempat parkir. Hal ini menimbulkan banyak pemikiran dan pengembangan sebuah sistem *smart parking*. Khaoula H. [4] telah melakukan tinjauan pustaka terkait pengembangan *smart parking* yang telah dikembangkan dari tahun 2012 hingga 2016. Secara umum, pengembangan *smart parking* menggunakan sensor-sensor seperti *magnetic* sensor, *RFID* [5], kamera [6], *IR* sensor untuk mendeteksi slot parkir yang digunakan. Data dari sensor dikirimkan ke server untuk dicatat. Server *smart parking* memberikan fasilitas kepada pengguna dengan cara menampilkan jumlah slot parkir yang terisi, data statistik penggunaan lahan parkir, dapat melakukan reservasi slot parkir. Penggunaan aplikasi mobile pun turut dikembangkan untuk mengakses informasi tersebut.

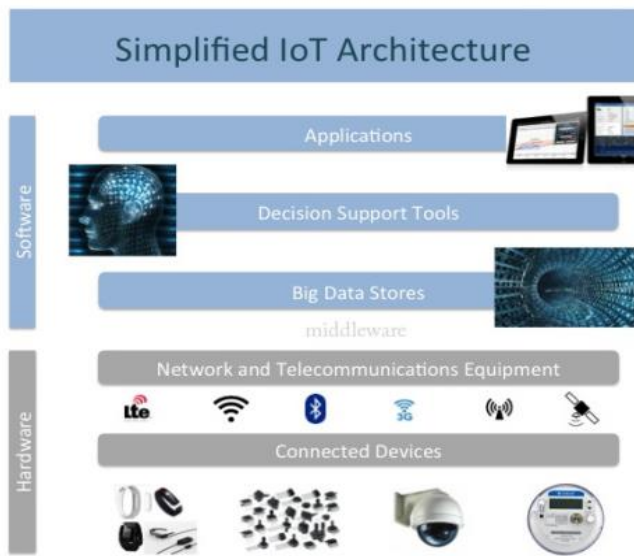
Berdasarkan kajian yang telah dilakukan dari beberapa penelitian terkait, konsep *Internet of Things* memiliki peranan yang penting dalam menyiapkan informasi terkait lahan parkir. Penelitian yang akan dilakukan kali ini lebih menambahkan fasilitas yang dapat diterapkan oleh sistem *smart parking*. Fitur yang akan ditambahkan ialah penyediaan informasi *traffic* pada suatu lahan parkir secara langsung. Informasi *traffic* yang akan diberikan ialah jumlah slot secara umum yang tersedia, laju kepadatan kendaraan, dan estimasi waktu kepadatan lahan parkir itu penuh. Sensor yang akan digunakan berupa sensor kamera yang akan menghitung jumlah kendaraan mobil yang masuk atau keluar. Sensor tersebut diletakkan di dua tempat yang berbeda yaitu pintu masuk dan pintu keluar kendaraan. Studi kasus akan dilakukan pada lahan parkir Universitas Kristen Duta Wacana Yogyakarta.

A. Internet of Things (IoT)



Gambar 1. Potensi teknologi IoT

Merupakan sebuah paradigma baru dalam bidang teknologi komunikasi dan informasi di mana seluruh obyek saling terhubung yang dibantu juga dengan jaringan internet [7]. Pengembangan dan penelitian terkait bidang ini sedang banyak dilakukan karena didukung dengan banyaknya pembuat produk yang sangat membantu dalam menerapkan teknologi ini. Beberapa bidang aplikasi yang dapat dikembangkan dengan paradigma *IoT* diperlihatkan seperti Gambar 1. Saat ini berbagai macam pengembangan teknologi meliputi sensor, aktuator, *embedded system*, *cloud computing*, dan yang paling penting adalah biaya yang sangat murah dan ukuran yang kecil dari sebuah perangkat *wireless* sehingga semuanya komponen itu tadi dapat saling terhubung. Pemanfaatannya pun beragam, seperti dikehidupan sehari-hari, akademisi, maupun di industri.



Gambar 2. Arsitektur *IoT* [8]

Secara arsitektur sebuah paradigma *IoT* dapat diperlihatkan seperti pada Gambar 2. Berdasarkan Gambar 2 terlihat bahwa arsitektur *IoT* dibagi menjadi 2 bagian yaitu *hardware* dan *software*. Bagian *hardware* bertugas menyediakan data dari spesifik obyek dan bagian *software* bertugas untuk mengolah data lebih lanjut untuk menghasilkan informasi yang berguna atau menyediakan layanan bagi pengguna.

B. Motion Object Detection

Sensor kamera memiliki peranan yang sangat penting dalam sebuah *surveillance system*. Obyek bergerak dalam sebuah citra yang dihasilkan dari sensor kamera dianalisis dari kumpulan beberapa citra. Sebuah citra dapat mengandung banyak sekali obyek bergerak. *Motion object detection* merupakan algoritma yang digunakan untuk mendeteksi obyek bergerak pada sebuah citra. Banyak cara dikembangkan untuk dapat memberikan hasil yang baik

dalam proses mendeteksi obyek bergerak. Salah satu metode yang dapat digunakan adalah *background subtraction*. Pendeteksian obyek bergerak di sebuah citra diasumsikan bahwa background pada citra tetap sedangkan obyek pada citra yang bergerak. Selanjutnya, obyek pada citra bergerak pada domain waktu, sehingga lebih dari satu citra harus digunakan untuk melihat perpindahan obyek tersebut.

