

PERANCANGAN DAN IMPLEMENTASI GARASI PRIBADI DENGAN PINTU OTOMATIS BERDASARKAN PENGENALAN PLAT KENDARAAN BERBASIS PENGOLAHAN CITRA DIGITAL

(DESIGN AND BUILD OF AUTOMATIC PRIVATE GARAGE DOOR WITH VEHICLE PLATE RECOGNITION BASED ON DIGITAL IMAGE PROCESSING)

¹Dimas Mustaqim, ²Fiky Yosef Suratman, Dr.Ing., ³Ekki Kurniawan.Ir., MT.

^{1,2,3}Prodi S1 Teknik Elektro, Fakultas Teknik, Universitas Telkom
Jl. Telekomunikasi, Dayeuh Kolot Bandung 40257 Indonesia

¹ mustaqimdims@gmail.com, ² @telkomuniversity.ac.id, ³@gmail.co.id

ABSTRAK

Kendaraan sebagai alat yang digunakan manusia untuk berpindah dari satu tempat ke tempat lainnya memiliki beraneka ragam bentuk dan ukuran. Dalam penggunaannya, suatu kendaraan memerlukan tempat penyimpanan atau garasi ketika kendaraan tersebut setelah atau sebelum digunakan oleh pemiliknya. Dengan alasan keamanan dan kenyamanan, maka dibuatlah suatu garasi otomatis dimana pintu dari garasi otomatis ini hanya dapat terbuka untuk kendaraan yang diizinkan saja.

Setiap kendaraan menggunakan plat yang berisi nomor polisi sebagai identitas yang akan membedakan kendaraan yang satu dengan yang lainnya. Suatu plat kendaraan dapat dijadikan parameter untuk pembuatan suatu garasi otomatis. Maka pada tugas akhir ini, penulis merancang suatu sistem pintu otomatis pada garasi pribadi berdasarkan pembacaan nomor plat mobil. Pembacaan plat nomor mobil dilakukan oleh kamera sebagai sensor. Plat nomor mobil akan ditangkap oleh kamera berupa *image* dan terjadi proses *image processing* untuk mendapatkan karakter plat nomor kendaraan. Nomor plat mobil yang didapatkan akan dibandingkan melalui proses *template matching* dengan data referensi yang sudah ditentukan. Jika nomor plat mobil yang didapatkan sesuai dengan data referensi, maka mikrokontroler Arduino Uno akan mengatur kondisi kontaktor. Mikrokontroler akan menentukan arah dan besar sudut putaran dari motor DC. Ketika motor DC berputar maka pintu garasi akan terbuka sehingga mobil bisa masuk ke dalam garasi.

Harapan dan tujuan yang ingin dicapai dalam pembuatan garasi pribadi otomatis berdasarkan pengenalan plat nomor kendaraan ini adalah mampu memberikan kenyamanan bagi pengguna khususnya dari sisi keamanan penyimpanan kendaraan.

Kata Kunci : *image processing, character recognition, mikrokontroler, arduino uno, template matching.*

ABSTRACT

Vehicle as a tool that humans use to move from one place to another has variegated shapes and sizes. In its application, requires a vehicle storage area or garage when the vehicle is after or before they are used by their owners. For security reasons and the convenience, then made an automatic garage door from the garage where automated can only be open to vehicles permitted only.

Every vehicle using a plate containing the number of the police as an identity that would distinguish one vehicle to another. A plate of the vehicle can be used as a parameter for making an automatic garage. So in this final project, the authors designed a system of automatic doors in a private garage by car number plate reading. Car license plate readings performed by the camera as a sensor. Car license plate will be captured by the camera in the form of an image and a process of image processing to get the license plate characters. Car license plate number obtained will be compared through the template matching process with a predetermined reference data. If the license plate obtained in accordance with the reference data, the microcontroller Arduino Uno will set the conditions contactor. Microcontroller will determine the direction and magnitude of the rotation angle of the DC motor. When the DC motor rotates the garage door opens so that the car can go into the garage.

Expectations and objectives to be achieved in the construction of private garage automatically based on vehicle number plate recognition to provide the convenience for users, especially on the security side of the vehicle garage.

Keyword : image processing, character recognition, mikrokontroler, arduino uno, template matching

1. PENDAHULUAN

1.1 Latar Belakang

Seiring perkembangan teknologi yang semakin kompleks dan pertumbuhan manusia yang terus bertambah di muka bumi ini, maka manusia akan terus menciptakan sesuatu yang baru untuk menyesuaikan atau mempermudah kondisi kehidupan pada zaman dimana manusia itu hidup. Salah satu contohnya yaitu, manusia menciptakan alat transportasi sebagai alat untuk mereka berpindah dari satu tempat ke tempat lain dalam waktu sesingkat mungkin dengan penggunaan energi seminimal mungkin. Mobil sebagai salah satu jenis kendaraan memerlukan sebuah garasi untuk tempat penyimpanan ketika mobil setelah atau sebelum digunakan. Inilah yang menjadi dasar perancangan suatu sistem garasi otomatis berbasis mikrokontroler. Tingkat keamanan dan kenyamanan sangat diperlukan dalam pembuatan suatu garasi otomatis. Dari sisi otomatis, sistem garasi ini hanya akan membuka pintunya ketika mobil yang akan masuk menggunakan plat nomor yang diizinkan. Sedangkan dari sisi keamanan, mobil yang menggunakan plat nomor yang berbeda atau tidak diizinkan tidak akan bisa masuk ke dalam garasi.

Pada tugas akhir ini, kemampuan sistem ANPR akan dipelemntasikan pada sebuah *prototype* garasi otomatis yang dikendalikan dengan mikrokontroller Arduino Uno dan sebuah *Personal Computer* dengan *software* Matlab sebagai pengolahan citra digitalnya. Pengidentifikasian karakter ANPR dilakukan setelah kamera aktif ketika sensor ultrasonik mengidentifikasi ada objek di depan garasi. Karakter yang diidentifikasi bersifat spesifik sehingga tidak akan terjadi pemalsuan nomor plat kendaraan sehingga pintu garasi hanya akan terbuka ketika mobil yang akan masuk hanya menggunakan plat nomor yang dikehendaki.

