PERANCANGAN DAN IMPLEMENTASI SISTEM PENDETEKSIAN OBJEK MENGGUNAKAN METODE YCBCR PADA ROBOWAITER DRU99RWE4-V13

Ferry Ebitnaser[1], Taufiq Nuzwir Nizar[2], John Adler[3]

[1,2,3] Jurusan Teknik Komputer Unikom, Bandung
[1]ebitnaser2009@gmail.com, [2]taufiq.nizar@gmail.com, [3]john.adler007 @gmail.com

ABSTRAK

Keterbatasan fisik yang dialami orang penyandang cacat seperti mengambil dan memindahkan makanan dari suatu tempat ke tempat yang lain menjadi hambatan dalam melaksanakan aktivitasnya sehari-hari. Robot merupakan salah satu solusi atau aleternatif yang dapat membantu meringankan pekerjaan atau aktivitas mereka.

Dalam menjalankan tugasnya, robot membutuhkan kemampuan untuk mengetahui posisi makanan yang ingin diambil. Robot dapat mengetahui objek berdasarkan ciri-cirinya, seperti warna, ukuran, bentuk, berat, suhu dan sebagainya. Warna objek tersebut akan digunakan sebagai acuan untuk mengenalinya dengan menggunakan kamera atau sensor optik lainnya. Dalam merancang sebuah sistem pendeteksi objek menggunakan sensor optik memiliki tantangan yang harus dihadapi, yaitu nilai warna yang dapat terpengaruh oleh kondisi intensitas cahaya di sekitarnya.

Berdasarkan permasalahan di atas, didapatkan sebuah ide beserta solusi agar robot dapat melaksanakan tugasnya, yaitu dengan merancang sebuah sistem pendeteksi objek menggunakan metode YCbCr yang mampu mengenali dan mengetahui keberadaan atau posisinya dalam kondisi cahaya yang berbeda, kemudian sistem pendeteksi objek ini diterapkan pada robot untuk melakukan pengambilan objek tersebut.

Kata kunci : Robot, Pendeteksi objek, Metode YCbCr.

1. PENDAHULUAN

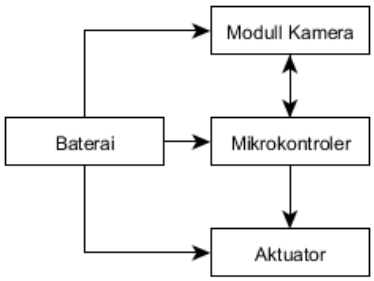
Robot memiliki peranan penting dalam membantu meringankan atau memudahkan pekerjaan yang dilakukan oleh manusia. Salah satu peranan robot yang dapat dirasakan penting manfaatnya adalah robot yang dapat membantu aktivitas orang penyandang cacat khususnya dalam mengambil, memindahkan dan meletakkan makanan dari suatu tempat ke tempat lain.

Robot membutuhkan kemampuan mengenali dan mencari atau mendeteksi keberadaan makanan tersebut agar dapat melaksanakan tugasnya. Robot dapat mengenali makanan tersebut berdasarkan ciri-cirinya, seperti ukuran, bentuk, warna, suhu, berat, dan sebagainya. Karena makanan diletakkan di atas piring, robot dapat mendeteksi keberadaan piring tersebut berdasarkan warnanya. Pendeteksian objek berdasarkan warna dapat menggunakan kamera atau sensor optik lainnya, akan tetapi pendeteksian objek yang menggunakan sensor optik memiliki tantangan berupa perubahan kecerahan atau intensistas cahaya yang dapat mempengaruhi nilai keterbacaan suatu warna.

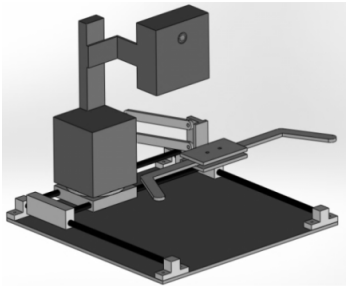
Berdasarkan permasalahan tersebut, maka dilakukan sebuah penelitian mengenai perancangan sistem pendeteksian objek berdasarkan warna yang handal dalam kondisi intensitas cahaya yang berbeda serta pengambilan objek tersebut secara otomatis.