Beberapa penelitian menggunakan dasar *background subtraction* untuk melakukan *motion detection* sebuah obyek, seperti penelitian yang dilakukan oleh Qinghua Ji dan Suping Yu [9] dengan penelitian berjudul "Motion Detection Based on Adaptive Mixture Gaussian Model and Four-Frame Subtraction" serta Yu Xiaoyang dkk. [10] dengan penelitian berjudul "A Novel Motion Object Detection Method Based on Improved Frame Difference and Improved Gaussian Mixture Model".

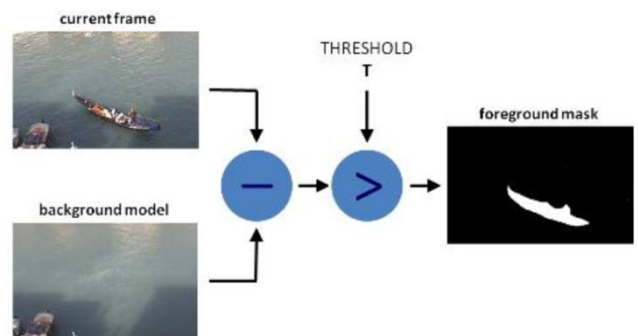
Motion object detection digunakan untuk mendeteksi obyek yang bergerak pada area pintu masuk dan keluar. Sebuah obyek bergerak yang akan ditangkap ialah kendaraan mobil yang akan masuk atau keluar. Sehingga dari proses tersebut, sistem dapat menghitung berapa kendaraan yang masuk dan keluar.

C. Background Substraction

Background subtraction adalah proses untuk mendeteksi gerakan atau perbedaan yang signifikan dalam *frame* video, yang dibandingkan dengan referensi, dan untuk menghapus semua komponen non-signifikan (*background*) [11]. *Background subtraction* diterapkan di berbagai kondisi, seperti *surveillance system* (fokus pada segmen benda yang bergerak saja). Rumus perhitungan *background subtraction* ditunjukkan pada rumus 1.

$$D(x, y) = \begin{cases} 1 & \text{if } F(x, y) - B(x, y) > T \\ 0 & \text{if } F(x, y) - B(x, y) \leq T \end{cases} \tag{1}$$

Dimana:
B = *background*
F = *frame* saat ini
T = *threshold* / nilai ambang batas



Gambar 3. Ilustrasi *background subtraction* [12]

Background Substraction menggunakan metode perbandingan gambar saat ini dan gambar background untuk mendeteksi adanya benda bergerak. Setelah gambar *background* $B(x, y)$ diperoleh, maka dilakukan perhitungan dengan cara gambar *background* $B(x, y)$ dikurangi *frame* $F(x, y)$ yang akan menghasilkan selisih piksel seperti yang ditunjukkan pada gambar 3. Jika selisih piksel lebih besar dari ambang batas (*threshold*) yang ditetapkan yaitu T , maka piksel tersebut dideteksi sebagai obyek, jika nilai piksel tersebut berada di bawah nilai T maka piksel tersebut tetap merupakan sebuah *background* [13]. Piksel-piksel yang dianggap sebagai obyek akan ditandai dengan warna putih, sedangkan piksel *background* akan ditandai dengan warna hitam.

Kelemahan dari *background subtraction* adalah adanya kondisi eksternal yang sulit atau tidak bisa dikendalikan, misalnya kondisi pencahayaan yang selalu berubah yang disebabkan oleh pergerakan matahari (pagi, siang, malam). Oleh karena itu terdapat sedikit modifikasi pada metode ini untuk selalu memperbarui *background* secara berkala, metode ini disebut *running average background subtraction* [14]. Rumus perhitungan untuk memperbarui *background* ditunjukkan pada rumus 2.

$$B_{i+1}(x, y) = \alpha \cdot F_i(x, y) + (1 - \alpha) \cdot B_i(x, y) \quad (2)$$

Dimana :

- B_{i+1} = *background* terbaru
- α = *learning rate*
- F_i = *frame* saat ini
- B_i = *background* sebelumnya

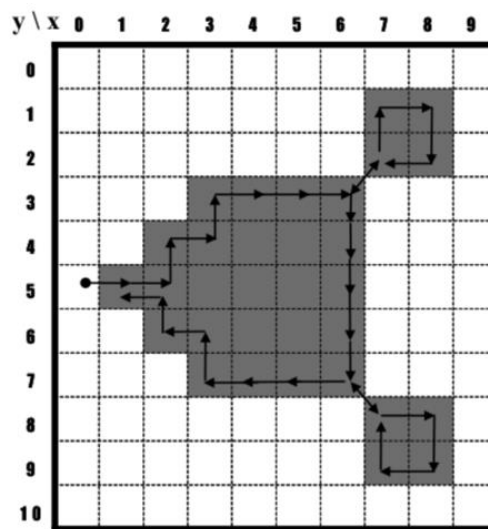
Pada rumus 2 terdapat nilai α yang merupakan nilai laju belajar (*learning rate*), yaitu nilai yang digunakan untuk mengatur perubahan besar kecilnya perubahan pada *background* yang digunakan untuk proses *background subtraction*. Hal ini diperlukan untuk melakukan penyesuaian supaya proses pembaruan *background* dapat dilakukan secara optimal.