Sebelumnya ada beberapa metode pembelajaran yang telah diperkenalkan untuk menjawab persoalan diatas, khususnya di Telkom University yaitu pada tugas akhir dengan judul "Pengenalan Plat Kendaraan Berbasis Pengolahan Citra Digital dan Jaringan Syaraf Tiruan *Self-Organizing Maps (SOM)*" oleh Yuli Sun Hariyani.^[1] Namun pada penelitian sebelumnya belum sampai pada pengembangan yang lebih baik untuk diimplementasikan dalam kehidupan sehari-hari sehingga pada tugas akhir ini penulis akan membuat sistem garasi atau penyimpanan mobil berdasarkan pembacaan dan pengenalan plat nomor kendaraan. Penulis akan menggunakan metode ANPR (*Automatic Number Plate Recognition*) sebagai algoritma untuk menentukan keakuratan sistem identifikasi plat nomor kendaraan yang diharapkan dapat meningkatkan keamanan dan kenyamanan pemilik kendaraan.

1.2 Tujuan

Tujuan dari penulisan dari tugas akhir ini mengenali dan mencari karakter pada suatu plat nomor kendaraan menggunakan kamera sebagai sensor, merancang dan membangun suatu garasi otomatis berdasarkan pengenalan plat nomor kendaraan serta membangun *interface* sistem pengolahan citra digital pada PC dengan menggunakan mikrokontroler Arduino Uno.

2. DASAR TEORI

2.1 Pengolahan Citra Digital

Pengolahan citra digital dilakukan untuk memproses suatu citra sehingga menghasilkan citra yang sesuai dengan keinginan kita atau kualitasnya menjadi lebih baik. Secara umum dan sederhana, citra dapat didefinisikan sebagai representasi visual dari suatu objek. Sebuah citra analog tidak bisa diproses secara langsung oleh computer. Agar sebuah citra analog bisa diproses di dalam komputer, maka harus diubah terlebih dahulu menjadi citra digital.

2.2 Citra Analog

Citra atau *image* merupakan suatu gambar pada bidang 2 dimensi. Citra analog adalah citra yang bersifat kontinu. Citra analog tidak dapat direpresentasikan dan diproses langsung di dalam computer. Agar citra analog dapat diproses di dalam komputer maka harus dilakukan proses konversi terlebih dahulu yaitu proses konversi dari citra analog ke citra digital. Contoh citra analog seperti, lukisan, pemandangan alam, gambar pada monitor televisi dsb.

2.3 Citra Digital

Citra digital adalah citra kontinu yang koordinat dan intensitas cahayanya diubah ke dalam bentuk diskrit. Citra digital merupakan representasi sebuah citra dua dimensi sebagai kumpulan nilai digital yang disebut *pixel* atau *picture element*. Piksel merupakan satuan titik terkecil dari suatu citra digital yang mengandung nilai terkuantisasi dan mewakili *brightness* dari sebuah warna pada titik tertentu. Suatu objek akan memantulkan kembali sebagian cahaya ketika ada sumber cahaya yang menerangi. Kemudian alat pengindera optik akan menangkap pantulan cahaya dari objek tersebut. Contoh alat pengindera optik misalnya, mata manusia, kamera dsb. Bayangan objek dipantulkan dan terekam sesuai intensitas pantulan cahaya. Ketika pantulan cahaya dari suatu objek direkam oleh alat optik, maka akan dihasilkan citra digital. Citra digital mempunyai kontinuitas intensitas cahaya sesuai resolusi alat perekam.

Sebuah citra digital memiliki suatu fungsi kontinu dalam dua dimensi dari intensitas cahaya (x,y) . Dimana x dan y untuk merepresentasikan suatu sumbu koordinat dengan f pada setiap titik x dan y yang menyatakan tingkat kecerahan (*brightness/gray level*) pada suatu titik. Nilai $f(x,y)$ harus sebanding dengan

energi yang dipancarkan oleh sumber cahaya, oleh sebab itu besar intensitas $f(x,y)$ tidak boleh nol dan harus berhingga, yaitu:

$$0 < f(x,y) < \infty \dots\dots\dots (2.1)$$

Fungsi $f(x,y)$ dapat dibagi menjadi dua bagian, yaitu:

- a. Jumlah cahaya yang berasal dari sumbernya disimbolkan oleh $i(x,y)$, nilainya antara 0 dan ∞ .
- b. Derajat kemampuan objek memantulkan cahaya $r(x,y)$, nilainya antara 0 dan 1.

Besar $f(x,y)$ merupakan hasil dari perkalian keduanya,

$$f(x,y) = i(x,y) \times r(x,y) \dots\dots\dots (2.2)$$

Nilai $i(x,y)$ ditentukan oleh sumber cahaya, sedangkan $r(x,y)$ ditentukan oleh karakteristik objek di dalam gambar. Nilai $r(x,y) = 0$ menyatakan penyerapan lokal, sedangkan nilai $r(x,y) = 1$ menyatakan pemantulan total. Jika permukaan mempunyai derajat pemantulan nol maka fungsi intensitas cahaya $f(x,y)$ juga nol. Sebaliknya, jika permukaan mempunyai derajat pemantulan 1 maka fungsi intensitas cahaya sama dengan i (iluminasi) yang diterima oleh permukaan tersebut. Derajat keabuan (*gray level*) adalah intensitas $f(x,y)$ di titik (x,y) , dimana derajat keabuan bergerak dari hitam ke putih. Sedangkan citranya disebut citra skala keabuan (*grayscale image*). Derajat keabuan atau *grayscale image* memiliki nilai rentang dari $I_{min} < f(x,y) < I_{max}$. Rentang nilai ini sering digeser menjadi selang $[0,L]$ dimana 0 menyatakan hitam, dan L menyatakan putih.