2. PERANCANGAN

Perancangan yang dilakukan terdiri dari perancangan mekanik, elektronik dan algoritma. Perancangan mekanik pada sistem pendeteksi objek ini memiliki dimensi 23 cm x 23cm x 18 cm dengan berat ±1.3 Kg. Material yang digunakan dalam perancangan adalah plastic, nilon, akrilik, aluminium dan besi. Gambar 1 menunjukan diagram blok sistem yang akan dibangun dan Gambar 2 menunjukan desain rancangan sistem.



Gambar 1. Diagram Blok Sistem



Gambar 2. Desain Perancangan Perangkat Keras

Kamera

A. Modul Kamera

Dalam perancangan perangkat keras, modul kamera yang digunakan adalah modul kamera CMUcam3. Modul ini merupakan sebuah modul kamera yang terintegrasi dengan prosesor, modul ini dapat bekerja dengan berbagai model warna pada citra dan memiliki berbagai fitur. Fitur dari modul yang digunakan pada sistem adalah deteksi warna menggunakan metode YCbCr.



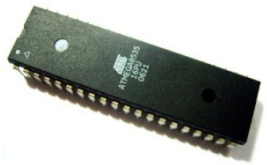
Gambar 3.*CMUCAM3.*

Di dalam perancangan, digunakan satu buah modul kamera yang berfungsi untuk mendeteksi dan mengetahui posisi objek terhadap *gripper*.

Kontroler

A. Mikrokontroler

Mikrontroler digunakan untuk memberi perintah ke modul kamera dan mengambil data keluarannya kemudian memutuskan aksi apa yang akan dilakukan berdasarkan data tersebut. Didalam perancangan dan pembuatan, mikrokontroler yang digunakan adalah Atmega8535. Pemilihan jenis ini karena jumlah pin I/O, memori dan fitur yang mencukupi untuk kebutuhan perancangan sistem ini.



Gambar 4. Mikrokontroler Atmega 8535.

B. IC *Driver Motor*

Mikrokontroler umumnya tidak dapat secara langsung mengendalikan kecepatan motor dc karena keterbatasan tegangan dan arus listrik yang dapat dialirkan oleh pin I/O pada mikrokontroler. Oleh karena itu, dibutuhkan komponen lain yang dapat memenuhi kekurangan tersebut. Ic *driver motor* dapat mengendalikan arah dan kecepatan motor dc secara langsung. Dalam perancangan ini digunakan ic L298 yang dapat mengendalikan 2 buah motor dc secara bersamaan. Ic L298 ini memiliki 3 buah pin input untuk mengatur sebuah motor dc, yaitu 2 pin untuk mengendalikan arah dan pin yang lainnya untuk mengatur kecepatannya. Ic L298 dapat mengalirkan tegangan sampai 46Vdc dan arus listrik sampai 2A.



Gambar 5. *IC DriverMotor* L298.

Aktuator

A. Motor DC

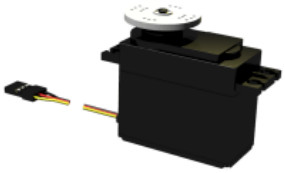
Motor dc yang digunakan pada aktuator berfungsi untuk menggerakan mekanik *gripper* ke kanan-kiri dan maju-mundur saat menyesuaikan posisi *gripper* dengan objek. Motor dc dapat bekerja dengan memberikan daya listrik ke kedua kutubnya (kutub + dan kutub -). Motor dc yang digunakan pada perancangan perangkat keras ini memiliki tegangan kerja 5Vdc.



Gambar 6. Motor DC.

B. Motor Servo

Motor servo pada aktuator berfungsi untuk menggerakkan *gripper* naik-turun dan mengambil objek. Motor servo bekerja dengan perintah berupa pulsa yang diberikan oleh mikrokontroler dan bekerja dengan tegangan 4,8 Vdc hingga 6,8 Vdc.

