PERANCANGAN DAN IMPLEMENTASI PROTOTIPE SISTEM KEAMANAN MOBIL BERBASIS PENGENALAN WAJAH DAN SISTEM OPERASI ANDROID SEBAGAI KONTROL

Design and Implementation of Car Security System Based on Face Recognition and Android Operating System as Controller

Novian Permana[1], Inung Wijayanto, S.T.,M.T.[2], Angga Rusdinar, S.T.,M.T.,Ph.D.[3]

[1,2,3]Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom Jln. Telekomunikasi No.1 Terusan Buah Batu Bandung 40257 Indonesia

novianpermana@students.telkomuniversity.ac.id iwijayanto@telkomuniversity.ac.id anggarusdinar@telkomuniversity.ac.id

Abstrak

Penelitian ini membahas perancangan sebuah sistem keamanan mobil berbasis pengenalan wajah yang dibantu oleh sebuah aplikasi pengontrol berbasis sistem operasi android. Metode yang digunakan pada penelitian ini adalah *Local Binary Patterns* (LBP). Kamera yang dipasang pada dashboard mobil akan memindai wajah pengguna, jika cocok arus listrik dari power igniter yang telah diputus oleh relay akan disambung sehingga mesin mobil dapat diaktifkan. Terdapat beberapa pesan yang akan dikirimkan sistem yang terpasang di mobil kepada aplikasi di *smartphone* android, yaitu jika pengenalan wajah berhasil, pengenalan wajah gagal, relay berhasil diaktifkan dan dinonaktifkan oleh aplikasi *smartphone*. Secara keseluruhan tingkat akurasi sistem pada pernelitian ini mencapai 98.5%. Hasil yang didapatkan dari penelitian ini menunjukkan bahwa kondisi cahaya terang dengan ekspresi senyum memiliki tingkat performansi yang terbaik, pencapaian tingkat akurasi sebesar 100% keberhasilan dengan rata-rata nilai confidence 20,06547 dan 2.6883 detik untuk rata-rata waktu komputasi.

Kata kunci : *Face Recognition*, Sistem Keamanan Mobil, Kontrol Mesin Mobil, *Local Binary Patterns* , Raspberry Pi.

Abstract

This Research explains how to design and implement car security system based on face recognition using an application based on android operating system as controller. The method used in this research is Local Binary Patterns (LBPH). The Camera that located in car dashboard will recognize user's face, if it's match, relay will connect the electric current from the battery to the dynamo starter so the engine can be activated. There will be some message that will be sent to smartphone as notification if face recognition accepted, face recognition failed, relay successful activated and deactivated by the app in smartphone. Overall, the successful rate of the system reach 98.5%. The experiment of this research shows that bright light condition with smile expression gave the best result with 100% successful rate and the mean of confidence value is 20,06547 and 2.6883 seconds for computing time.

Keywords : Face Recognition, Car Security System, Car Engine Control, Local Binary Patterns, Raspberry Pi.

1. Pendahuluan

1.1 Latar Belakang Masalah

Komunikasi antar mesin merupakan hal yang lazim dewasa ini, bahkan hal tersebut kini dapat juga dijumpai pada perangkat mobile seperti kendaraan dan *smartphone*. Sistem keamanan mobil yang menggunakan kunci mekanik dirasa sudah ketinggalan jaman, terlebih mobil-mobil jaman sekarang telah banyak mengadopsi sistem *keyless entry*. Mesin mobil dapat diaktifkan hanya dengan menekan tombol *start and stop* tanpa harus memasukkan kunci mekanik. Namun *keyless entry* tidaklah cukup, sebuah mobil harus mengenal siapa pemiliknya untuk dapat menjalankan tugasnya.

Tomy [1], telah melakukan perancangan sistem keamanan kendaraan bermotor menggunakan pin sebagai password, namun kunci tersebut dinilai masih terlalu riskan, karena password dapat dimiliki oleh siapa saja sehingga siapapun dapat mengakifkan mesin kendaraan tersebut. Penelitian yang dilakukan oleh Setya [2], menggunakan sistem pengenalan wajah sebagai kunci pada sistem keamanan rumah, sistem tersebut sangat baik, namun akan lebih bagus jika diimplementasikan pada objek yang dapat bergerak. Pada Tugas Akhir kali ini, dirancang sebuah sistem keamanan mobil dirancang sebuah sistem keamanan kendaraan bermotor yang dapat mengontrol aktivitas mesin kendaraan bahkan saat pemilik kendaraan berada dalam posisi yang sangat jauh dengan kendaraannya. Metode yang digunakan adalah *Local Binary Patterns* yang merupakan pengembangan dari metode Eigenface dan Fisherface, sehingga diharapkan sistem memiliki tingkat akurasi yang cukup baik untuk diimplementasikan di mobil.