D. Contour Tracing

Contour tracing merupakan proses pemisahan antara obyek dengan *background* berdasarkan warna dari obyek untuk mendapatkan bentuk (*contour*) dari obyek tersebut [15]. Cara kerja *contour tracing* ditunjukkan pada gambar 4, yaitu dengan mengambil titik piksel awal dari obyek yang dikehendaki, kemudian melakukan penelusuran titik dengan cara mengelilingi piksel-piksel yang membungkus obyek tersebut hingga kembali pada titik awal piksel.

Apabila *contour* sebuah obyek pada gambar telah berhasil didapatkan, maka dapat dilakukan beberapa perhitungan untuk mencari keliling dan luas dari obyek tersebut. Perhitungan luas dari obyek diperlukan untuk menentukan apakah obyek yang ditangkap oleh kamera merupakan mobil atau bukan. Jika obyek tersebut

merupakan sebuah mobil, maka akan diproses lebih lanjut. Namun jika obyek tersebut bukan mobil (misalnya sepeda motor atau pejalan kaki), maka obyek tersebut dapat diabaikan dan tidak akan diolah lebih lanjut oleh sistem.



Gambar 4. Ilustrasi contour tracing [15]

E. Object Tracking

Object tracking merupakan salah satu konsep pengembangan dari pengolahan citra, di mana metode yang diterapkan mencoba untuk melakukan *tracking* atau pengamatan terhadap sebuah obyek yang diinginkan. Konsep ini banyak diterapkan pada teknologi CCTV untuk tujuan pengamanan (*surveillance*), di mana kamera dapat mendeteksi apabila merekam obyek yang bergerak di dalamnya [16].

Pada penelitian ini, konsep *object tracking* digunakan untuk mengamati pergerakan mobil yang masuk atau keluar di setiap *frame*-nya. Metode ini akan membandingkan obyek yang terdeteksi pada *frame* pertama dan *frame* kedua untuk mengetahui apakah kedua obyek tersebut merupakan obyek yang sama atau tidak, karena terdapat kemungkinan bahwa obyek yang ada pada *frame* pertama dan kedua merupakan obyek mobil yang berbeda. Proses *object tracking* tersebut dilakukan dengan mencari posisi titik pusat (*centroid*) dari obyek mobil yang terdeteksi di masing-masing *frame*, kemudian membandingkan kedua *centroid* tersebut dengan menggunakan rumus *Euclidean Distance* yang ditunjukkan pada rumus 3.

$$d = \sqrt{(F1_x - F2_x)^2 + (F1_y - F2_y)^2} \quad (3)$$

Dimana :

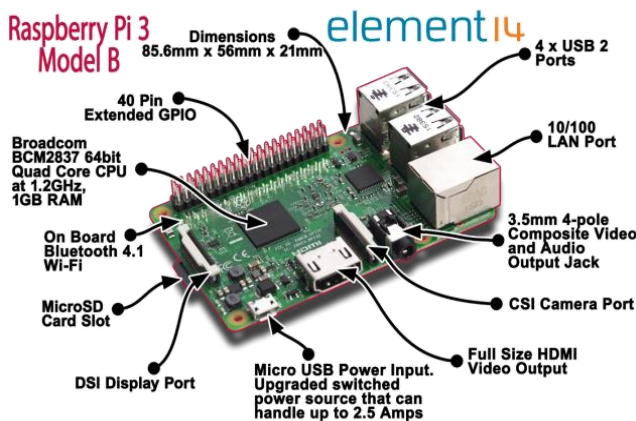
- $F1_x$ = koordinat x *centroid* di *frame* 1
- $F2_x$ = koordinat x *centroid* di *frame* 2

$F1_y$ = koordinat y *centroid* di frame 1
 $F2_y$ = koordinat y *centroid* di frame 2

Apabila nilai yang dihasilkan melalui rumus 3 berada di bawah nilai ambang (*threshold*) yang telah ditentukan, maka *centroid* pertama dan *centroid* kedua merupakan *centroid* dari obyek yang sama. Sedangkan apabila nilainya berada di atas *threshold*, maka *centroid* pertama dan *centroid* kedua merupakan *centroid* dari dua obyek yang berbeda.