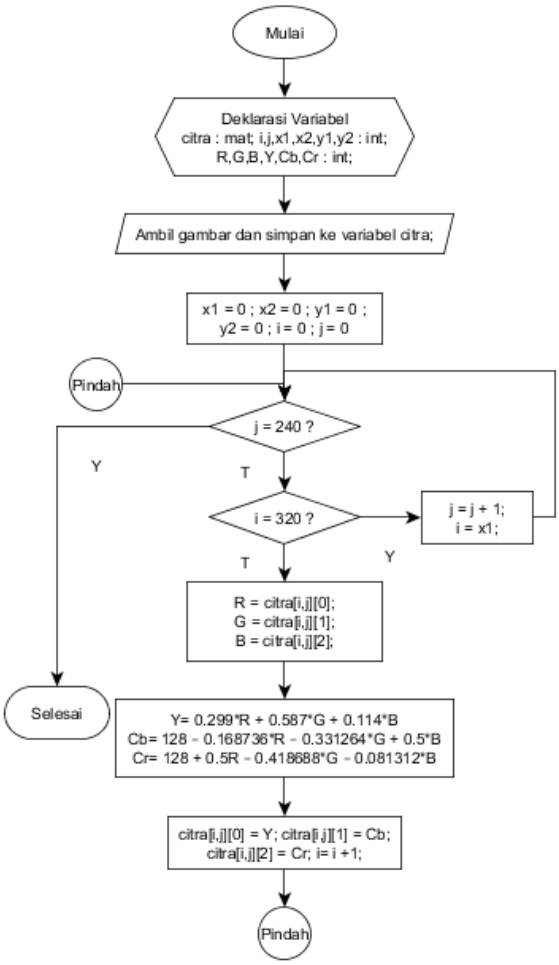


Gambar 7. Motor Servo.

ALGORITMA

Setelah melakukan perancangan dan integrasi antar perangkat keras, selanjutnya membuat perancangan algoritma yang akan diterapkan pada sistem ini.

Rancangan algoritma yang digunakan untuk mengkonversi model warna citra dari RGB menjadi YCbCr ditunjukan pada gambar 8. Pertama-tama modul kamera akan mengambil gambar objek, kemudian gambar tersebut yang akan menjadi masukan ke program, berikut ini flowchart untuk konversi model warna dari RGB menjadi YCbCr