Gambar 2.1 Sumbu Koordinat Citra^[2]

Pada Gambar 2.1 mengilustrasikan citra digital sebagai fungsi dua variabel yaitu $f(x,y)$, dimana x dan y adalah letak titik kordinat dan nilai $f(x,y)$ menyatakan intensitas citra pada titik koordinat tersebut.

Citra digital yang berukuran $M \times N$ biasanya dinyatakan dalam bentuk matriks yang berukuran M baris dan N kolom, sebagai berikut:

$$F = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \vdots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix} \dots\dots\dots (2.3)$$

1.4 ANPR ((Automatic Number Plate Recognition)^[5]

Pengenalan plat nomor otomatis adalah metode pengawasan massa yang menggunakan pengenalan karakter pada gambar untuk membaca plat kendaraan. ANPR menggunakan *optical character recognition* (OCR) pada gambar yang diambil oleh kamera. Beberapa pengaturan plat menggunakan variasi ukuran font dan sistem penentuan posisi ANPR harus mampu mengatasi perbedaan-perbedaan tersebut secara efektif. Sistem yang lebih rumit dapat mengatasi varian internasional, meskipun banyak program secara individual disesuaikan dengan masing-masing Negara.

Kamera yang digunakan dapat mencakup penegakan ada jalan-aturan atau kamera televisi sirkuit tertutup, serta unit mobile, yang biasanya melekat pada kendaraan. Beberapa sistem menggunakan kamera inframerah untuk mengambil gambar yang lebih jelas dari plat kendaraan. Ada tujuh algoritma primer yang membutuhkan perangkat lunak untuk mengidentifikasi karakter pada plat kendaraan :

1. Lokalisasi posisi plat - bertanggung jawab untuk mencari dan mengisolasi plat pada gambar.
2. Orientasi Plat dan pengaturan ukuran - mengkompensasi orientasi plat dan menyesuaikan dimensi dengan ukuran yang dikehendaki.
3. Normalisasi - menyesuaikan kecerahan dan kontras gambar.
4. Segmentasi karakter - menemukan karakter individu di piring.
5. *Optical Character Recognition*.
6. Sintaksis/analisis geometris - memeriksa karakter dan posisi terhadap peraturan spesifik Negara.
7. Penyamaan rata-rata nilai karakter yang didapat dari beberapa *image* untuk mendapatkan hasil yang lebih tepat. Terutama karena setiap gambar yang *capture* mungkin terdapat cahaya yang terpantulkan, sehingga menjadi *blur* atau efek sementara lainnya.

Kompleksitas dari masing-masing subbagian ini program menentukan keakuratan sistem. Selama fase ketiga (normalisasi), beberapa sistem menggunakan teknik *edge detection* untuk meningkatkan perbedaan gambar antara huruf dan dukungan plat. Dan teknik *median filter* juga dapat digunakan untuk mengurangi *noise* visual pada gambar.

1.5 Matlab R2013a

Matlab adalah sebuah bahasa (pemrograman) dengan unjuk kerja tinggi (*high-performance*) untuk komputasi teknis yang mengintegrasikan komputasi, visualisasi, dan pemrograman di dalam lingkungan yang mudah penggunaannya dalam memecahkan persoalan dengan solusinya dinyatakan dengan notasi matematik. Nama matlab merupakan singkatan dari *Matrix Laboratory*. Penggunaan Matlab untuk:

1. Matematika dan komputasi
2. Pengembangan dan algoritma
3. Pemodelan, simulasi dan pembuatan *prototype*
4. Analisis data, eksplorasi dan visualisasi
5. Grafik untuk sains dan teknik
6. Pengembangan aplikasi, termasuk pembuatan antarmuka grafis untuk pengguna.

Matlab adalah sebuah sistem interaktif yang menggunakan elemen data dasarnya adalah *array* yang tidak membutuhkan dimensi. Fitur-fitur Matlab untuk penyelesaian spesifik disebut *toolbox*. *Toolbox* adalah koleksi komprehensif dari fungsi-fungsi matlab (M-Files) yang memperlebar lingkungan Matlab dalam menyelesaikan kelas-kelas tertentu dari permasalahan. Beberapa *toolbox* yang tersedia meliputi bidang: pengolahan sinyal, sistem kendali, jaringan saraf tiruan, logika *fuzzy*, *wavelet*, simulasi dan sebagainya.

1.6 Mikrokontroler Arduino Uno^[5]

Mikrokontroler adalah sebuah sistem fungsional lengkap pada *embeded system* yang terkandung dalam sebuah chip/IC. Mikrokontroler berbeda dari mikroprocessor, karena dalam sebuah mikrokontroler berisi komponen-komponen pendukung, yaitu: *processor*, *memory (RAM/ROM)*, dan I/O. Mikrokontroler merupakan perkembangan dari mikroprocessor yang merupakan suatu *Integrated Circuit (IC)* yang bekerja berdasarkan program dan dirancang secara khusus untuk aplikasi sistem kendali atau monitoring. Mikrokontroler merupakan komputer didalam chip yang digunakan untuk mengontrol peralatan elektronik, yang menekankan efisiensi dan efektifitas biaya. Secara harfiahnya bisa disebut "pengendali kecil" dimana sebuah sistem elektronik yang sebelumnya banyak memerlukan komponen-komponen pendukung seperti IC TTL dan CMOS dapat direduksi/diperkecil dan akhirnya terpusat serta dikendalikan oleh mikrokontroler ini. Dengan kata lain, mikrokontroler adalah versi mini atau mikro dari sebuah komputer karena mikrokontroler sudah mengandung beberapa periferal yang langsung bisa dimanfaatkan, misalnya port paralel, port serial, komparator, konversi digital ke analog (DAC), konversi analog ke digital dan sebagainya hanya menggunakan sistem minimum yang tidak rumit atau kompleks. Dalam sebuah chip mikrokontroler telah terintegrasi *memory*, *CPU* dan I/O baik serial maupun paralel. Sifat mikrokontroler yang mampu diprogram (*programmable*) menyebabkan mikrokontroler mempunyai kemampuan aplikasi yang sangat luas seperti digunakan dalam produk dan alat yang dikendalikan secara otomatis, seperti sistem kontrol mesin, remote controls, mesin kantor, peralatan rumah tangga, alat berat, dan mainan.