F. Raspberry Pi

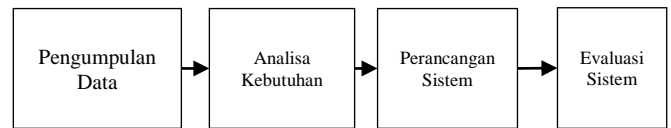
Raspberry Pi merupakan sebuah perangkat komputer mini sebesar sebuah kartu nama yang dikembangkan oleh Raspberry Pi Foundation. Raspberry Pi dapat berfungsi selayaknya sebuah komputer dengan ukurannya yang kecil. Raspberry Pi memiliki banyak model dan ukuran. Semua bergantung pada kebutuhan sistem yang akan dibangun. Salah satu versi terbarunya adalah Raspberry Pi 3 yang sudah tersedia chip komunikasi *wireless* dan *bluetooth*. Raspberry Pi sangat cocok dalam digunakan dalam pengembangan sistem berbasis *IoT*. Kelebihan lain yang bisa dilakukan oleh perangkat Raspberry Pi ialah terbukanya GPIO yang bisa digunakan untuk dihubungkan dengan perangkat / sensor lain. Gambar 5 memperlihatkan sebuah perangkat Raspberry Pi 3 beserta deskripsi komponen di dalamnya.



Gambar 5. Raspberry Pi 3 [17]

III. METODOLOGI PENELITIAN

Alur dari penelitian yang akan dilakukan oleh penulis digambarkan pada gambar 6 yang terdiri dari proses pengumpulan data, analisa kebutuhan, perancangan sistem, dan evaluasi sistem.



Gambar 6. Blok diagram

A. Pengumpulan Data

Langkah awal yang dilakukan oleh penulis adalah dengan melakukan pengumpulan data. Data yang dikumpulkan adalah data berupa foto untuk mengetahui berbagai jenis bentuk kendaraan yang masuk dan keluar area Kampus UKDW. Data tersebut dibutuhkan untuk melakukan analisa kebutuhan agar sistem yang dibangun dapat memenuhi target yang diharapkan.

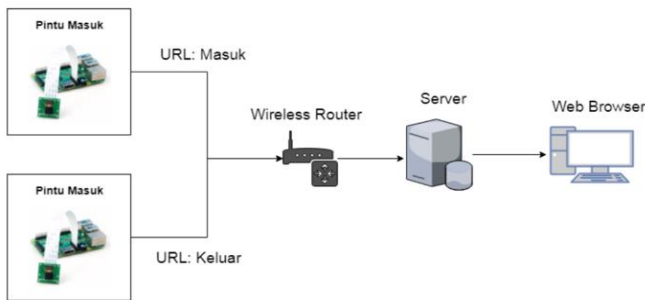
B. Analisa Kebutuhan

Analisa Kebutuhan dilakukan setelah penulis mendapatkan data yang cukup. Data-data yang diperoleh dari proses pengumpulan data dibutuhkan oleh penulis untuk melakukan analisa mengenai apa yang dibutuhkan oleh sistem yang akan dibangun. Salah satu komponen yang akan dianalisis adalah ukuran obyek yang nantinya akan terekam oleh kamera. Terdapat banyak jenis obyek yang dapat terekam kamera seperti pejalan kaki, sepeda motor, mobil, truk, bus kecil, dan bus besar. Masing-masing obyek tersebut memiliki ukuran yang bervariasi, hal ini tentunya akan membedakan besar kecilnya lahan yang dibutuhkan untuk mengakomodasi obyek tersebut. Dengan demikian, sistem hendaknya memiliki kemampuan untuk mengategorikan obyek-obyek tersebut berdasarkan ukurannya, sehingga data yang diolah menjadi lebih presisi.

C. Perancangan Sistem

Perancangan sistem adalah proses membangun rancangan dari sistem dan mengimplementasikannya berdasarkan analisa kebutuhan yang telah dilakukan sebelumnya. Sistem yang akan dibangun menggunakan arsitektur client-server, di mana kamera pengawas berperan sebagai client yang akan mengirimkan data server apabila terdapat kendaraan yang masuk atau keluar, kemudian server akan mengolah data tersebut dan mengubahnya menjadi informasi yang siap diakses oleh pengguna. Diagram sistem diperlihatkan seperti pada Gambar 7. Sistem pengawas akan menggunakan komputer berukuran kecil dilengkapi dengan kamera yang sanggup menerjemahkan gambar kendaraan yang didapat, untuk menentukan apakah kendaraan tersebut termasuk kategori kecil (sepeda, sepeda motor), sedang (mobil), besar (bus, truk), atau bukan kendaraan (orang). Sistem pada perangkat Raspberry Pi dibangun dengan menggunakan

bahasa C++, sedangkan sistem pada server dibangun dengan menggunakan bahasa Python.



Gambar 7. Arsitektur sistem

D. Evaluasi

Evaluasi mula-mula dilakukan dengan cara melakukan pengujian sistem. Pengujian dilakukan pada tiga kondisi waktu yang berbeda, yaitu pagi, siang, dan sore. Hal ini dikarenakan setiap kondisi waktu memiliki karakteristik yang berbeda karena pengaruh perbedaan letak posisi matahari, sehingga posisi bayangan pada obyek-obyek yang terekam oleh kamera juga akan berbeda. Pengujian dilakukan pada dua titik yang berbeda, yaitu pada pintu masuk dan pintu keluar.