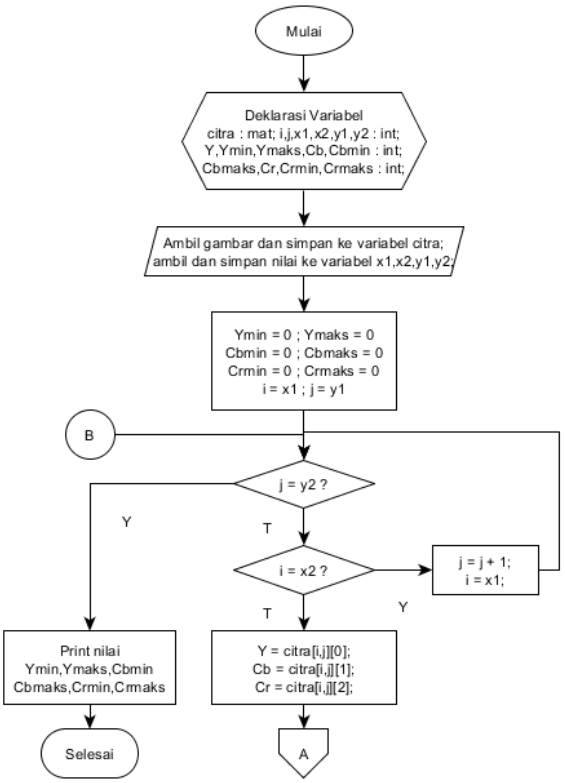


Gambar 8. Flowchar konversi model warna RGB menjadi YCbCr

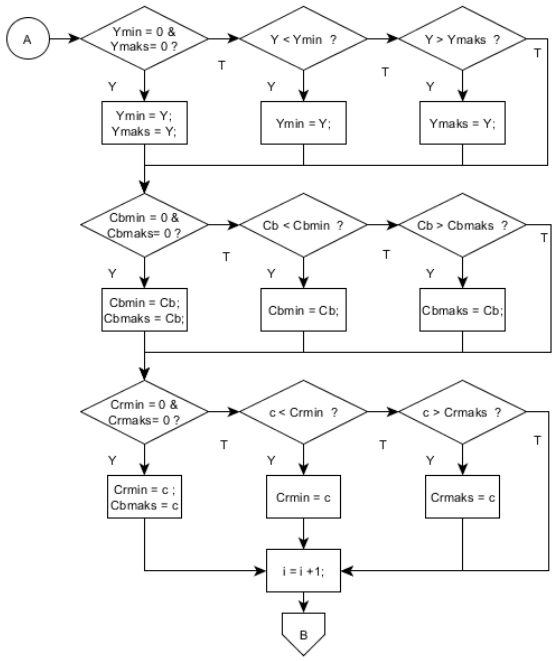
Gambar 8 di atas menunjukkan flowchart konversi model warna citra dari RGB menjadi YCbCr, proses konversi dilakukan dengan mengambil nilai warna dari seluruh *pixel*, kemudian nilai warna tersebut dimasukkan ke dalam persamaan 1.a, 1.b dan 1.c, dan nilai hasil persamaan tersebut disimpan kembali ke dalam nilai pada *pixel* yang sama.

$$Y = 0 + 0.299R + 0.587G + 0.114B \quad (1.a)$$
$$Cb = 128 - 0.168736R - 0.331264G + 0.5B \quad (1.b)$$
$$Cr = 128 + 0.5R - 0.418688G - 0.081312B \quad (1.c)$$

Untuk mengetahui atau mengidentifikasi warna dari objek, pertama-tama modul kamera akan mengambil gambar objek dan dikonversi ke model warna YCbCr, kemudian gambar tersebut disimpan di komputer, gambar inilah yang akan menjadi masukan ke program, berikut ini flowchart untuk identifikasi nilai warna objek:



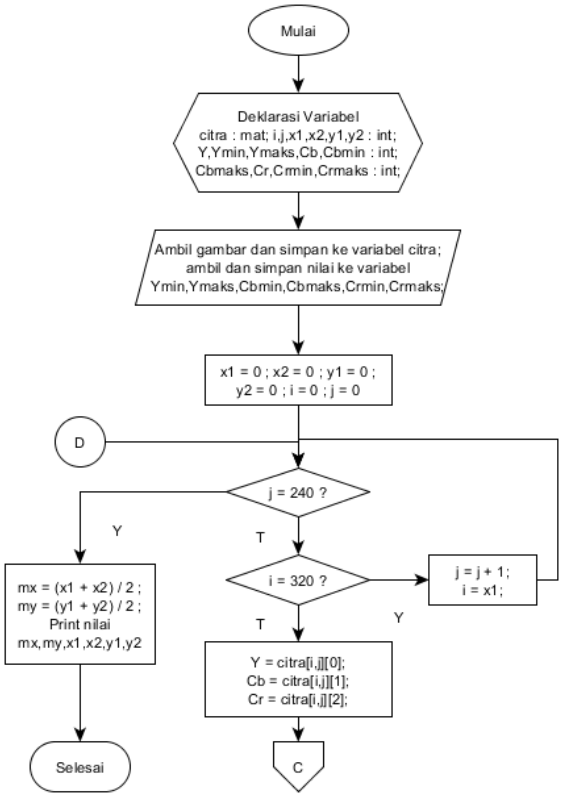
Gambar 9. Flowchart identifikasi objek (1/2).



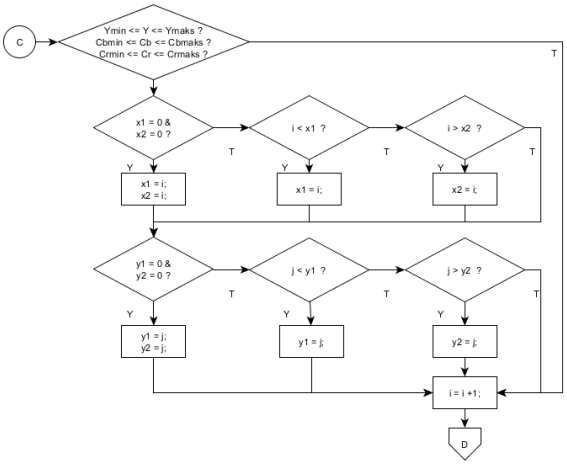
Gambar 10. Flowchart identifikasi objek (2/2).