2. Dasar Teori

2.1 Pengolahan Citra Digital

Citra adalah suatu representasi, kemiripan, atau imitasi dari suatu objek atau benda yang dapat ditampilkan pada bidang dua atau tiga dimensi (dwimatra atau trimatra) [Kamus Webster]. Representasi citra dalam bentuk lain misalnya : foto mewakili entitas diri seseorang di depan kamera, foto sinar-X thorax mewakili keadaan bagian dalam tubuh seseorang, data dalam suatu file BMP (Bitmap) mewakili apa yang digambarkannya. Citra sebagai keluaran dari suatu sistem perekaman data dapat bersifat :

1. Optik berupa foto/ gambar
2. Analog berupa sinyal video seperti gambar pada monitor televisi
3. Digital yang dapat langsung disimpan pada media penyimpan magnetik

Citra juga dapat dikelompokkan menjadi 2 bagian yaitu :

1. Citra tampak (foto, gambar, lukisan, apa yang nampak di layar monitor/ televisi, hologram , dan lain-lain)
2. Citra tidak tampak (data foto/ gambar dalam file, citra yang direpresentasikan dalam fungsi matematis).

Citra digital adalah suatu citra $f(x,y)$ yang memiliki koordinat spatial, dan tingkat kecerahan yang diskrit. Fungsi $f(x,y)$, dapat dilihat sebagai fungsi dengan dua unsur. Unsur yang pertama merupakan kekuatan sumber cahaya yang melingkupi pandangan kita terhadap objek (illumination). Unsur yang ke dua merupakan besarnya cahaya yang direfleksikan oleh objek ke dalam pandangan (reflectance components). Keduanya dituliskan sebagai fungsi $i(x,y)$ dan $r(x,y)$, yang digabungkan sebagai perkalian fungsi untuk membentuk fungsi $f(x,y)$. Fungsi $f(x,y)$, dapat dituliskan dengan persamaan :

$$f(x,y) = i(x,y) * r(x,y) \quad (1)$$

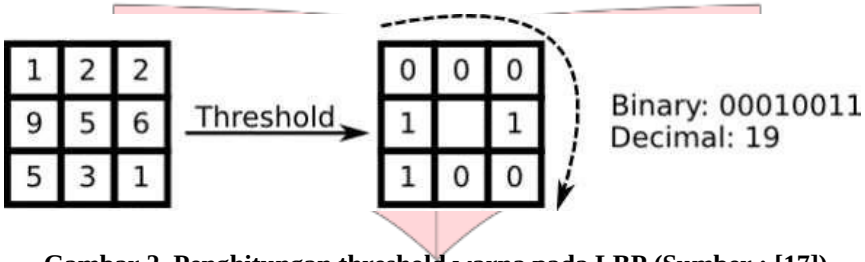
Dengan : $0 < i(x,y) < \infty$ dan $0 < r(x,y) < 1$. Citra digital merupakan suatu matriks yang terdiri dari baris dan kolom, setiap pasangan indeks baris dan kolom menyatakan suatu titik pada citra. Nilai matriksnya menyatakan nilai kecerahan titik tersebut. Titik-titik tersebut dinamakan sebagai elemen citra, atau pixel (*picture element*). [Gonzales and Woods, 2002].

Gambar 1. Pengolahan Citra (Sumber : [1])

2.2 *Local Binary Patterns* (LBP)

LBP (*Local Binary Patterns*) merupakan pengembangan dari algoritma Eigenface dan Fisherface. Konsentrasi dalam algoritma ini adalah untuk mengekstrak fitur-fitur local dari sebuah gambar dengan tidak melihat gambar sebagai vector berdimensi tinggi, tetapi dengan hanya melihat fitur-fitur local yang kuat dari sebuah objek. Fitur-fitur yang disektrak ini akan memiliki dimensi yang rendah secara implisit.

Ide utama dari *Local Binary Pattern*s adalah untuk meringkaskan struktur local dari suatu gambar dengan membandingkan tiap-tiap piksel dengan piksel tetangganya. Ambil sebuah piksel sebagai piksel tengah dan jadikan patokan terhadap piksel-piksel tetangga. Jika nilai piksel tetangga lebih dari atau sama dengan piksel tengah, maka piksel tetangga tersebut akan bernilai 1, lainnya 0. Jika terdapat 8 piksel tetangga, maka ada 2^8 kemungkinan kombinasi.