Setelah pengujian dilakukan, maka langkah berikutnya adalah melakukan validasi terhadap akurasi jumlah mobil yang masuk atau keluar. Sistem hanya akan menghitung obyek mobil yang masuk dan keluar area kampus UKDW. Obyek selain mobil seperti sepeda motor, bus, dan pejalan kaki akan diabaikan oleh sistem. Rumus perhitungan persentase akurasi ditunjukkan pada rumus 4. Apabila perhitungan menghasilkan nilai minus, maka akan dianggap sebagai 0.

$$A = \frac{T - F}{T_a} \quad (4)$$

Dimana :

T = jumlah mobil menurut sistem

F = jumlah kendaraan non-mobil yang dianggap sebagai mobil oleh sistem

T_a = jumlah mobil sesungguhnya

IV. HASIL DAN PEMBAHASAN

Sistem yang dibangun diimplementasikan dengan cara memasang perangkat Raspberry Pi beserta kamera pada pintu masuk dan pintu keluar kampus UKDW. Kedua perangkat tersebut dipasang pada gedung lantai 3 yang terletak di depan pintu masuk, bersebelahan dengan perangkat CCTV yang digunakan untuk mengamati pintu masuk seperti yang ditunjukkan pada gambar 8.



Gambar 8. Instalasi perangkat Raspberry Pi dan kamera



(a)



(b)

Gambar 9. a) Pintu masuk Kampus UKDW; b) Pintu keluar Kampus UKDW

Hasil foto dari kamera pada pintu masuk dan keluar ditunjukkan pada gambar 9. Pada foto tersebut terlihat bahwa kamera dapat menangkap area pintu masuk dan pintu keluar area Kampus UKDW mulai dari sebuah kendaraan masuk dari jalan raya hingga memasuki area Kampus UKDW, atau keluar dari area Kampus UKDW hingga masuk ke jalan raya.

Berdasarkan foto dari kamera yang didapat, penulis menentukan area pada foto tersebut sebagai acuan untuk menentukan jumlah kendaraan yang masuk. Area tersebut

ditandai dengan warna hijau seperti ditunjukkan pada gambar 10. Apabila pada kamera pintu masuk merekam sebuah mobil memasuki area tersebut kemudian meninggalkannya, maka sistem akan menambah jumlah mobil yang masuk ke dalam kampus sebanyak satu buah. Demikian juga pada pintu keluar, apabila kamera merekam adanya sebuah mobil yang memasuki area tersebut dan meninggalkannya, maka sistem akan mengurangi jumlah mobil yang berada di dalam kampus sebanyak satu buah.



(a)



(b)

Gambar 10. a) Pintu masuk Kampus UKDW; b) Pintu keluar Kampus UKDW

Ketika sistem pertama kali dijalankan, mula-mula perlu dilakukan inisialisasi *background* terlebih dahulu. Untuk berikutnya, *background* tersebut akan diperbarui secara terus menerus menggunakan rumus 2. *Background* yang menjadi inisialisasi merupakan *background* yang tidak terdapat satu obyek pun seperti yang ditunjukkan pada gambar 11. Selanjutnya, *background* tersebut akan terus diperbarui menyesuaikan dengan kondisi lingkungan yang berubah karena faktor pencahayaan dan arah bayangan yang selalu berubah.



Gambar 11. *Background* inisialisasi pada pintu masuk

Pada tahap berikutnya, sistem akan melakukan perhitungan *background subtraction* pada setiap *frame* yang direkam oleh perangkat kamera dari Raspberry Pi untuk mendeteksi adanya mobil yang masuk atau keluar area Kampus UKDW. Kamera tersebut merekam video secara *real time* dengan pengaturan *frame* sebanyak 10 *frame* per detik dengan resolusi 640 x 480 piksel. Nilai 10 *frame* per detik didapatkan setelah melalui beberapa kali percobaan dengan pertimbangan keterbatasan performa (*processor* dan *memory*) pada perangkat Raspberry Pi, di mana perangkat ini tidak mampu untuk melakukan komputasi dengan kompleksitas yang cukup tinggi dalam waktu yang relatif singkat. Berbeda dengan komputer berbasis *desktop* atau *server* yang mampu melakukan komputasi dengan tingkat kompleksitas yang tinggi serta mengolah data berjumlah besar dalam waktu yang singkat. Selain itu, nilai 10 *frame* per detik masih sanggup digunakan untuk mengamati proses pergerakan obyek mobil yang bergerak memasuki atau keluar area Kampus UKDW. Hal ini didasarkan pada pengukuran jarak antara jalan raya hingga area dalam kampus UKDW memiliki jarak sekitar 8 meter, sementara kecepatan mobil tidak akan mencapai 8 meter per detik (28.8 km/jam). Karena berdasarkan pengamatan, mobil yang masuk ke dalam area Kampus UKDW akan mengurangi kecepatan menjadi sekitar 5 s/d 15 km/jam, di mana nilai ini berada jauh di bawah nilai batas yaitu 28.8 km/jam. Dengan demikian, proses mobil masuk atau keluar area Kampus UKDW memerlukan waktu kurang lebih 2 hingga 6 detik, sehingga pengaturan *frame* kamera dengan nilai 10 *frame* per detik masih sanggup digunakan untuk mengamati proses perpindahan obyek mobil tersebut.