Gambar 9 dan 10 menunjukkan flowchart identifikasi nilai warna objek yang akan dideteksi dengan mencari nilai warna pada objek tersebut. Sebelum mencari nilai warnanya, harus ditentukan terlebih dahulu posisi atau koordinat dimana objek berada dengan cara menentukan 4 nilai dalam *pixel* (x1,x2,y1,y2) yang akan menjadi titik-titik sudut sebuah segi 4 {(x1,y1) ; (x1,y2) ; (x2,y1) ; (x2,y2)}. Kemudian dilakukan pencarian nilai warna pada seluruh area di dalam segi 4 tersebut, nilai inilah yang nantinya akan menjadi nilai acuan dalam proses pendeteksian objek.penghitungan dengan mensubstitusi nilai R, G, dan B ke dalam persamaan 1, 2, dan 3. Kemudian hasilnya disimpan kembali ke dalam nilai pixel yang diambil.

Rancangan algoritma yang digunakan untuk mendeteksi objek berdasarkan warna menggunakan metode gambar yang telah diambil oleh modul kamera akan menjadi masukan ke program, berikut ini flowchart untuk pendeteksian objek:



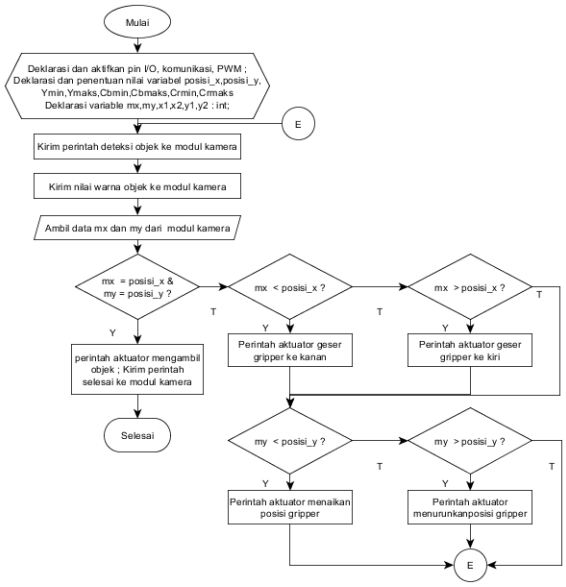
Gambar 11. Flowchart pendeteksian objek (1/2).



Gambar 12. Flowchart pendeteksian objek (2/2).

Gambar 11 dan 12 menunjukkan flowchart pendeteksian objek untuk mengetahui keberadaan dan posisi objek pada citra. Setelah memperoleh citra yang ingin diolah, tahap berikutnya adalah mengambil nilai warna objek yang ingin dideteksi sebagai pembanding, nilai ini didapatkan dari proses identifikasi objek. Kemudian proses pendeteksian ini dilakukan dengan cara mencari *pixel* mana saja yang memiliki nilai warna yang sesuai dengan nilai pembanding tersebut, jika ditemukan ada *pixel* dengan nilai yang sesuai, maka koordinat atau *pixel* tersebut langsung disimpan. Pencarian ini dilakukan ke seluruh *pixel* dari *pixel* pojok kiri atas sampai pojok kanan bawah. Setelah menyelasaikan proses ini, ke-4 nilai *pixel* yang menjadi titik sudut sebuah segi 4 yang melingkupi seluruh *pixel* yang terdeteksi (x1,x2,y1,y2) telah diperoleh. Dari ke-4 nilai tersebut maka dapat dicari posisi titik tengah objek yang terdeteksi tersebut. Kemudian informasi mengenai posisi objek tersebut akan dikirim ke mikrokontroler.

Berikut ini adalah rancangan algoritma di mikrokontroler yang digunakan untuk memberikan perintah deteksi objek kepada modul kamera, menerima dan mengolah data berupa informasi keberadaan objek dan menentukan perintah yang akan diberikan ke aktuator untuk menyesuaikan posisi *gripper* dengan objek lalu mengambilnya

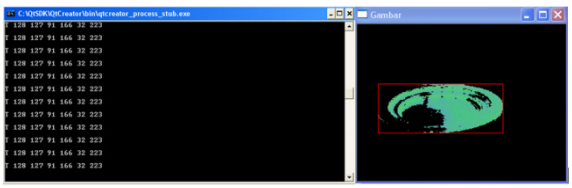


Gambar 13. Flowchart mikrokontroler secara keseluruhan.