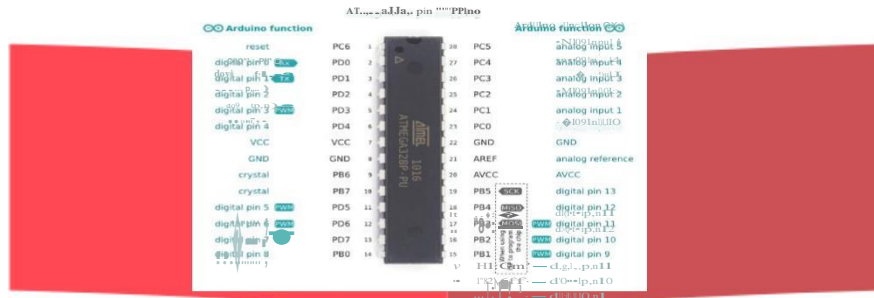
Mikrokontroler Arduino Uno adalah board mikrokontroler berbasis ATmega328 (datasheet). Memiliki 14 pin input dari output digital dimana 6 pin input tersebut dapat digunakan sebagai output PWM dan 6 pin input analog, 16 MHz osilator kristal, koneksi USB, jack power, ICSP header, dan tombol reset. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan Board Arduino Uno ke komputer dengan menggunakan kabel USB atau listrik dengan AC yang-ke adaptor-DC atau baterai untuk menjalankannya.

Arduino Uno berbeda dengan semua board sebelumnya dalam hal koneksi USB-to-serial yaitu menggunakan fitur Atmega8U2 yang diprogram sebagai konverter USB-to-serial berbeda dengan board sebelumnya yang menggunakan chip FTDI driver USB-to-serial. Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi serial melalui USB dan sebagai port virtual com untuk perangkat lunak pada komputer. Firmware 8U2 menggunakan driver USB standar COM, dan tidak ada driver eksternal yang diperlukan. Namun, pada Windows diperlukan, sebuah file inf. Perangkat lunak Arduino terdapat monitor serial yang memungkinkan digunakan memonitor data tekstual sederhana yang akan dikirim ke atau dari board Arduino. LED RX dan TX di papan tulis akan berkedip ketika data sedang dikirim melalui chip USB-to-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1). Sebuah SoftwareSerial library memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada board Uno's.

ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino termasuk perpustakaan Kawat untuk menyederhanakan penggunaan bus I2C, lihat dokumentasi untuk rincian. Untuk komunikasi SPI, menggunakan perpustakaan SPI. Spesifikasi umum pada mikrokontroler Arduino Uno R3:

1. Mikrokontroler ATmega328
2. Operasi dengan daya 5V Voltage
3. Input Tegangan (disarankan) 7-12V
4. Input Tegangan (batas) 6-20V
5. Digital I / O Pins 14 (dimana 6 memberikan output PWM)
6. Analog Input Pin 6
7. DC Lancar per I / O Pin 40 mA

- 8. Saat 3.3V Pin 50 mA DC
 - 9. Flash Memory 32 KB (ATmega328) yang 0,5 KB digunakan oleh bootloader
 - 10. SRAM 2 KB (ATmega328)
 - 11. EEPROM 1 KB (ATmega328)
 - 12. Clock Speed 16 MHz
- Konfigurasi pin dapat dilihat pada Gambar 2.2



Gambar 2.2 Konfigurasi Pin Arduino Uno R3

1.7 Sensor Ultrasonik (Fisal Nur, 2011)

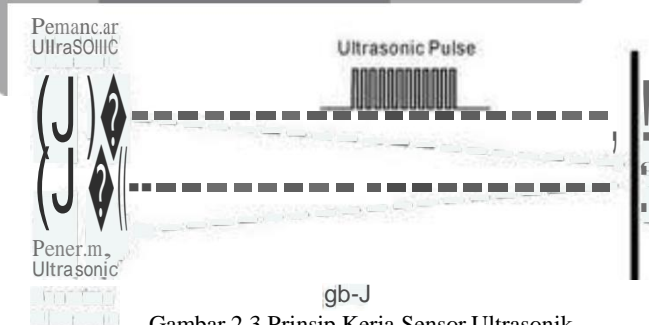
Sensor Ultrasonik adalah alat elektronika yang kemampuannya bisa mengubah dari energy listrik menjadi energy mekanik dalam bentuk gelombang suara ultrasonik. Sensor ini terdiri dari rangkaian pemancar Ultrasonik yang dinamakan transmitter dan penerima ultrasonik yang disebut receiver. Alat ini digunakan untuk mengukur gelombang ultrasonik. Gelombang ultrasonik adalah gelombang mekanik yang memiliki ciri-ciri longitudinal dan biasanya memiliki frekuensi di atas 20 Khz. Gelombang Ultrasonik dapat merambat melalui zat padat, cair maupun gas. Gelombang Ultrasonik adalah gelombang rambatan energi dan momentum mekanik sehingga merambat melalui ketiga element tersebut sebagai interaksi dengan molekul dan sifat enersia medium yang dilaluinya.