Gambar 12. Contoh gambar hasil *background subtraction*

Proses *background subtraction* digunakan untuk mendeteksi seluruh obyek yang tertangkap oleh kamera dari Raspberry Pi seperti ditunjukkan pada gambar 12. Obyek-obyek yang terekam oleh sistem akan dianalisa terlebih dahulu apakah obyek tersebut merupakan mobil atau bukan. Penentuan obyek tersebut dilakukan dengan cara melakukan proses *countour tracing*, yaitu proses pencarian bentuk obyek, untuk mengetahui jumlah dan bentuk dari obyek yang dihasilkan oleh proses *background subtraction*, kemudian melakukan perhitungan dari luas untuk masing-masing obyek tersebut. Apabila luas dari obyek tersebut berada di antara nilai batas bawah dan nilai batas atas yang telah ditentukan, maka obyek tersebut merupakan mobil. Namun jika obyek tersebut memiliki nilai luas di luar nilai batas bawah dan batas atas, maka obyek tersebut dianggap bukan merupakan sebuah mobil. Berdasarkan hasil pengamatan yang dilakukan oleh penulis, maka ditetapkan nilai batas bawah dan batas atas untuk luas mobil pada pintu masuk dan pintu keluar sesuai dengan yang ditunjukkan pada tabel I. Nilai tersebut didapatkan dari beberapa kali percobaan dengan cara menghitung luas dari seluruh obyek mobil yang masuk dan keluar, kemudian menentukan nilai batas bawah dan batas atas berdasarkan luas minimal dan luas maksimal dari mobil yang didapatkan pada masing-masing pintu masuk dan pintu keluar.

TABEL I
TABEL BATAS BAWAH DAN BATAS ATAS

Lokasi	Luas Batas Bawah (piksel)	Luas Batas Atas (piksel)
Pintu Masuk	460 piksel	1400 piksel
Pintu Keluar	410 piksel	1350 piksel



(a)



(b)

Gambar 13. a) Hasil deteksi pada pintu masuk; b) Hasil deteksi pada pintu keluar

Hasil pendeteksian mobil yang masuk dan keluar area Kampus UKDW menggunakan kombinasi metode *background subtraction* dan perhitungan luas dengan menggunakan *contour tracing* ditunjukkan pada gambar 13. Berikutnya, penulis melakukan pengujian akurasi dengan cara mengambil *sample* dari seluruh data yang diperoleh. Masing-masing *sample* berbentuk potongan video dengan durasi 5 menit dengan jumlah total sebanyak 20 video. Sample tersebut diambil pada 3 macam kondisi yaitu pagi, siang, dan sore di masing-masing pintu masuk dan keluar. Nilai hasil pengujian tersebut ditunjukkan pada tabel II hingga tabel VII. Pada tabel tersebut, kolom T merupakan jumlah mobil dan dianggap mobil oleh sistem, kolom F merupakan jumlah obyek non-mobil yang dianggap mobil oleh sistem, kolom T_a merupakan jumlah mobil sebenarnya, dan kolom A merupakan nilai akurasi yang didapatkan berdasarkan rumus 4.

TABEL II
TABEL PENGUJIAN PINTU MASUK PAGI HARI

No	T	F	T _a	A
1	10	0	10	100%
2	2	0	2	100%
3	10	0	10	100%
4	8	2	9	67%
5	9	0	9	100%
6	12	2	12	83%
7	17	4	19	68%
8	10	4	11	55%
9	13	1	13	92%
10	5	0	5	100%
11	10	0	10	100%
12	8	0	8	100%
13	12	0	12	100%
14	3	0	3	100%
15	4	0	4	100%
16	6	0	6	100%
17	4	0	4	100%
18	3	1	3	67%
19	5	1	5	80%
20	2	0	2	100%
Rata-rata				91%

TABEL III
TABEL PENGUJIAN PINTU MASUK SIANG HARI

No	T	F	T _a	A
1	1	0	1	100%
2	2	0	2	100%
3	1	1	1	0%
4	3	3	3	0%
5	1	0	1	100%
6	3	0	3	100%
7	1	1	1	0%
8	3	0	3	100%
9	1	0	1	100%
10	2	0	2	100%
11	1	0	1	100%
12	3	3	6	0%
13	4	0	4	100%
14	5	0	5	100%
15	2	2	2	0%
16	5	2	5	60%
17	4	0	4	100%
18	3	5	3	0%
19	6	2	6	67%
20	2	0	2	100%
Rata-rata				66%

TABEL IV
TABEL PENGUJIAN PINTU MASUK SORE HARI

No	T	F	T _a	A
1	1	0	1	100%
2	4	0	4	100%
3	3	0	3	100%
4	1	0	1	100%

5	2	0	2	100%
6	2	0	2	100%
7	3	0	3	100%
8	2	0	2	100%
9	2	0	2	100%
10	2	0	2	100%
11	2	1	3	33%
12	2	0	2	100%
13	5	0	5	100%
14	2	0	2	100%
15	3	0	3	100%
16	2	0	2	100%
17	3	0	3	100%
18	1	0	1	0%
19	5	2	5	60%
20	2	0	2	100%
Rata-rata				90%