Gambar 13 menunjukkan proses kerja mikrokontroler secara keseluruhan dari member perintah ke modul kamera, mengolah informasi berupa posisi objek yang diterima dari modul kamera, member perintah ke actuator untuk menyesuaikan posisi gripper dan juga perintah untuk mengambil objek tersebut.

3. PENGUJIAN

Tahap pertama pengujian dilakukan terhadap algoritma yang digunakan untuk mendeteksi keberadaan dan posisi objek. Pengujian ini dilakukan menggunakan warna RGB dan YCbCr. Untuk mengambil data berupa posisi titik tengah objek pada citra digunakan sebuah program untuk menampilkannya di layar monitor, tampilan tersebut akan ditunjukkan pada gambar 13.



Gambar 13. Tampilan monitor hasil deteksi.

Intensitas Cahaya (lux)	Default		YCbCr			RGB		
	mx	my	mx	my	Persentase Keberhasilan	mx	my	Persentase Keberhasilan
17	139	126	126	139	90.52%	134	146	90.10%
25	127	136	129	135	98.96%	129	133	98.13%
47	127	129	128	135	97.19%	129	136	96.46%
60	127	129	128	135	97.19%	129	135	96.88%
73	127	129	128	135	97.19%	129	135	96.88%
80	127	128	128	135	96.77%	130	135	96.15%
92	127	128	128	127	99.27%	129	135	96.46%
106	127	128	128	126	98.85%	129	135	96.46%
113	127	129	128	127	98.85%	130	135	96.56%
122	127	130	128	126	98.02%	129	135	97.29%
135	128	129	128	126	98.75%	129	135	97.19%
146	127	129	128	126	98.44%	129	135	96.88%
155	127	128	128	126	98.85%	129	135	96.46%
165	127	129	128	126	98.44%	128	135	97.19%
180	127	128	128	126	98.85%	128	135	96.77%
Rata- rata					97.74%			96.39%

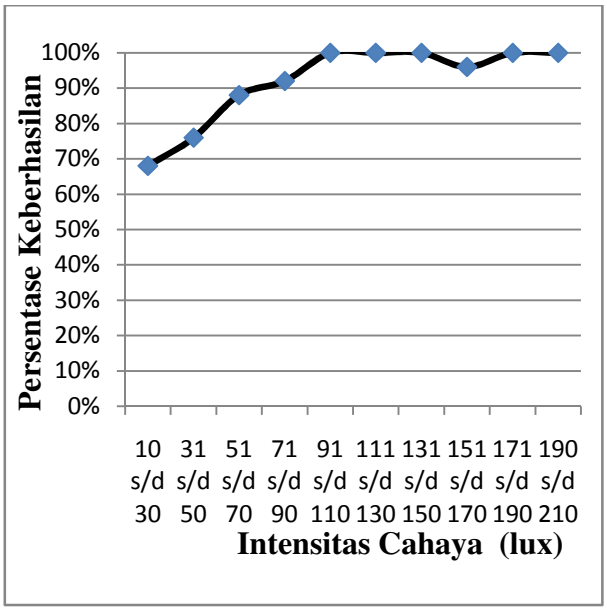
Tabel 1. Data pengujian algoritma pendeteksian.

Data yang diperoleh dari hasil pengujian ini ditampilkan pada tabel 1 yang berisi nilai pixel yang menyatakan posisi titik tengah objek terhadapa citra, nilai intensitas cahaya di sekitar objek yang diukur menggunakan lux meter, dan ketepatan dalam menentukan posisi titik tengah objek. Berdasarkan tabel 1, tingkat ketepatan rata-rata penentuan posisi titik tengah sebesar 97,74% menggunakan warna YCbCr.

Karena persentase rata-rata ketepatan dalam menentukan posisi titik tengah YCbCr lebih baik dibandingkan RGB, maka sistem pendeteksian yang diterapkan pada saat mengambil objek menggunakan warna YCbCr. Setelah menyelesaikan pengujian pada algoritma pendeteksian maka dilanjutkan dengan pengujian sistem saat mendeteksi dan mengambil objek tersebut.Data hasil pengujian ditampilkan dalam tabel 2.