Gambar 2. Penghitungan threshold warna pada LBP (Sumber : [17])

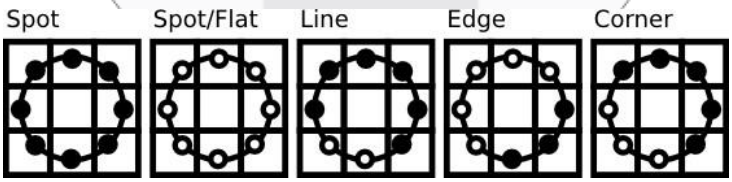
Berikut merupakan Algoritma pada *Local Binary Patterns* (LBP) :

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c) \quad (2)$$

Dengan (x_c, y_c) sebagai piksel tengan, dengan i_c dan i_n menjadi intensitas dari piksel tetangganya. Maka s dapat didefinisikan dengan :

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & lainnya \end{cases} \quad (3)$$

Fungsi tersebut memungkinkan kita untuk mendapatkan titik-titik detail pada gambar dengan sangat baik. Setelah itu diberikan catatan bahwa piksel-piksel tetangga gagal untuk dikodekan secara rinci dalam skala. Jadi operator ditambahkan dengan menggunakan variabel tetangga untuk meluruskan nomor-nomor piksel tetangga kedalam lingkaran dengan variabel radius, yang memungkinkan kita menangkap piksel tetangga berikutnya :



Gambar 3. Tipe hasil threshold warna pada LBP (Sumber :[17])

Dengan piksel tetangga didefinisikan sebagai (x_p, y_p), $p = P$ maka dapat dihitung bahwa :

$$x_p = x_c + R\cos\left(\frac{2\pi p}{P}\right) \quad (4)$$

$$y_p = y_c - R\sin\left(\frac{2\pi p}{P}\right) \quad (5)$$

Dengan R adalah radius dari lingkaran dan P adalah jumlah dari *sample point.* Jika koordinat titik tidak sesuai dengan koordinat gambar, maka titik tersebutakan diinterpolasi. Pada OpenCV digunakan interpolasi bilinear :

$$f(x,y) \approx [1-x \quad x] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} \begin{bmatrix} 1-y \\ y \end{bmatrix} \qquad (6)$$

Dengan definisi tersebut, operator LBP merupakan penguatan terhadap transformasi grayscale yang monoton. Variabel confidence pada LBPH merupakan nilai pertidaksamaan antara dua data, nilai ini didapat dari persamaan chi square distance:

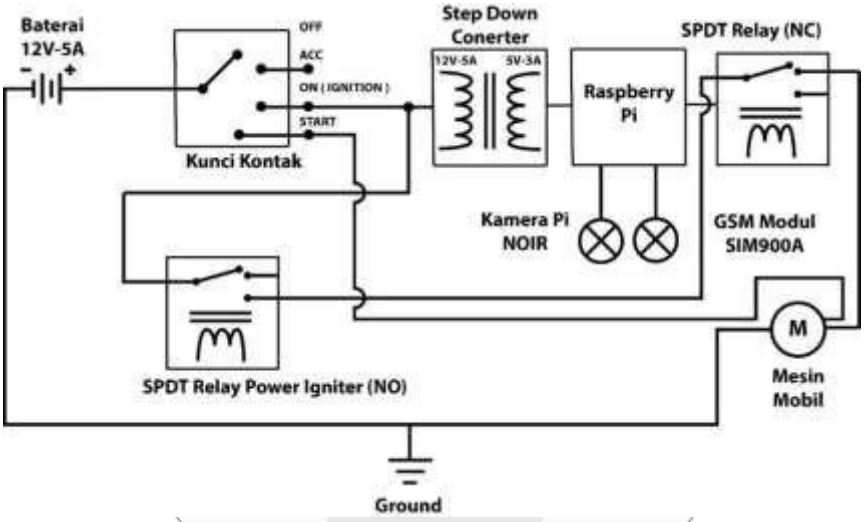
$$\chi^2(S,M) = \sum_{b=1}^{B} \frac{(S_b - M_b)^2}{S_b + M_b} \qquad (7)$$

Dengan S dan M ditunjukkan (diskrit) sebagai data uji dan data latih masing-masing. S_b dan M_b adalah probabilitas dari *bin b* pada data latih dan data uji. B adalah jumlah *bins* dalam distribusi.