Sensor ini dapat mengukur jarak antara 2 cm sampai 300 cm. keluaran dari sensor ini berupa pulsa yang lebarnya merepresentasikan jarak. Lebar pulsanya bervariasi dari 115 uS sampai 18,5 mS. Sensor ultrasonik ping parallax terdiri dari sebuah chip pembangkit sinyal 40KHz, sebuah speaker ultrasonik dan sebuah mikropon ultrasonik. Speaker ultrasonik mengubah sinyal 40 KHz menjadi suara sementara mikropon ultrasonik berfungsi untuk mendeteksi pantulan suaranya. Prinsip kerja dari sensor ultrasonik adalah sebagai berikut:

1. Sinyal dipancarkan oleh pemancar ultrasonik. Sinyal tersebut berfrekuensi diatas 20 kHz, biasanya yang digunakan untuk mengukur jarak benda adalah 40kHz. Sinyal tersebut di bangkitkan oleh rangkaian pemancar ultrasonic.
2. Sinyal yang dipancarkan tersebut kemudian akan merambat sebagai sinyal / gelombang bunyi dengan kecepatan bunyi yang berkisar 340 m/s. Sinyal tersebut kemudian akan dipantulkan dan akan diterima kembali oleh bagian penerima Ultrasonik.
3. Setelah sinyal tersebut sampai di penerima ultrasonik, kemudian sinyal tersebut akan diproses untuk menghitung jaraknya. Jarak dihitung berdasarkan rumus persamaan:

$$S = 340 \cdot t / 2 \dots\dots\dots (2.7)$$

dimana S adalah jarak antara sensor ultrasonik dengan bidang pantul, dan t adalah selisih waktu antara pemancaran gelombang ultrasonik sampai diterima kembali oleh bagian penerima ultrasonik. Dapat dilihat pada Gambar 2.3.

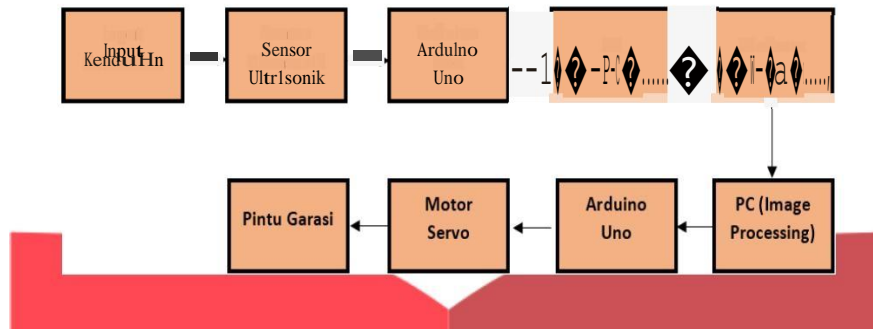


Gambar 2.3 Prinsip Kerja Sensor Ultrasonik

3. PERANCANGAN DAN IMPLEMENTASI SISTEM

3.1 Cara Kerja Sistem

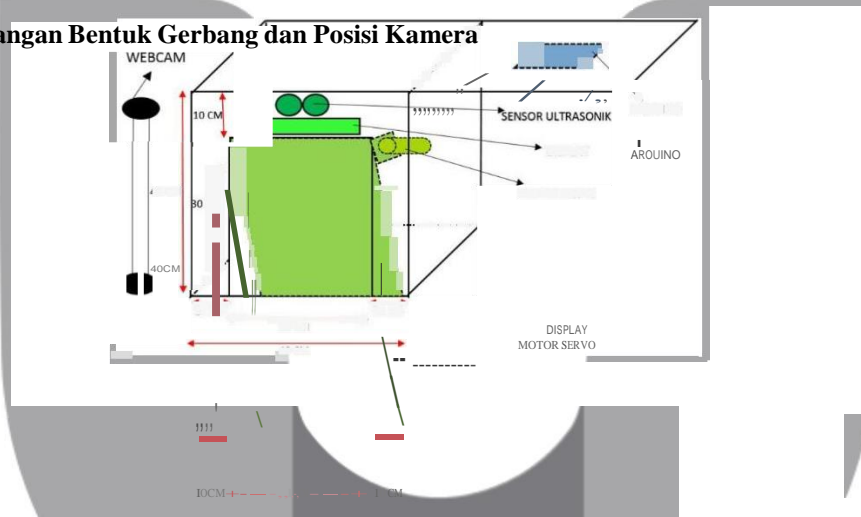
Secara umum sistem akan bekerja seperti blok diagram berikut ini:



Gambar 3.1 Proses Sistem Bekerja Secara Umum

Gambar 3.1 di atas merupakan proses yang akan dilakukan sistem secara umum. Input merupakan masukan yang akan diproses yaitu berupa kendaraan roda empat atau mobil yang memiliki plat nomor kendaraan yang akan diidentifikasi setiap karakter dari plat nomor kendaraan tersebut. Selanjutnya sensor ultrasonik akan mendeteksi keberadaan benda ± 100 cm di depannya lalu mikrokontroler Arduino akan mengolah input jarak dan mengeluarkan output sistem sebagai *trigger* atau untuk mengaktifkan *webcam* yang telah dihubungkan langsung melalui port pada PC. Melalui komunikasi UART TTL, antar muka perangkat dilakukan untuk mengirimkan data dari PC ke mikrokontroler Arduino Uno atau sebaliknya. PC atau *Personal computer* merupakan otak dari pengolahan citra yang akan dilakukan yaitu untuk mengidentifikasi setiap karakter plat nomor kendaraan yang telah diambil gambarnya oleh *Webcam*. Setelah proses pengidentifikasian selesai, kemudian sistem akan membandingkan data hasil proses pengidentifikasian dengan database yang sudah ada sebelumnya. Setelah data hasil proses identifikasi sesuai dengan database yang sudah ada, maka motor servo sebagai penggerak yang terhubung dengan pintu garasi akan membuka pintu garasi.