Intensitas Cahaya (lux)	Jumlah Percobaan	Jumlah Berhasil	Jumlah Gagal	Waktu rata-rata (detik)	Persentase Keberhasilan
10 s/d 30	50	42	8	8.43	84.0%
31 s/d 50	50	43	7	8.67	86.0%
51 s/d 70	50	46	4	8.21	92.0%
71 s/d 90	50	48	2	7.92	96.0%
91 s/d 110	50	49	1	9.14	98.0%
111 s/d 130	50	50	0	8.53	100.0%
131 s/d 150	50	50	0	9.17	100.0%
151 s/d 170	50	49	1	9.53	98.0%
171 s/d 190	50	50	0	8.52	100.0%
190 s/d 210	50	50	0	8.75	100.0%
Rata - Rata					95.4%

Tabel 2. Data pengujian sistem saat mendeteksi dan mengambil objek.



Gambar 14. Grafik keberhasilan pengujian sistem.

Berdasarkan data pengujian dari tabel 2 dan grafik keberhasilan pada gambar 14, pengujian dilakukan dalam intensitas cahaya ruangan yang berbeda-beda dengan interval 20 *lux* sebanyak 50 kali percobaan, sehingga didapatkan persentase keberhasilan rata-rata mencapai 95,4 % dalam 500 kali percobaan. Tingkat keberhasilan sistem tampak linier dengan perubahan intensitas cahaya yang semakin cerah sampai dengan nilai 210 *lux*.

4. SIMPULAN DAN SARAN

Dari hasil penelitian, ujicoba dan analisa data yang telah dilakukan, dapat diambil kesimpulan bahwa :

- Modul kamera CMUcam3 yang digunakan dapat bekerja dengan baik dalam mengambil gambar dan mendeteksi objek dari gambar tersebut dengan resolusi 320 x 240 *pixel*.
- Berdasarkan hasil pengujian sistem dalam mendeteksi dan menentukan titik tengah objek menggunakan metode YCbCr dengan kondisi intensitas cahaya di sekitar objek 17 – 180 lux, didapatkan rata-rata tingkat keberhasilan mencapai 97.74%.
- Telah dapat dibangun sebuah sistem pendeteksi objek yang mampu mendeteksi

keberadaaan dan posisi objek menggunakan metode YCbCr yang handal dalam kondisi cahaya yang berbeda. Hal ini berdasarkan pada persentase tingkat keberhasilan selama penelitian dengan kondisi intensitas cahaya di sekitar objek 10 - 210 lux yang dilakukan sebanyak 500 kali percobaan, diperoleh rata-rata tingkat keberhasilan mencapai 95.4% yang ditunjukkan pada grafik gambar 14.

5. DAFTAR PUSTAKA

[1] Solomon, C., & Breckon, T., Fundamental of Digital Image processing, Wiley-Blackwell, 2011.

[2] Richardson, I. E. G., H.264 and MPEG Video Compression, Wiley-Interscience, 2003.

[3] Hamilton, E., JPEG File Interchange Format, 1992.

[4] Koschan, A., & Abidi, M., Digital Color Image Processing, Wiley-Interscience, 2008.

[5] Camera, Wikipedia, (online), (http://en.wikipedia.org/wiki/Camera, diakses tanggal 25 Maret 2013)

[6] Susilo, D.(2010).48 jam kupas tuntas Mikrokontroler MCS51 & AVR. Yogyakarta: Andi.

[7] STMicroelectronics, L298, (online), 2000 (http://pdf1.alldatasheet.com/datasheet-pdf/view/22437/STMICROELECTRONICS/L298/+Q2448UORlHDyRHOIpa/1XXyxeo/1HOO+/datasheet.pdf diakses tanggal 24 Maret 2013).

[8] *Light meter*, Wikipedia, (onlie), (http://en.wikipedia.org/wiki/Light_meter , diakses tanggal 20 Agustus 2013)

[9] Qt Creator, Wikipedia, (online), (http://en.wikipedia.org/wiki/Qt_Creator, diakses tanggal 1 April 2013)

[10] MCS-Electronics, Bascom-AVR, (online), (http://www.mcselec.com/?option=com_content&task=view&id=14&Itemid=41 diakses tanggal 20 Maret 2013)