3. Desain Model Sistem

3.1 Skema Kelistrikan Sistem

Pada tahap ini dijelaskan skema kelistrikan sistem secara keseluruhan. Berikut merupakan model kelistrikan sistem :



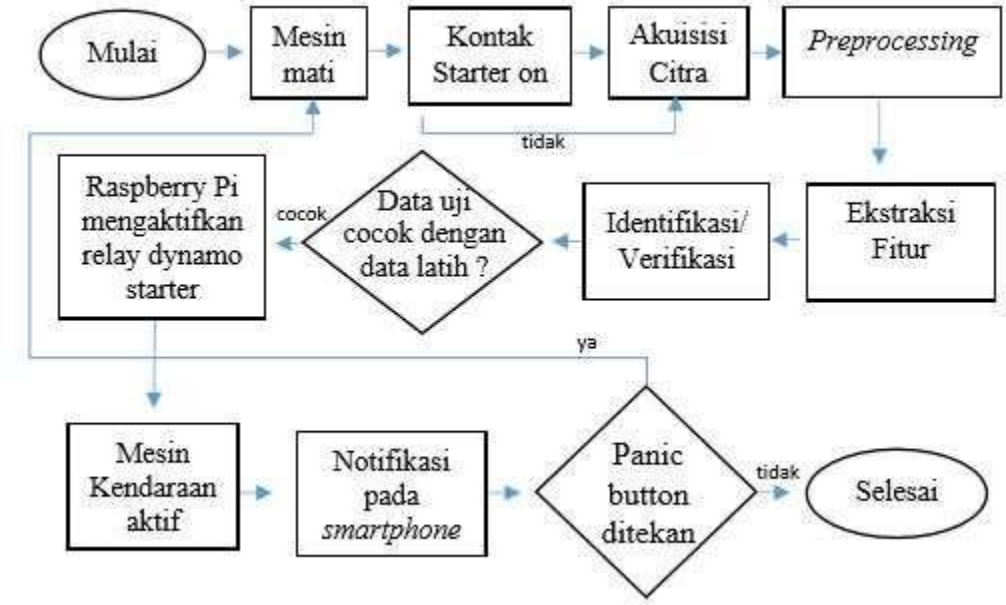
Gambar 4. Skema Kelistrikan Sistem

Baterai mobil yang bertegangan 12V adalah sumber daya dari sistem. Arus yang mengalir dari baterai mobil akan diteruskan ke kunci kontak mobil. Kunci kontak mobil mempunyai empat kondisi, yaitu : Off, ACC, ON dan start. Sistem akan aktif saat kunci kontak berada di posisi ON. Pada kondisi ON, arus diteruskan dan dibagi secara parallel menuju step down converter dan relay power igniter, yang mana arus dari relay power igniter telah diputus oleh relay Raspberry Pi. Step down converter berfungsi mengubah tergangan dari baterai yang semula 12V-5A menjadi 5V-3A sehingga aman untuk mensupply arus untuk Raspberry Pi.

Arus yang didapat Raspberry Pi akan dibagi menuju Pi Camera, GSM modul dan relay. Raspberry Pi akan memindai wajah calon *user*. Jika cocok, maka arus listrik akan diteruskan ke power igniter pada mesin mobil, sehingga mesin mobil bisa diaktifkan.

3.2 Desain sistem keseluruhan

Tahap ini adalah proses kerja sistem secara keseluruhan. Berikut adalah diagram alir dari keseluruhan sistem :

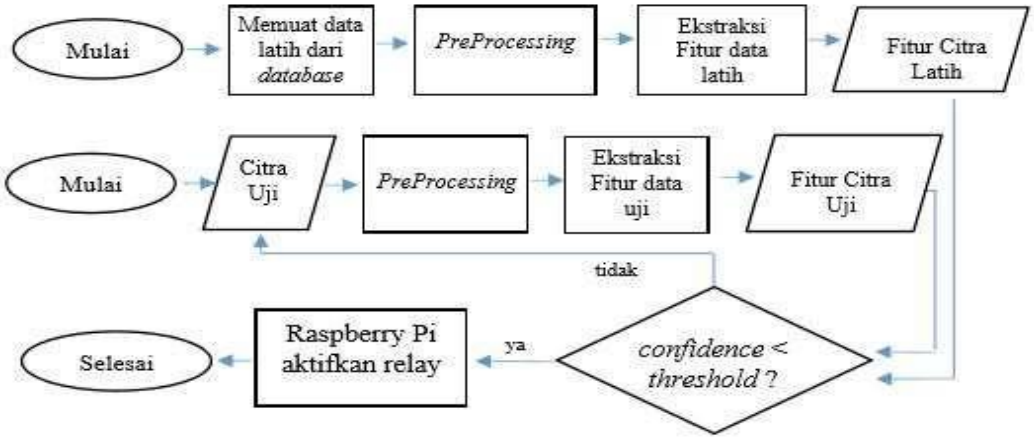


Gambar 5. Diagram Alir Sistem

Untuk dapat menghidupkan Raspberry Pi, maka dibutuhkan arus listrik yang didapat dari baterai mobil yang bertegangan 12V, disambungkan pada kontak starter diposisi on. Namun tegangan 12V terlalu besar untuk Raspberry Pi yang hanya membutuhkan tegangan 5V, sehingga dibutuhkan step down converter untuk mengkonversi tegangan 12V-5A menjadi 5V-3A. Relay power igniter diset dalam keadaan *Normally Open* (NO), sedangkan SPDT relay yang terhubung pada raspberry pi diset dalam keadaan *Normally Close* (NC).