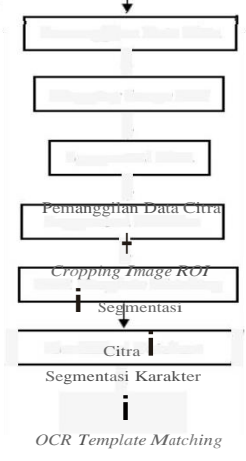
3.2 Perancangan Bentuk Gerbang dan Posisi Kamera



Gambar 3.2 Posisi Webcam dan Bentuk Garasi

Bentuk prototipe garasi yang dibuat sesuai dengan Gambar 3.2 dengan posisi *webcam* yang melakukan *capture image* berada pada ketinggian 40 cm diatas permukaan tanah dan diarahkan kepada *bumper* mobil diletakkan secara terpisah pada depan garasi. Sedangkan untuk sensor ultrasonik diletakkan di depan pintu garasi dengan ketinggian 40 cm untuk mengoptimalkan proses identifikasi ada tidaknya mobil di depan pintu garasi. Posisi mobil yang akan dilakukan proses *capture* ditentukan jaraknya yaitu sekitar ± 100 cm dari depan sensor ultrasonik.

3.3 Perancangan Software Sistem





Gambar 3.3 Diagram Alir Algoritma Pengolahan Citra Digital

Perancangan software pengolahan citra digital yang dilakukan menggunakan Matlab R2013a. Diagram alir pengolahan citra digital dapat dilihat pada Gambar 3.3 di atas. Dan berikut ini adalah keterangan dari diagram alir Gambar 3.3:

1. Proses inisialisasi komunikasi mikrokontroler Arduino dengan PC. Proses ini bertujuan untuk menyelaraskan port yang digunakan mikrokontroler dan webcam sehingga proses pengiriman data tidak mengalami interupsi data lain. Setelah inisialisasi dilakukan ultrasonik pada mikrokontroler akan terus membaca jarak didepannya secara langsung dikirimkan Matlab sebagai pengolah data. Ketika jarak < 100 cm maka data '1' akan diterima selanjutnya webcam akan melakukan proses penangkapan citra.
2. Hasil citra yang ditangkap akan disimpan sementara pada penyimpanan PC, selanjutnya diolah untuk identifikasi citra.
3. Proses pengolahan yang pertama adalah *cropping image*. *Cropping image* ini bertujuan untuk mendapatkan ROI (*Region of interest*), sehingga data citra yang tidak diinginkan dapat diminimalkan.
4. Segmentasi citra adalah proses pengelompokan citra yang telah diolah terlebih dahulu untuk memisahkan nomor pelat dengan barcode yang digunakan.
5. Setelah proses segmentasi berhasil dilakukan, program akan melakukan *template matching* untuk identifikasi karakter pada pelat kendaraan.
6. Data yang didapat dari proses identifikasi akan digabungkan dalam sebuah *array*, lalu dibandingkan dengan data *array* yang telah disimpan sebelumnya sebagai database pengguna parkir yang telah terdaftar.
7. Ketika data yang diidentifikasi benar dan cocok dengan database yang telah ada, maka Matlab akan mengirimkan perintah berupa data '1' kepada mikrokontroler untuk selanjutnya menggerakkan pintu garasi.

Dan berikut ini adalah diagram alir dari proses sistem kendali yang akan dibuat. Proses ini dimulai dengan inisialisasi sensor jarak pada mikrokontroler dan deprogram untuk pembacaan jarak < 100 cm. Kamera akan aktif ketika PC menerima data '1' dari mikrokontroler. Citra yang ditangkap akan dilakukan pengolahan citra digital untuk mengetahui karakter dari citra tersebut. Apabila data hasil citra sesuai dengan database maka motor servo akan membuka pintu garasi.

4. PENGUJIAN DAN ANALISA

4.1 Sensor Ultrasonik

Tujuan pengujian ini adalah untuk mengetahui ketepatan atau akurasi sensor ultrasonik dalam mengukur jarak dan apakah sensor dapat bekerja dengan baik. Jarak yang diukur adalah jarak pada rentang 0-120 cm. Pengujian dilakukan dengan cara menghubungkan sensor ultrasonik yang digunakan dengan mikrokontroler Arduino Uno yang telah diprogram pembacaan jarak. Hasil jarak yang dideteksi akan ditampilkan melalui LCD. Hasil jarak ini akan dibandingkan dengan jarak benda sebenarnya yang telah diukur sebelumnya. Hasil pengujian meliputi data jarak perbandingan antara jarak sebenarnya dengan jarak pengukuran dengan sensor ultrasonik. Berikut ini adalah hasil pengujian dan analisis sensor ultrasonik yang dilakuka

No.	Jarak Benda Sebenarnya (cm)	Rata-Rata Jarak Pembacaan Sensor (cm)
1	10	11.2
2	20	20.6
3	30	32.1
4	40	41.3
5	50	50.4
6	60	59.4
7	70	71.5
8	80	82.1
9	90	90.3
10	100	101.2
11	110	112.1
12	120	121.1

Tabel 4.1 Hasil Pengujian dan Analisis Sensor Ultrasonik

Berdasarkan hasil pengujian dan analisis sensor ultrasonik pada Tabel 4.1 diketahui bahwa, kinerja sensor ultrasonik yang digunakan cukup baik namun masih belum akurat jarak yang dideteksi. Pengujian ini

dilakukan uji coba sebanyak 10 kali percobaan. Sedangkan untuk hasil data yang diambil untuk pembacaan sensor adalah rata-rata dari seluruh percobaan.



4.2 Pengujian Pengambilan Gambar (Capture Image)

Pengujian ini dilakukan bertujuan untuk mengetahui lama waktu yang dibutuhkan ketika proses pengambilan citra dan mengetahui apakah sistem yang dibuat dapat berjalan dengan baik. Parameter pengujian yang akan dilihat pada bagian ini adalah jarak pembacaan sensor, pengiriman data, dan waktu yang dibutuhkan ketika pengambilan gambar. Pengujian dilakukan dengan melakukan percobaan, yaitu meletakkan plat nomor kendaraan percobaan di depan *prototype* garasi yang sudah dibuat. Setelah itu program akan dijalankan melalui *personal computer* yang didalamnya terdapat *software* Matlab lalu sensor ultrasonik akan melakukan pendeteksian keberadaan benda di depannya. Ketika sensor ultrasonik telah mendeteksi keberadaan benda, maka sensor ultrasonic juga menghitung jarak benda tersebut. Ketika jarak yang dideteksi berada pada jarak < 100 cm, maka mikrokontroller Arduino Uno akan mengirimkan data ‘1’ ke PC. Selanjutnya *webcam* akan aktif dan mulai untuk melakukan pengambilan gambar. Hasil pengujian dari percobaan ini ditunjukkan pada Tabel 4.2.