TABEL V
TABEL PENGUJIAN PINTU KELUAR PAGI HARI

No	T	F	T _a	A
1	2	1	3	33%
2	2	1	2	50%
3	2	1	2	50%
4	2	2	4	0%
5	4	0	4	100%
6	2	1	2	50%
7	1	0	1	100%
8	2	1	3	33%
9	3	0	3	100%
10	3	0	3	100%
11	3	0	3	100%
12	2	1	2	50%
13	5	1	5	80%
14	3	3	3	0%
15	2	0	2	100%
16	5	3	5	40%
17	1	0	1	100%
18	3	0	3	100%
19	2	0	2	100%
20	3	0	3	100%
Rata-rata				69%

TABEL VI
TABEL PENGUJIAN PINTU KELUAR SIANG HARI

No	T	F	T _a	A
1	0	4	4	0%
2	0	3	3	0%
3	4	3	7	14%
4	0	1	1	0%
5	3	0	3	100%
6	6	1	6	83%
7	1	3	2	0%
8	2	2	4	0%
9	4	2	5	40%
10	5	3	6	33%
11	2	2	2	0%

12	3	0	3	100%
13	2	2	2	0%
14	4	2	4	50%
15	4	3	4	25%
16	6	5	8	13%
17	2	2	3	0%
18	5	1	6	67%
19	3	1	3	67%
20	1	1	2	0%
Rata-rata				30%

TABEL VII
TABEL PENGUJIAN PINTU KELUAR SORE HARI

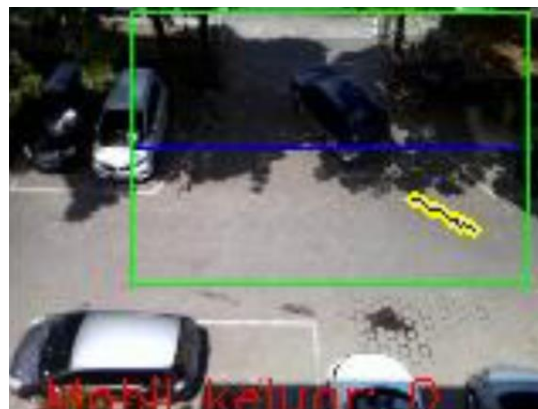
No	T	F	T _a	A
1	4	0	4	100%
2	7	1	7	86%
3	4	0	4	100%
4	8	1	9	78%
5	7	0	7	100%
6	8	3	8	63%
7	1	1	2	0%
8	4	1	5	60%
9	3	1	4	50%
10	6	0	6	100%
11	13	3	13	77%
12	6	0	6	100%
13	4	2	5	40%
14	3	0	3	100%
15	5	1	5	80%
16	6	0	6	100%
17	8	1	8	88%
18	1	0	1	100%
19	6	1	6	83%
20	2	2	2	0%
Rata-rata				75%

TABEL VIII
TABEL RANGKUMAN PENGUJIAN

Lokasi	Pagi	Siang	Sore	Rata-rata
Pintu masuk	91%	66%	90%	82%
Pintu keluar	69%	30%	75%	58%
Rata-rata	80%	48%	83%	70%

Berdasarkan data rangkuman hasil pengujian yang ditunjukkan pada tabel VIII, dapat disimpulkan bahwa rata-rata persentase akurasi secara keseluruhan adalah 70%. Dari tabel VIII terlihat bahwa nilai rata-rata akurasi persentase pada pintu masuk lebih baik dibandingkan dengan pintu keluar, yaitu sebesar 82%. Hal ini disebabkan pada pintu keluar terdapat adanya obyek-obyek yang cukup mengganggu proses pendeteksian mobil, seperti pohon-pohon tinggi yang menghasilkan bayangan cukup besar. Kondisi semakin buruk ketika matahari berada di posisi tepat di atas lokasi Kampus UKDW (siang hari), di mana bayangan pohon yang dihasilkan cukup tebal sehingga

mobil sulit terdeteksi seperti yang ditunjukkan pada gambar 14(a).



(a)



(b)

Gambar 14. a) Pintu keluar pada siang hari; b) Hasil *background subtraction* pada gambar 14a.

Pada gambar 14b terlihat bahwa proses *background subtraction* tidak berhasil mendapatkan obyek mobil dengan baik, hal ini dikarenakan posisi mobil terletak pada bayangan pohon yang mengakibatkan mobil tersebut sulit terdeteksi. kondisi ini banyak terjadi pada siang hari karena posisi matahari tepat berada di atas lokasi, sehingga cahaya yang dipancarkan sangat terang dan mengakibatkan bayangan yang terbentuk sangat tebal. Hal ini ditunjukkan pada tabel VIII untuk kondisi siang hari, nilai rata-rata persentase akurasi berada pada angka paling rendah dibandingkan dengan kondisi lain, yaitu sebesar 48%.