3.3 Klasifikasi

Pada tahap ini, data latih yang telah disimpan di database akan dicocokkan dengan data uji untuk selanjutnya Raspberry Pi akan melakukan aksi apakah mesin dapat diaktifkan atau tidak sesuai hasil klasifikasi pengenalan wajah. Berikut merupakan diagaram alir klasifikasi :



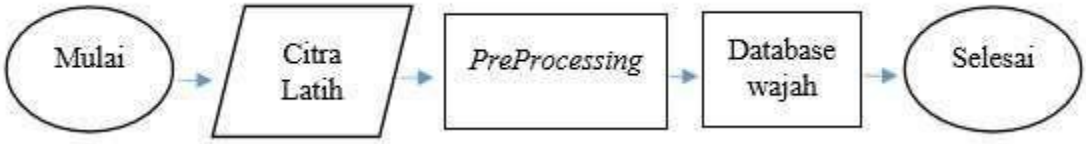
Gambar 6. Diagram Alir Pengenalan Wajah

Proses pada tahap ini dimulai dengan memuat seluruh data latih dari database, setelah itu dilakukan ekstraksi fitur-fitur lokal dari citra latih menggunakan metode *Local Binary Patterns*, kemudian hasil perhitungan dari ekstraksi fitur tersebut akan disimpan terlebih dahulu. Proses selanjutnya dilakukan akuisisi citra uji, setelah itu citra uji tersebut masuk pada tahap preprocessing untuk dapat diolah pada proses selanjutnya yaitu ekstraksi ciri menggunakan *Local Binary Pattern*s.

Setelah didapat hasil perhitungan ekstraksi ciri dari citra latih dan citra uji, kedua hasil perhitungan tersebut akan dibandingkan menggunakan *Chi Square Formula*. Parameter yang digunakan dalam klasifikasi pada proses verifikasi adalah confidence threshold, jika citra uji dinilai cocok oleh sistem sesuai dengan nilai threshold yang telah ditentukan, maka komputer papan tunggal Raspberry Pi yang ertindak sebagai mikrokontroller akan melakukan aksinya mempersilahkan arus dari baterai untuk menuju power igniter dengan mengubah kondisi relay yang telah diatur dalam kondisi *Normally Open*.

3.4 Pengambilan Data Latih

Data latih dibutuhkan sebagai password untuk selanjutnya dicocokkan pada data latih. Berikut merupakan diagram alir pengambilan data latih :

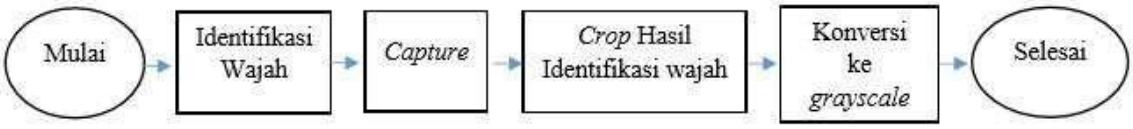


Gambar 7. Pengambilan data latih

Proses pertama yang dilakukan adalah akuisisi citra atau pengambilan citra, selanjutnya akan dilakukan *preprocessing* (akan dijelaskan pada sub bab selanjutnya), setelah *preprocessing* data akan disimpan dalam *database*.

3.5 *PreProcessing*

Pada tahap ini, data input diolah agar bisa fitur-fitur yang terdapat dalam citra dapat diekstraksi. Terdapat beberapa proses dalam tahap ini, berikut merupakan diagram alir *preprocessing :*