No.	Pembacaan Sensor (cm)	Data Dikirim	Data Diterima	Pengambilan Gambar	Waktu Pengambilan (s)
1	20	1	1	Berhasil	5
2	30	1	1	Berhasil	5
3	40	1	1	Berhasil	6
4	50	1	1	Berhasil	5
5	70	1	1	Berhasil	6
6	80	1	1	Berhasil	5
7	100	1	1	Berhasil	5
8	110	0	0	Gagal	-

Tabel 4.2 Hasil Pegujian dan Analisis Pengambilan Gambar

Dari 8 kali percobaan dengan jarak yang berbeda dapat disimpulkan, bahwa sistem bekerja dengan baik namun waktu pengambilan gambar tidak linier. Ketika pengujian dilakukan terhadap benda dengan jarak > 100 cm, maka program mengasumsikan tidak ada benda di depan *prototype* garasi. Dari pengujian ini juga, dapat disimpulkan bahwa pada jarak 80-100 cm merupakan jarak ideal untuk melakukan pengambilan gambar dengan kualitas gambar yang baik.

4.3 Pengujian Deteksi Plat Nomor Kendaraan

Tujuan dari pengujian ini adalah untuk mengetahui tingkat keberhasilan deteksi dan pengolahan citra digital sistem yang dibuat terhadap intensitas cahaya ruangan pengujian. Pengujian ini dilakukan dengan melakukan pengolahan citra ketika gambar telah diambil sebelumnya. Proses yang terjadi adalah *cropping image* atau pemotongan gambar untuk mendapatkan ROI (*region of interest*) atau bagian yang hanya akan dideteksi sehingga gambar yang dipotong hanya menyisakan bagian nomor platnya saja. Proses selanjutnya adalah proses *grayscale detection* untuk pendeteksian plat nomor. Intensitas cahaya ruangan pada saat pengujian berbeda-beda untuk mengetahui tingkat keberhasilan pada saat pengolahan citra dilakukan. Hasil pengujian dan analisis yang dilakukan dapat dilihat pada Tabel 4.3. Hasil pengujian berdasarkan 10 kali percobaan dengan intensitas cahaya yang berbeda, yaitu kondisi cahaya ruangan terang, redup, dan gelap.

No.	Kondisi Cahaya	Keberhasilan	
		Berhasil	Gagal
1	Terang	√	
2	Terang	√	
3	Terang	√	
4	Terang	√	
5	Redup	√	
6	Redup		√
7	Redup		√
8	Gelap		√
9	Gelap		√

10	Gelap		√
----	-------	--	---

Tabel 4.3 Hasil Pengujian dan Analisis Deteksi Plat Nomor

4.4 Pengujian Identifikasi Karakter Plat Nomor Kendaraan

Pengujian ini dilakukan untuk mengetahui tingkat keberhasilan proses identifikasi karakter plat nomor kendaraan. Hasil dari proses identifikasi akhirnya akan melalui proses *Template Matching*. Citra yang telah melalui proses pengolahan citra digital selanjutnya akan diidentifikasi setiap karakternya melalui proses segmentasi karakter. Selanjutnya setelah proses segmentasi selesai, akan didapatkan data berupa *char* untuk masing masing karakter. Data berupa *char* yang telah didapatkan akan melalui proses *Template Matching*. Hasil pengambilan gambar sangat mempengaruhi proses identifikasi karakter yang akan dilakukan, maka diperlukan beberapa kali percobaan untuk mendapatkan hasil identifikasi yang baik. Berikut ini adalah contoh plat yang akan diidentifikasi setiap karakternya.



Gambar 4.1 Contoh Plat yang Diidentifikasi

Hasil pengujian dapat dilihat pada Gambar 4.2 di bawah ini.



Gambar 4.2 Hasil Proses Identifikasi Karakter

Hasil proses identifikasi karakter pada Gambar 4.8 akan melalui proses *Template Matching* sehingga didapatkan data dalam bentuk *char*. *Char* yang didapatkan dapat dilihat pada Tabel 4.4.

No.	Char Image	Char Template Matching
1	D	D
2	3	3
3	I	I, T
4	O	O

Tabel 4.4 Karakter Proses *Template Matching*

Dapat dilihat dari Tabel 4.4 bahwa identifikasi karakter dengan menggunakan metode Template Matching tidak selalu benar 100% dengan data yang sebenarnya. Persentase hasil analisis dapat dilihat pada dibawah ini.

$$\frac{3}{4} \times 100\% = 75\%$$

Selanjutnya, kita akan melihat tingkat akurasi identifikasi karakter yang dilakukan melalui 5 kali percobaan. Hasil tingkat akurasi identifikasi dapat dilihat pada Tabel 4.5 di bawah ini.

No.	Karakter	Hasil Identifikasi Percobaan Ke-								Tingkat Akurasi
		1	2	3	4	5	6	7	8	
1	D	D	D	D	D	D	D	D	D	100%
2	3	3	3	3	3	3	3	3	3	100%
3	I	I	I	I	T	I	I	I	T	75%
4	O	O	O	O	O	O	O	O	O	100%

Tabel 4.5 Hasil Tingkat Akurasi Identifikasi Karakter

Berdasarkan hasil tingkat akurasi identifikasi karakter pada Tabel 4.5 dapat diketahui, bahwa terdapat beberapa karakter yang tingkat akurasinya sangat rendah. Bentuk huruf atau *font* dari plat nomor kendaraan yang dideteksi sangat mempengaruhi hasil identifikasi karakter yang akan dilakukan. Sehingga ada beberapa pengujian dimana database yang dimasukkan menyesuaikan dengan plat yang akan dideteksi, seperti huruf D diganti dengan huruf O atau angka 2 diganti dengan huruf Z.