V. KESIMPULAN

Berdasarkan hasil pengujian yang telah dilakukan oleh penulis, rata-rata tingkat akurasi sistem secara keseluruhan adalah sebesar 70%. Terdapat beberapa kendala dalam proses pendeteksian mobil masuk dan keluar, yaitu berkaitan dengan pencahayaan dan bayangan yang terbentuk oleh matahari khususnya pada siang hari. Permasalahan tersebut dapat menjadi masukan pada penelitian sejenis berikutnya, yaitu menggunakan metode atau proses

tambahan yang dapat mengatasi permasalahan pendeteksian mobil yang terletak di dalam bayangan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Fakultas Teknologi Informasi Universitas Kristen Duta Wacana yang telah memberikan bantuan berupa dana penelitian, sehingga penelitian ini dapat terlaksana dengan baik. Penulis juga mengucapkan terima kasih secara khusus kepada saudara Yudha Satria Kurniawan sebagai asisten penelitian yang membantu penulis dalam mengerjakan penelitian ini mulai dari tahap awal hingga publikasi.

DAFTAR PUSTAKA

- [1] "Badan Pusat Statistik," 2014. [Online]. Available: <https://www.bps.go.id/linkTabelStatis/view/id/1413>. [Accessed 2017].
- [2] M. Fraifer and M. Fernström, "Designing a Smart Car Parking System (PoC) Prototype Utilizing CCTV Nodes: A vision of an IoT parking system via UCD process," in *Internet of Things (WF-IoT), 2016 IEEE 3rd World Forum*, Reston, VA, USA, 2016.
- [3] M. Alam, B. Fernandes, J. Almeida, J. Ferreira and J. Fonseca, "Integration of smart parking in distributed ITS architecture," in *Open Source Systems & Technologies (ICOSST), 2016 International Conference*, Lahore, Pakistan, 2017.
- [4] K. Hassoune, W. Dachry, F. Moutaouakkil and H. Medromi, "Smart parking systems: A survey," in *Intelligent Systems: Theories and Applications (SITA)*, Mohammedia, Morocco, 2016.
- [5] A. Roy, J. Siddiquee, A. Datta, P. Poddar, G. Ganguly and A. Bhattacharjee, "Smart traffic & parking management using IoT," in *Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, Vancouver, BC, Canada, 2016.
- [6] S. A. El-Seoud, H. El-Sofany and I. Taj-Eddine, "owards the development of smart parking system using mobile and web technologies," in *Interactive Mobile Communication, Technologies and Learning (IMCL)*, San Diego, CA, USA, 2016.
- [7] L. Atzori, A. Iera and G. Morabito, "The Internet of Things: A survey," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 54, no. 15, pp. 2787-2805, October 2010.
- [8] IoTCo, "Understanding The IoT Landscape: The Building Blocks," 2014. [Online]. Available: <https://iotco.org/2014/11/09/understanding-the-iot-landscape-the-building-blocks/>. [Accessed 2017].
- [9] Q. Ji and S. Yu, "Motion Object Detection Based on Adaptive Mixture Gaussian Model and Four-Frame Subtraction," in *Computational and Information Sciences (ICCIS)*, Shiyang, China, 2013.
- [10] Y. Xiaoyang, Y. Yang, Y. Shuchun, S. Yang, Y. Huimin and L. Xifeng, "A novel motion object detection method based on improved frame difference and improved Gaussian mixture model," in *Measurement, Information and Control (ICMIC)*, Harbin, China, 2013.
- [11] V. D. R. and N. L. Kumar, "Object Tracking Using Background Subtraction Algorithm," *International Journal of Engineering Research and General Science*, vol. 3, no. 1, pp. 237-243, January 2015.
- [12] OpenCV, "How to Use Background Subtraction Methods," 2016. [Online]. Available: https://docs.opencv.org/trunk/Background_Subtraction_Tutorial_Scheme.png. [Accessed 2017].
- [13] L. Zhang and Y. Liang, "Motion Human Detection Based on Background Subtraction," in *Education Technology and Computer Science (ETCS)*, Wuhan, China, 2010.
- [14] F. Y. A. Rahman, A. Hussain, W. M. D. W. Zaki, H. B. Zaman and N. M. Tahir, "Enhancement of background subtraction techniques using a second derivative in gradient direction filter," *Journal of Electrical and Computer Engineering*, vol. 2013, January 2013.
- [15] J. Seo, S. Chae, J. Shim, D. Kim, C. Cheong and T.-D. Han, "Fast Contour-Tracing Algorithm Based on a Pixel-Following Method for Image Sensors," *Sensors*, vol. 16, no. 3, 2016.
- [16] S. Ojha and S. Sakhare, "Image processing techniques for object tracking in video surveillance- A survey," in *Pervasive Computing (ICPC)*, Pune, India, 2015.
- [17] J. Lendino, "Raspberry Pi 3 launches with faster 64-bit processor and Wi-Fi for the same \$35," 2016. [Online]. Available: <https://www.extremetech.com/computing/223769-raspberry-pi-3-launches-with-faster-64-bit-processor-and-wi-fi-for-the-same-35>. [Accessed 2017].