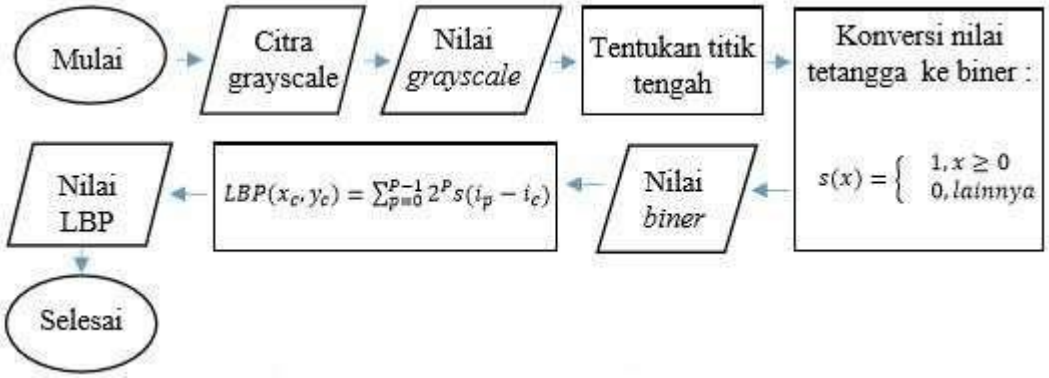


Gambar 8. *PreProcessing*

Proses pertama yang dilakukan sistem adalah mendeteksi apakah ada wajah yang tertangkap oleh kamera. Jika ada, maka proses selanjutnya adalah *capture* citra untuk selanjutnya citra tersebut dipotong ke ukuran 320x240 piksel, proses pada tahap ini diakhiri dengan konversi citra dari RGB ke *grayscale*.

3.6 Ekstraksi Ciri

Pada tahap ini, citra diolah agar dapat ditentukan cirinya, berikut merupakan diagram alir ekstraksi ciri :



Gambar 9. Ekstraksi Ciri

Proses pertama yang dilakukan adalah mendapatkan nilai *grayscale* yang didapat dari *preprocessing*. Selanjutnya menentukan titik tengah sehingga secara otomatis didapat piksel-piksel tetangga. Nilai-nilai piksel tetangga tersebut dikonversi ke nilai biner dengan *threshold* nilai titik tengah dan dibaca dalam satu arah sehingga didapatkan nilai LBP dalam bentuk desimal.

4. Hasil Pengujian

Pengujian dilakukan dengan ekspresi yang berbeda dan kondisi cahaya yang juga berbeda. Dilakukan pengambilan citra wajah (capture) sebanyak maksimal sepuluh kali sampai akhirnya wajah dapat dikenali, jika jumlah capture melebihi 10 kali, maka sistem akan mengirim notifikasi ke *smartphone* Selain itu dilakukan perubahan nilai confidence yaitu 25 dan 30 untuk mengetahui performansi sistem untuk mengenali tiap user yang diperbolehkan mengakses mobil. Secara keseluruhan, tingkat keberhasilan sistem dalam mengenali wajah pengguna sebesar 98.5% dengan rata-rata nilai confidence sebesar 20.5740, 4.1102 detik waktu komputasi dan 1.7315 kali pengambilan citra uji. Pada saat pengujian, waktu proses yang dibutuhkan aplikasi untuk dapat mengubah kondisi relay mencapai angka yang cukup konstan yaitu 19 – 20 detik, begitupun waktu proses yang dibutuhkan mobill untuk mengirimkan notifikasi ke *smartphone* yaitu 7-8 detik. Karena menggunakan media SMS, tentunya waktu proses pengiriman bergantung pada kondisi trafik operator. Berikut adalah tabel hasil pengujian :

4.1 Hasil Pengujian menggunakan threshold confidence < 25

Tabel 1. Kondisi Cahaya Terang (54-77 Lux) dengan threshold confidence <25

Ekspresi Normal	Tingkat keberhasilan	100%
	Rata-rata confidence	19.5129575
	Nilai confidence terbaik	10.5874
	Nilai confidence terendah	24.9308
	Rata-rata waktu komputasi	4.879255
	Waktu komputasi terbaik	1.8212
	Waktu komputasi terlambat	22.0956
	Rata-rata capture	2.225
	Capture terbaik	1
	Capture terbanyak	10
Ekspresi Senyum	Tingkat keberhasilan	97.50%
	Rata-rata confidence	18.4830387
	Nilai confidence terbaik	10.9284

	Nilai confidence terendah	24.0894
	Rata-rata waktu komputasi	3.34295897
	Waktu komputasi terbaik	2.1305
	Waktu komputasi terlambat	10.1353
	Rata-rata capture	1.41025641
	Capture terbaik	1
	Capture terbanyak	4

Pada pengujian di kondisi cahaya terang ekspresi senyum diperoleh nilai rata-rata confidence terbaik, hal ini karena fitur citra terlihat lebih jelas dan unik disbandingkan dengan ekspresi normal.

Tabel 2. Kondisi Cahaya Gelap (0 Lux) dengan threshold confidence <25

Ekspresi Normal	Tingkat keberhasilan	95%
	Rata-rata confidence	19.63485789
	Nilai confidence terbaik	10.5874
	Nilai confidence terendah	24.669
	Rata-rata waktu komputasi	5.535762895
	Waktu komputasi terbaik	1.8212
	Waktu komputasi terlambat	24.0452
	Rata-rata capture	2.342105263
	Capture terbaik	1
	Capture terbanyak	10
Ekspresi Senyum	Tingkat keberhasilan	100.00%
	Rata-rata confidence	20.80679
	Nilai confidence terbaik	10.4192
	Nilai confidence terendah	24.7644
	Rata-rata waktu komputasi	5.971715
	Waktu komputasi terbaik	2.1897
	Waktu komputasi terlambat	22.3372
	Rata-rata capture	2.375
	Capture terbaik	1
	Capture terbanyak	10

Pada pengujian di kondisi cahaya gelap dengan ekspresi senyum memperoleh waktu komputasi paling lama, hal ini karena fitur citra kurang terlihat dengan jelas disertai threshold confidence yang cukup ketat.