4.5 Pengujian Pengiriman Data Akhir

Tujuan pengujian ini adalah untuk mengirimkan data akhir output '0' atau '1' ke mikrokontroler Arduino Uno dan memberikan perintah kepada motor servo untuk membuka atau menutup pintu garasi serta mengeluarkan tampilan hasil data akhir pada layar LCD. Data akhir yang didapatkan melalui proses pengolahan citra akan dibandingkan dengan database yang sudah ditetapkan sebelumnya untuk menghasilkan data '0' atau '1' sebagai data yang akan dikirimkan ke mikrokontroler Arduino Uno. Pengiriman data membutuhkan waktu selama proses berlangsung dikarenakan pemrograman ini dilakukan secara berkelanjutan. Hasil pengujian dan analisis pengiriman data akhir dapat dilihat pada Tabel 4.6 dan Tabel 4.7 dengan setiap pengujian dilakukan percobaan sebanyak 10 kali. Tabel 4.6 merupakan hasil pengujian

pengiriman data akhir output berupa data '0' atau '1'. Sedangkan Tabel 4.7 merupakan hasil pengujian jeda waktu yang dibutuhkan selama proses pengiriman data akhir dari Matlab ke mikrokontroller Arduino Uno yang dilakukan.

No.	Data yang Dikirim	Hasil Percobaan Ke-									
		1	2	3	4	5	6	7	8	9	10
1	'0'	-	√	-	-	-	√	-	√	-	-
2	'1'	√	-	√	√	√	-	√	-	√	√

Tabel 4.6 Hasil Pengujian Pengiriman Data Akhir

No.	Data yang Dikirim	Jeda Waktu (detik) Percobaan Ke-									
		1	2	3	4	5	6	7	8	9	10
1	'0'	-	5	-	-	-	5	-	4	-	-
2	'1'	6	-	5	6	6	-	5	-	5	5

Tabel 4.7 Jeda Waktu Pengujian Pengiriman Data Akhir

Dari Tabel 4.6 kita dapat menghitung tingkat keberhasilan pengiriman data akhir yang dilakukan. Persentase hasil analisis tingkat keberhasilan pengujian data akhir yang dilakukan adalah.

$$\frac{\text{Jumlah Percobaan Berhasil}}{\text{Jumlah Percobaan}} \times 100\% = \frac{7}{10} \times 100\% = 70\%$$

Data yang dikirim melalui pengujian ini adalah data '0' dan '1'. Kondisi pengiriman data akhir '1' adalah saat hasil pengolahan citra dan *Template Matching* yang dilakukan sesuai dengan database sehingga pada akhirnya akan menggerakkan motor servo untuk membuka pintu garasi. Sebaliknya, kondisi pengiriman data akhir '0' adalah saat hasil pengolahan citra dan *Template Matching* yang dilakukan tidak sesuai dengan database yang membuat motor servo tidak melakukan respon sama sekali terhadap pintu garasi.

5. Kesimpulan

Dari hasil pengujian dan analisis yang telah dilakukan pada perancangan dan implementasi garasi pribadi dengan pintu otomatis berdasarkan pengenalan plat kendaraan berbasis pengolahan citra digital, maka dapat ditarik beberapa kesimpulan yaitu:

1. Pengolahan citra digital sangat membantu pekerjaan manusia, khususnya dari sistem otomatis sehingga dapat memudahkan pekerjaan dan mengurangi *human error* pada saat suatu sistem bekerja.
2. Keberhasilan pengolahan citra digital pada sistem ini sangat bergantung pada bentuk huruf atau *font* plat nomor kendaraan yang dideteksi untuk menghindari *error* pada saat proses *Template Matching*.
3. Proses *cropping image* yang tidak sempurna yang diakibatkan oleh pengambilan jarak yang kurang tepat sering membuat sistem ini menjadi *error* atau data yang dipakai tidak valid.
4. Masih terdapat *delay* yang signifikan ketika sistem melakukan pemrograman pengolahan citra dan pengiriman data dari mikrokontroller Arduino Uno ke Matlab ataupun sebaliknya.
5. Waktu yang dibutuhkan sistem ini bekerja adalah sekitar 20 detik untuk setiap pengolahan citra dan pengiriman data akhir dengan tingkat performansi akurasi sebesar 70%.

6. Daftar Pustaka

- [1]. Hariyani, Yuli Sun. 111060089. (2010). *Pengenalan Plat Kendaraan Berbasis Pengolahan Citra Digital dan Jaringan Syaraf Tiruan Self-Organizing Maps (SOM)*, Institut Teknologi Telkom, Bandung.
- [2]. Gonzalez, Rafael C. and Woods, Richard. (2008). *Digital Image Processing*, 3rd Ed. USA: Prentice Hall, New Jersey.
- [3]. Cakrawala, Hardy. (2011). *Aplikasi Komputer Pada Pencitraan Digital*. Purwokerto, Jawa Tengah.
- [4]. Pramarta, I Made Andi. (2012). *Identifikasi Karakter Huruf dan Angka Pada Nomor Kendaraan Polisi Dengan Menggunakan Metode Edge Detection, Edge Enhancement dan Backpropagation*. Universitas Udayana, Bukit Jimbaran.
- [5]. Reswandi, Jhordy. (2015). *Rancang Bangun Prototype Kendali Pintu Gerbang Parkir Berbasis Pelat Nomor Polisi dan Barcode Menggunakan Pengolahan Citra Digital*. Telkom University, Bandung.
- [6]. Puspitasari, Astri. Usman, Koredianto. Iwut, Iwan. (2008). *Implementasi dan Perbandingan Sistem Pengenalan Plat Nomor Mobil Dengan Menggunakan Jaringan Syaraf Tiruan Backpropagation dan Metode Template Matching*. Institut Teknologi Telkom, Bandung.