Tabel 3. Kondisi Cahaya Terang (54-77 Lux) dengan threshold confidence <30

Ekspresi Normal	Tingkat keberhasilan	100%
	Rata-rata confidence	21.2587
	Nilai confidence terbaik	10.5874
	Nilai confidence terendah	29.7689
	Rata-rata waktu komputasi	3.00532

	Waktu komputasi terbaik	1.8212
	Waktu komputasi terlambat	9.9653
	Rata-rata capture	1.3
	Capture terbaik	1
	Capture terbanyak	4
Ekspresi Senyum	Tingkat keberhasilan	100.00%
	Rata-rata confidence	20.0646
	Nilai confidence terbaik	10.9284
	Nilai confidence terendah	28.9995
	Rata-rata waktu komputasi	2.68832
	Waktu komputasi terbaik	2.1305
	Waktu komputasi terlambat	7.3675
	Rata-rata capture	1.125
	Capture terbaik	1
	Capture terbanyak	3

Pengujian di kondisi cahaya terang ekspresi senyum di threshold confidence 30, diperoleh waktu kondisi tercepat. Hal ini karena nilai threshold confidence yang lebih longgar dan fitur citra terlihat jelas.

Tabel 4. Kondisi Cahaya Gelap (0 Lux) dengan threshold confidence <30

Ekspresi Normal	Tingkat keberhasilan	100%
	Rata-rata confidence	21.19005
	Nilai confidence terbaik	10.5874
	Nilai confidence terendah	29.4365
	Rata-rata waktu komputasi	4.1789
	Waktu komputasi terbaik	1.8212
	Waktu komputasi terlambat	15.7432
	Rata-rata capture	1.8
	Capture terbaik	1
	Capture terbanyak	7
Ekspresi Senyum	Tingkat keberhasilan	100.00%
	Rata-rata confidence	23.54251
	Nilai confidence terbaik	10.4192
	Nilai confidence terendah	29.9044
	Rata-rata waktu komputasi	3.332005
	Waktu komputasi terbaik	2.1897
	Waktu komputasi terlambat	12.1137
	Rata-rata capture	1.275
	Capture terbaik	1
	Capture terbanyak	4

Pengujian di kondisi cahaya gelap dengan threshold confidence 30 memiliki rata-rata waktu komputasi yang lebih cepat dibandingkan threshold confidence 25 di kondisi cahaya gelap.

Tabel 5. User tak dikenali

Kondisi Cahaya Terang (54-77 Lux)	Tingkat keberhasilan	0%
	Rata-rata confidence	54.89066
	Nilai confidence terbaik	40.5814
	Nilai confidence terendah	112.319
	Rata-rata waktu komputasi	-
	Waktu komputasi terbaik	-
	Waktu komputasi terlambat	-
	Rata-rata capture	-
	Capture terbaik	-
	Capture terbanyak	-
Kondisi Cahaya Gelap (0 Lux)	Tingkat keberhasilan	0%
	Rata-rata confidence	61.77376
	Nilai confidence terbaik	46.6518
	Nilai confidence terendah	80.8008
	Rata-rata waktu komputasi	-
	Waktu komputasi terbaik	-
	Waktu komputasi terlambat	-
	Rata-rata capture	-
	Capture terbaik	-
	Capture terbanyak	-

Tidak ada user tidak dikenali yang berhasil masuk ke sistem, dengan confidence terbaik adalah 40.5814.

5. Kesimpulan

Berdasarkan hasil pengujian dan analisis pada penelitian ini, dapat ditarik kesimpulan sebagai berikut :

1. Sistem keamanan mobil yang telah dibuat bekerja dengan baik dan mampu memberikan tingkat keamanan yang tinggi.
2. Metode *Local Binary Patterns* cocok digunakan untuk sistem keamanan mobil karena mempunyai performansi yang sangat baik.
3. Kondisi cahaya dan ekspresi wajah mempengaruhi performansi sistem.
4. Hasil pengujian yang paling tepat untuk sistem keamanan mobil berdasarkan tingkat keberhasilan dan rata-rata waktu komputasi tercepat adalah di kondisi cahaya terang, berekspresi senyum dengan batas confidence <30.
5. Batas confidence yang paling cocok untuk sistem keamanan mobil adalah <30 karena mempunyai tingkat keberhasilan sistem sebesar 100% dan rata-rata waktu komputasi yang cepat.
6. Pengujian pada kondisi cahaya terang secara keseluruhan menghasilkan performa yang lebih baik dibandingkan kondisi cahaya gelap.
7. Hasil pengenalan wajah terbaik didapat oleh user no 3 di kondisi cahaya minim dengan ekspresi senyum dengan tingkat confidence 10.4192.
8. Waktu komputasi tercepat didapat oleh user no 2 di kondisi cahaya minim dengan ekspresi senyum dengan waktu 2.1897 detik.
9. Waktu proses yang dibutuhkan untuk *smartphone* dapat mengontrol relay mencapai 19-20 detik, sedangkan waktu proses mobil untuk mengirimkan notifikasi mencapai 7-8 detik.
10. Aplikasi kontrol mesin berbasis android menggunakan SMS mempunyai *delay* yang dipengaruhi kondisi trafik operator.

6. Daftar Pustaka

[1] O.S.Y. Tomy, *"Penerapan Sistem Keamanan Kendaraan Bermotor Dengan Menggunakan Security Key dan Sensor Kecepatan"*, Makalah Skripsi Telkom University, Bandung, 2015.

[2] Bayu Setya, "*Penerapan Face Recognition Dengan Metode Eigenface Dalam Intelligent Home Security*", Makalah Skripsi Institut Teknologi Sepuluh Nopember, Surabaya, 2009.

[3] I.Amulya dan Mr, K Sreenvisia Rao, "Keyless Car Entry Authentication System Based on A Novel Face-Recognition Structure,"*International Journal of Engineering Trends and Technology (IJETT)*, vol.5, no.5, 2013.

[4] V.Saranya dan V.Sabitha Tamilanjani, "Face Identification In Smart Car Security System in Real Time," dalam *International Conference on Research Advances in Communication, Electrical Science and Structures (NCRACCESS), 2015.*

[5] Yi-Shin Liu, Wai-Seng Ng dan Chun-Wei Liu, "*A Comparison of Different Face Recognition Algorithms"*, National Taiwan University,2009.

[6] Chek Ling Ngo David, Beng Jin Teoh Anderw dan Hu Jiankun, *Biometric Security*, Cambridge Scholars Publishing, United Kingdom, 2015.

[7] Sunitha.M, "*Embedded Car Security System*", CVSR College of Engineering, JNTU Hyderabad, 2014.

[8] V. Balajee Seshasayee dan E. Manikandan, "*Automobile Security System Based on Face Recognition Structure Using GSM Network*", Sri Sairam Engineering College, India, 2013

[9] C.Nandakumar, G.Muralidaran dan N.Tharani, *"Real Time Vehicle Security System through Face Recognition"*, Madras Institute of Technology, Anna University, Chennai, India, 2014.

[10] K.Jaikumar dan B.Jaiganesh, "*An Economical Car Security Authentication System Based On Face Recognition Structure",* Saveetha University, Chennai, India, 2014.

[11] Hermono Hartatio Irfanhady, "*Security car system based GPS and SMS*", Makalah Skripsi Telkom University, Bandung, 2015.

[12] Purnama, A. (2012, Juni 29). *Kelebihan dan Kekurangan Solid State Relay (SSR)*. Diambil kembali dari http://elektronika-dasar.web.id [5 Desember 2015]

[13] Yovi, M. (2015, Februari). *Woocara*. Diambil kembali dari Sejarah android dan nama-nama versi android: http://woocara.blogspot.co.id [5 Desember 2015]

[14] Aggarwal, A. (2016). *Which is the best face recognition algorithm in OpenCV Python*. Diambil kembali dari https://www.quora.com/Which-is-the-best-face-recognition-algorithm-in-OpenCV-Python [24 Agustus 2016]

[15] Md. Abdur Rahim, Md. Najmul Hossain, Tanzillah Wahid & Md. Shafiul Azam,*" Face Recognition using Local Binary Patterns (LBP)"*, Global Journals Inc. (USA), Volume 13,4,1.0, 2013

[16] OpenCV. (2011). Face Recognition With OpenCV - OpenCV 2.4.13.1 documentation. Diambil kembali dari http://docs.opencv.org [24 Agustus 2016]

[17] Wagner, Philipp (2011). *Local Binary Patterns.* Diambil kembali dari http://bytefish.de/blog/local_binary_patterns/ [24 Agustus 2016]

[18] Zipprick, Jonas (2015). Projekt Quadrocopter. Diambil kembali dari http://jonas-zipprick.blogspot.co.id/p/1-msp430-tips-n-sketches.html [24 Agustus 2016)